



**SISTEM MONITORING PENDETEKSI KEBOCORAN  
PIPA AIR BERBASIS IOT**

**PROPOSAL TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat Untuk Mengambil Mata Kuliah Tugas Akhir

**Disusun oleh :**

**Nama : Erni Azizah**

**NIM : 20040141**

**PROGRAM STUDI DIPLOMA III TEKNIK KOMPUTER  
POLITEKNIK HARAPAN BERSAMA TEGAL**

**2023**

## HALAMAN PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : Erni Azizah  
Nim : 20040141  
Jurusan / Program Studi : DIII Teknik Komputer  
Jenis Karya : Tugas Akhir

Adalah mahasiswa Program Studi DIII Teknik Komputer Politeknik Harapan Bersama Tegal, dengan ini saya menyatakan bahwa Laporan Tugas Akhir yang berjudul:

### **“SISTEM MONITORING PENDETEKSI KEBOCORAN PIPA AIR BERBASIS IOT”**

Merupakan hasil pemikiran dan kerjasama sendiri secara orisinil dan saya susun secara mandiri dengan tidak melanggar kode etik hak karya cipta. Pada laporan Tugas Akhir ini juga bukan merupakan karya yang pernah diajukan untuk memperoleh gelas akademik tertentu di suatu Perguruan Tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau disebutkan dalam daftar pustaka.

Apabila dikemudian hari ternyata Laporan Tugas Akhir ini terbukti melanggar kode etik hak karya cipta atau merupakan karya yang dikategorikan mengandung unsur plagiatisme, maka saya bersedia untuk melakukan penelitian baru dan menyusun laporannya sebagai Laporan Tugas Akhir, sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini kami buat dengan sesungguhnya.

Tegal , 21 September 2023



( Erni Azizah )

**HALAMAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPERLUAN AKADEMIS**

Sebagai civitas akademik Politeknik Harapan Bersama Tegal, saya yang bertanda tangan dibawah ini :

Nama : Erni Azizah  
Nim : 20040141  
Jurusan/Program : DIII Teknik Komputer  
Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Harapan Bersama Tegal **Hak Bebas Royalti Noneksklusif** (*None-exclusive Royalty Free Right*) atas Tugas Akhir saya yang berjudul :

**“SISTEM MONITORING PENDETEKSI KEBOCORAN PIPA AIR BERBASIS IOT”**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Politeknik Harapan Bersama Tegal berhak menyimpan, mengalih media/formatika, mengelola dalam bentuk pangkalan data (database), merawat dan mempublikasikan Tugas Akhir saya selama tetap mencantumkan nama sebagai penulis/pencipta dan pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Tegal  
Pada Tanggal : 21 September 2023

Yang menyatakan



Erni Azizah  
NIM. 20040141

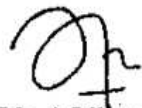
## HALAMAN PERSETUJUAN

Tugas Akhir (TA) yang berjudul “SISTEM MONITORING PENDETEKSI KEBOCORAN PIPA AIR BERBASIS IOT” yang disusun oleh Erni Azizah, NIM 20040141 telah mendapat persetujuan pembimbing dan siap dipertahankan di depan tim penguji Tugas Akhir (TA) Program Studi DIII Teknik Komputer Harapan Bersama Tegal.

Tegal, 3 Juli 2023

Menyetujui,

Pembimbing I,



Ida Afrilliana, ST, M.Kom  
NIPY. 12.013.168

Pembimbing II,



Abdul Basit, S.Kom., MT  
NIPY. 01.015.198

## HALAMAN PENGESAHAN

Judul : SISTEM MONITORING PENDETEKSI KEBOCORAN  
PIPA AIR BERBASIS IOT  
Nama : Erni Azizah  
NIM : 20040141  
Program Studi : Teknik Komputer  
Jenjang : Diploma III

**Dinyatakan LULUS setelah dipertahankan di depan Tim Penguji Tugas Akhir Program Studi Diploma III Teknik Komputer Politeknik Harapan Bersama Tegal**

Tegal, September 2023

Tim Penguji :

Pembimbing I,



Ida Afriliana, ST, M.Kom  
NIPY. 12.013.168

Pembimbing II



Abdul Basit, S.Kom., MT  
NIPY. 01.015.198

Ketua Penguji,



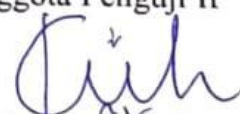
Rais, S.Pd, M.Kom  
NIPY. 07.011.083

Anggota Penguji I



M. Teguh Prihandoyo, M.Kom  
NIPY. 02.005.012

Anggota Penguji II



Abdul Basit, S.Kom., MT  
NIPY. 01.015.198

Mengetahui,

Ketua Program Studi DIII Teknik Komputer,  
Politeknik Harapan Bersama Tegal



Ida Afriliana, ST, M.Kom  
NIPY. 12.013.168

## HALAMAN MOTO

*“Allah tidak membenci seseorang melainkan sesuai dengan kesanggupannya.”*

(Q.S Al-Baqarah, 2:286)

*“Maka sesungguhnya bersama kesulitan itu ada kemudahan.*

*Sesungguhnya bersama kesulitan itu ada kemudahan”*

(Q.S Al-Insyirah, 94:5-6)

*“Untuk masa-masa sulitmu, biarlah Allah yang menguatkanmu. Tugasmu hanya berusaha agar jarak antara kamu dengan Allah tidak pernah jauh.”*

*“Orang lain gak akan paham struggle dan masa sulitnya kita, yang mereka ingin tahu hanya bagian success storiesnya aja. Jadi berjuanglah untuk diri sendiri meskipun gak akan ada yang tepuk tangan. Kelak diri kita di masa depan akan sangat bangga dengan apa yang kita perjuangkan hari ini.*

*Jadi tetap berjuang ya.”*

## HALAMAN PERSEMBAHAN

Tugas Akhir ini dipersembahkan kepada :

1. Bapak Agung Hendarto, S.E., M.A., selaku Direktur Politeknik Harapan Bersama Tegal
2. Ibu Ida Afriliana, ST, M.Kom., selaku Ketua Prodi DIII Teknik Komputer Harapan Bersama Tegal
3. Ibu Ida Afriliana, ST, M.Kom., selaku Pembimbing I
4. Bapak Abdul Basit, S.Kom., M.T., selaku Pembimbing II
5. Kepada Bapak dan Ibu  
- Darto  
- Warnidah  
yang selalu memberikan dukungan dan doa
6. Kepada Kakak - Kakaku  
- Soliri, A.Md. Ak.  
- Irwan Sahri, A.Md. T.  
yang selalu memberikan dukungan dan motivasi
7. Teman - teman, sahabat dan saudara yang telah mendoakan, mendukung dan memberi semangat dalam menyelesaikan Tugas Akhir ini.

## ABSTRAK

Air merupakan salah satu kebutuhan primer yang paling penting yang tidak terpisahkan dari kehidupan semua makhluk hidup apalagi bagi manusia. Air sangat penting untuk memenuhi kehidupan sehari-hari dan memiliki banyak kebutuhan. Seperti bagi manusia membutuhkan air untuk keperluan sehari-hari seperti halnya minum, memasak, mencuci, mandi, dan sebagainya. Oleh karena itu manusia tidak mungkin terlepas dari kebutuhan akan air. Karena itulah penyediaan air bersih sangatlah penting diperlukan untuk keperluan konsumsi. Tujuan dalam penelitian ini adalah untuk menghasilkan suatu sistem yang dapat memonitoring kecepatan debit air, saluran pipa, dan status aliran air melalui website berbasis IoT dan dapat mengetahui efektifitas alat pada pipa air. Metode penelitian yang digunakan dalam penelitian ini adalah metode SDLC (waterfall). Waterfall adalah metode kerja yang menekankan fase berurutan dan sistematis. Disebut air terjun karena prosesnya mengalir "ke bawah" dalam satu arah seperti air terjun. Metode waterfall ini harus dilakukan secara berurutan sesuai tahapan yang ada. Pengujian sistem informasi ini dimaksudkan untuk menguji apakah seluruh bagian website berfungsi sesuai dengan yang diharapkan. Dari hasil pengujian, Sistem Monitoring Pendeteksi Kebocoran Pipa Air berbasis IoT dapat bekerja dengan baik. Berdasarkan analisa sistem monitoring pendeteksi kebocoran pipa air yang dapat diambil kesimpulan bahwa Sistem dapat penelitian ini adalah sistem yang dapat memonitoring kecepatan debit air dan pendeteksi kebocoran pada pipa air melalui website berbasis monitoring.

Kata kunci : *Monitoring, IoT, Kebocoran, Debit air*



## KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Allah SWT, Tuhan Yang Maha Pengasih dan Maha Penyayang yang telah melimpahkan segala rahmat, hidayah dan inayah-Nya hingga terselesaikan laporan Tugas Akhir dengan judul “SISTEM MONITORING PENDETEKSI KEBOCORAN PIPA AIR BERBASIS IOT”

Tugas Akhir merupakan suatu kewajiban yang harus dilaksanakan untuk memenuhi salah satu syarat kelulusan dalam mencapai derajat Ahli Madya Komputer pada program studi DIII Teknik Komputer Politeknik Harapan Bersama Tegal. Selama melaksanakan penelitian dan kemudian tersusun dalam laporan Tugas Akhir ini, banyak pihak yang telah memberikan bantuan, dukungan, dan bimbingan.

Pada kesempatan ini, tidak lupa diucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Agung Hendarto, S.E, MA., selaku Direktur Politeknik Harapan Bersama Tegal.
2. Ibu Ida Afriliana, ST, M.Kom selaku Ketua Program Studi DIII Teknik Komputer Politeknik Harapan Bersama Tegal.
3. Ibu Ida Afriliana, ST, M.Kom dan Bapak Abdul Basit, S.Kom., MT selaku pembimbing Tugas Akhir
4. Kedua Orang Tua, dan keluarga yang selalu memberikan doa dan dukungan.
5. Teman-teman, sahabat dan saudara yang telah mendoakan, mendukung dan memberi semangat dalam menyelesaikan Tugas Akhir ini.
6. Semua pihak yang telah mendukung, membantu serta mendoakan penyelesaian penulisan laporan ini.

Tegal, 3 Juli 2023



Erni Azizah

20040141

## DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN PERNYATAAN KEASLIAN .....	ii
HALAMAN PERSETUJUAN PUBLIKASI.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PENGESAHAN.....	v
HALAMAN MOTO .....	vi
HALAMAN PERSEMBAHAN .....	vii
ABSTRAK .....	viii
KATA PENGANTAR .....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR .....	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.1 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan dan Manfaat.....	3
1.5 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA.....	7
2.1 Teori Terkait.....	7
2.1 Landasan Teori.....	9
BAB III METODOLOGI PENELITIAN.....	28
3.1 Alat Penelitian .....	28
3.2 Prosedur Penelitian.....	28
3.3 Metode Pengumpulan Data .....	30
3.4 Tempat dan Waktu Pelaksanaan.....	31
BAB IV ANALISA DAN PERANCANGAN SISTEM.....	32
4.1 Analisa Permasalahan.....	32
4.2 Analisa Kebutuhan Sistem .....	33
4.3 Perancangan Sistem.....	33
4.4 Desain Input/Output .....	46
BAB V HASIL DAN PEMBAHASAN.....	50

5.1	Implementasi System .....	50
5.2	Hasil Pengujian.....	54
BAB VI KESIMPULAN DAN SARAN .....		60
6.1	Kesimpulan.....	60
6.2	Saran.....	60
DAFTAR PUSTAKA .....		61
LAMPIRAN.....		62

## DAFTAR TABEL

	Halaman
Tabel 2. 1 Simbol Use Case Diagram .....	18
Tabel 2. 2 Simbol Activity Diagram .....	19
Tabel 2. 3 Simbol Squence Diagram .....	21
Tabel 2. 4 Simbol Class Diagram .....	22
Tabel 4. 1 Tabel Monitoring.....	44
Tabel 4. 2 Tabel rumah A .....	44
Tabel 4. 3 Tabel rumah B.....	44
Tabel 4. 4 Tabel User .....	45
Tabel 4. 5 Tabel Token_User.....	45
Tabel 5. 1 Hasil Pengujian Login.....	55
Tabel 5. 2 Hasil Pengujian Login Salah.....	55
Tabel 5. 3 Hasil Pengujian Rekap Data .....	57

## DAFTAR GAMBAR

	Halaman
Gambar 2. 1 Konsep Aliran M-V-C.....	13
Gambar 2. 2 PHP .....	13
Gambar 2. 3 BOOTSTRAP .....	14
Gambar 3. 1 Prosedur Penelitian.....	29
Gambar 3. 2 Maps Penelitian .....	31
Gambar 4. 1 Use Case Diagram.....	34
Gambar 4. 2 Activity Registrasi.....	35
Gambar 4. 3 Activity Login .....	35
Gambar 4. 4 Activity Dashboard .....	36
Gambar 4. 5 Activity rekap data rumah A .....	37
Gambar 4. 6 Activity rekap data rumah B .....	37
Gambar 4. 7 Activity Profile Admin.....	38
Gambar 4. 8 Activity Logout .....	39
Gambar 4. 9 Sequence Registrasi.....	39
Gambar 4. 10 Sequence Login .....	40
Gambar 4. 11 Sequence Dashboard .....	40
Gambar 4. 12 <i>Sequence</i> rekap data rumah A .....	41
Gambar 4. 13 Sequence rekap data rumah B .....	41
Gambar 4. 14 Sequence Profile Admin.....	42
Gambar 4. 15 Sequence Logout.....	42
Gambar 4. 16 Class Diagram .....	43
Gambar 4. 17 Halaman Registrasi akun.....	46
Gambar 4. 18 Halaman Login.....	46
Gambar 4. 19 Halaman Dashboard .....	47
Gambar 4. 20 Desain halaman rekap data rumah A.....	47
Gambar 4. 21 Desain halaman rekap data rumah B.....	48
Gambar 4. 22 Halaman Profile Admin .....	48
Gambar 4. 23 Halaman Logout.....	49
Gambar 5. 1 Halaman Registrasi.....	51
Gambar 5. 2 Halaman Login.....	51
Gambar 5. 3 Halaman Dashboard .....	52
Gambar 5. 4 Halaman rekap data rumah A.....	52
Gambar 5. 5 halaman rekap data rumah B.....	53
Gambar 5. 6 Halaman Profile Admin .....	53
Gambar 5. 7 Grafik Data Debit Rumah A.....	58
Gambar 5. 8 Grafik Data Debit Rumah B.....	59

## DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Source Code Halaman Dashboard.....	A-1
Lampiran 2. Surat Kesediaan Membimbing TA .....	B-1
Lampiran 3. Surat Izin Observasi .....	C-1

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kebutuhan dalam kehidupan manusia terdiri dari kebutuhan primer dan kebutuhan sekunder. Air merupakan salah satu kebutuhan primer yang paling penting yang tidak terpisahkan dari kehidupan semua makhluk hidup apalagi bagi manusia. Air sangat penting untuk memenuhi kehidupan sehari-hari dan memiliki banyak kebutuhan. Seperti bagi manusia membutuhkan air untuk keperluan sehari-hari seperti halnya minum, memasak, mencuci, mandi, dan sebagainya. Oleh karena itu manusia tidak mungkin terlepas dari kebutuhan akan air. Karena itulah penyediaan air bersih sangatlah penting diperlukan untuk keperluan konsumsi. [1]

Dibeberapa daerah penggunaan air biasanya diatur oleh petugas yang sudah di beri kewenangan dalam upaya penyediaan air bersih. penggunaan air bersih biasanya diatur oleh petugas Himpunan Penduduk Pemakaian Air Minum. [2] Permasalahan yang dihadapi petugas selain sebagai penyedia untuk memenuhi kebutuhan air bersih, petugas juga harus selalu siaga dengan adanya dugaan kebocoran air. Kebocoran air ini tidak mudah diketahui jika tidak di cek secara manual ke pusat aliran air. Hal ini di rasa kurang efektif dalam proses pengecekan. Mulai dari mengecek debit meteran air dan mencatat pemakaian debit air. Sehingga petugas membutuhkan proses tersebut dengan waktu yang tidak singkat.

Dengan perkembangan teknologi yang semakin berkembang, hal tersebut dapat teratasi oleh adanya internet of things (IOT) dengan bantuan sensor Water Flow yang dirasa cukup efektif dalam melakukan monitoring terhadap pengukuran aliran air. [3] Hal ini berfungsi untuk melihat debit air yang mengalir pada pusat pipa atau saluran yang melewati sensor. Manfaat adanya sensor ini memberikan kemudahan dalam melakukan pengecekan suatu debit air yang berkurang akibat adanya dugaan kebocoran pada pipa.

Berdasarkan uraian diatas, peneliti mencoba membuat sistem monitoring pengukuran kecepatan debit air dan deteksi kebocoran untuk mempermudah pekerjaan suatu petugas dengan cara memonitoring debit air. Dengan melihat debit air jarak jauh melalui *website*. Jadi, petugas mengetahui debit aliran air pada pusat pipa. Hal ini bertujuan agar pegawai bekerja lebih efektif dalam bekerja.

### **1.1 Rumusan Masalah**

1. Bagaimana mengimplementasikan sistem monitoring deteksi kebocoran pipa air berbasis *IoT*
2. Bagaimana mengetahui efektifitas alat pada pipa air berbasis *IoT* menggunakan sensor Waterflow, sensor Selenoid Valve.

### **1.3 Batasan Masalah**

Batasan masalah untuk Tugas Akhir ini adalah :

1. sistem diakses dalam bentuk *website*.



2. sistem hanya memonitoring elemen penting yaitu perbandingan kecepatan debit air rumah A dan rumah B, saluran pipa air, dan status pipa.
3. subjek penelitian merupakan perbandingan debit air antar rumah dan deteksi kebocoran pipa air.
4. perancangan sistem memakai UML.

## **1.4 Tujuan dan Manfaat**

### **1.4.1 Tujuan**

Tujuan dalam penelitian ini adalah untuk menghasilkan suatu sistem yang dapat memonitoring kecepatan debit air rumah A dan B, saluran pipa, dan status aliran air melalui *website* berbasis *IoT* dan dapat mengetahui efektifitas alat pada pipa air.

### **1.4.2 Manfaat**

#### **1. Bagi Mahasiswa**

- a. Menambahkan pengalaman sekaligus menerapkan teori yang didapat untuk dipraktikkan didunia kerja nantinya.
- b. Menambah wawasan bagi mahasiswa tentang perkembangan ilmu teknologi.
- c. Menyajikan hasil-hasil yang diperoleh dalam bentuk laporan.

## **2. Bagi Politeknik Harapan Bersama**

- a. Memberikan kesempatan para mahasiswa mempelajari bagaimana cara membuat sistem monitoring pendeteksi kebocoran pipa air berbasis *IoT*.
- b. Menjadi media pembelajaran untuk mahasiswa dalam meningkatkan pengetahuan tentang membuat sistem monitoring pendeteksi kebocoran pipa air berbasis *IoT*.
- c. Sebagai masukan untuk mengevaluasi sejauh mana mahasiswa memahami materi apa yang didapat selama perkuliahan.

## **3. Bagi Instansi**

- a. Memberikan kemudahan kepada petugas dalam memonitoring kecepatan debit air dan mendeteksi kebocoran.
- b. Meringankan petugas dalam mengecek keadaan pipa.

### **1.5 Sistematika Penulisan**

Adapun sistematika penulisan pada laporan Tugas Akhir ini adalah sebagai berikut :

#### **BAB I : PENDAHULUAN**

Pada bab ini, penulis menuliskan sebagian besar materi berupa penjelasan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan laporan Tugas Akhir.

**BAB II : TINJAUAN PUSTAKA**

Bab ini menjelaskan tentang gambaran dasar yang mendukung penulisan tugas akhir, meliputi penelitian yang digunakan, teori masalah dan gambaran singkat implementasi yang digunakan.

**BAB III : METODOLOGI PENELITIAN**

Bab ini membahas fase atau tahapan perencanaan dengan menggunakan metode, teknik dan alat yang berbeda seperti metode penelitian, metode pengumpulan data, dan di mana dan kapan melakukan penelitian, seperti dalam proposal tugas akhir.

**BAB IV : ANALISI DAN PERANCANGAN SISTEM**

Bab ini menjelaskan analisis semua masalah yang ada, memecahkan masalah yang muncul dengan bantuan penelitian. Bab ini juga menjelaskan secara detail mengenai desain penelitian, dan desain umum dari sistem yang akan dibangun serta desain yang lebih detail. Perancangan sistem meliputi analisis masalah, kebutuhan perangkat keras dan perangkat lunak, perancangan (UML), perancangan database dan tabel.

**BAB V : HASIL DAN PEMBAHASAN**

Berisi laporan hasil penelitian. Deskripsi hasil penelitian dapat diimplementasikan dalam bentuk teori, model, perangkat lunak, grafik atau bentuk representatif lainnya. Bagian ini juga menganalisis bagaimana hasil penelitian dapat menjawab pertanyaan tentang hakikat masalah.

## **BAB VI : KESIMPULAN DAN SARAN**

Berisi kesimpulan (hasil yang berhasil diselesaikan sesuai ruang lingkup batasan masalah) dan saran (untuk masalah yang tidak terselesaikan seperti pengembangan dan perbaikan) dari kasus Tugas Akhir.

## **DAFTAR PUSTAKA**

Daftar pustaka ini berisi tentang judul buku, artikel, dan jurnal yang digunakan sebagai referensi di dalam penyusunan laporan atau karya tulis.

## **LAMPIRAN**

Lampiran ini memuat keterangan penunjang sehubungan dengan atau permasalahan yang dianalisis.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Teori Terkait**

Berikut ini beberapa literature-literatur pendukung dalam pembuatan Tugas Akhir ini :

Ray Samudra Bagas D, Umar Ali Ahmad dan Jati Satria Wicaksana telah merancang Aplikasi Mobile Survei Kualitas Dan Debit Air Berbasis Firebase Studi Kasus Pdam Kabupaten Madiun (2022). Implementasi dan pengujian aplikasi yang telah dilakukan selama pengerjaan Tugas Akhir ini didapatkan suatu kesimpulan dalam pembuatan Aplikasi Mobile Survei Kualitas dan Debit Air Berbasis Firebase Studi Kasus PDAM Kabupaten Madiun ini adalah Sistem pada Aplikasi Monitoring sudah berjalan dengan baik sesuai dengan rancangan dan tujuan pembuatan aplikasi untuk Monitoring Kualitas dan Debit Air yang memenuhi 22/22 kebutuhan sistem dari pihak PDAM Kabupaten Madiun. Validasi fungsionalitas aplikasi telah dilakukan uji coba dengan menggunakan metode Blackbox dan sesuai dengan harapan. [4]

Tubagus Sundana, Farchan Aditya Johari, dan Fathan Al Ariiq dari Prodi Teknik Elektro, telah merancang prototype sistem monitoring kebocoran pipa distribusi air berbasis scada (2022). Dalam analisis perancangan diperoleh bahwa sistem yang dibuat sudah dapat melokalisir letak kebocoran pipa distribusi air pada dua lokasi, dengan menggunakan 3

titik pengukuran. Komunikasi terminal RTU pada setiap titik menggunakan jaringan RS485. [5]

Nurul Afthiroh, Nurul Hidayah, dan Samudi dari Prodi Teknik Komputer , telah merancang sistem monitoring penggunaan air pdam berbasis *website* (2019). Berdasarkan analisis sistem monitoring penggunaan air pdam berbasis website yang dapat diambil kesimpulan yaitu prototype yang dirancang dapat memantau penggunaan air dan kualitas kejernihan secara realtime melalui jaringan internet menggunakan aplikasi web browser, penggunaan alat ini diperuntukkan untuk jenis golongan R1 atau rumah tangga dengan klasifikasi pemakaian air 0-10 M<sup>3</sup>. [6]

Jeffry Ignatius Dimu, telah merancang alat deteksi letak kebocoran pipa menggunakan sensor flow meter berbasis Arduino (2022). Berdasarkan hasil dari penelitian bahwa sistem pendeteksi kebocoran pipa berbasis Arduino bekerja dengan baik dimana alat ini menghitung debit air yang mengalir pada tiap flowmeternya dan membandingkan nilai debit air untuk mendeteksi kebocoran pipa. [7]

Hendra Kusuma dan teman-teman, telah merancang prototype pendeteksi kebocoran pipa berbasis iot menggunakan nodemcu esp8266 melalui dashboard adafruit.io (2021). Dari hasil penelitian yang telah dilakukan maka dapat disimpulkan bahwa alat pendeteksi kebocoran pipa dapat digunakan untuk mencatat arus air dan tekanan udara yang masuk kedalam pipa namun pengukuran hanya efektif pada pipa berukuran ½. Data

yang dicatat dapat memudahkan user untuk mengetahui keadaan tekanan air yang sedang terjadi secara realtime. [8]

Akhmad Miftah F dan Rudi Hermawan, telah merancang sistem pendeteksi tekanan air berbasis internet of thing (iot) di pdam tirta galuh kabupaten ciamis (2020). Hasil dari pembuatan sistem monitoring tekanan air berbasis iot bahwa sistem monitoring tekanan air berbasis iot di pdam tirta galuh ciamis yang bisa memonitoring tekanan air real time dengan memanfaatkan sinar matahari sebagai sumber daya yang dionversi oleh solar panel menjadi energy listrik. [9]

## **2.1 Landasan Teori**

### **2.2.1 Sistem**

Sistem merupakan suatu kesatuan dari elemen atau komponen yang dihubungkan secara bersama untuk memungkinkan transfer materi, *energy* atau informasi. Suatu sistem digambarkan sebagai satu kesatuan yang utuh bagian-bagiannya saling berkaitan dan memiliki unsur penggerak.

Ada beberapa teori yang mendefinisikan sistem yang dikemukakan oleh berbagai ahli, diantaranya :

- a. Menurut Prajudi (2015) “ Bahwa pengertian sistem adalah suatu jaringan dari prosedur-prosedur yang berkaitan satu sama lain menurut skema atau pola yang bulat untuk menggerakkan suatu fungsi yang utama dan suatu usaha ataupun urusan.”

- b. Menurut W.J.S. Poerwadarminta (2015) “Sistem adalah sekelompok bagian-bagian atau alat dan sebagainya yang bekerja bersama-sama untuk melakukan sesuatu maksud.”

### **2.2.2 Monitoring**

Monitoring adalah evaluasi berkelanjutan dari fungsionalitas fitur program dalam kaitannya dengan entri data/rencana penggunaan nutrisi kelompok sasaran dalam kaitannya dengan ekspektasi yang direncanakan.

Suatu program tidak dapat dibuat hanya dengan evaluasi atau pemantauan saja, karena dalam program pemantauan tidak dapat dirancang tanpa mengetahui bagaimana data dan informasi akan dievaluasi dan seberapa efektifnya, karena tidak dapat dikumpulkan dan disimpan datanya.

### **2.2.3 IoT (Internet Of Things)**

*Internet of Things* adalah perangkat lunak, sensor, dan jaringan yang saling terhubung yang memungkinkan objek untuk berbagi informasi satu sama lain. Pada tahun 2013, *Internet of Things Standard (IOT-GSI)* mendefinisikan *Internet of Things* sebagai "infrastruktur untuk masyarakat informasi". IoT memungkinkan deteksi dan kendali jarak jauh objek dalam infrastruktur jaringan yang ada. Menciptakan peluang baru untuk integrasi langsung dunia fisik ke dalam sistem komputer dan meningkatkan efisiensi, akurasi, dan keuntungan finansial sistem.



#### 2.2.4 *Klasifikasi Website*

Berdasarkan jenisnya, *website* diklasifikasikan menjadi beberapa kelompok yaitu :

1. *Website Statis*

*Website statis* adalah suatu halaman *website* yang tampilannya tidak berubah (statis). Jika pemilik *website* ingin merubah tampilannya maka harus dilakukan secara manual yaitu dengan mengubah kode situs *web*. *Website* statis umumnya tidak memungkinkan terjadinya interaksi langsung antara pemilik *website* atau administrator dengan pengunjung *website*.

2. *Website Dinamis*

*Website dinamis* adalah jenis *website* yang dirancang khusus untuk dapat *update* konten sesering mungkin. *Website* dinamis dapat disesuaikan dengan kebutuhan anda, baik dari segi tampilan maupun dari fungsional. *Website* dinamis biasanya didesain dengan konsep visual dan interaksi yang tinggi dengan pengguna.

3. *Website Interaktif*

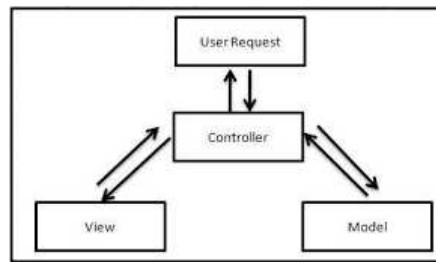
*Website interaktif* adalah ragam situs *web* cara untuk berinteraksi dengan orang lain secara online. Pada umumnya pemakai *website interaktif* merupakan pengguna ataupun komunitas internet yang berperan.

### 2.2.5 *Framework CodeIgniter*

*CodeIgniter* adalah sebuah *framework php* yang bersifat *open source* dan menggunakan metode MVC (*Model, View, Controller*) untuk memudahkan *developer* atau *programmer* dalam membuat aplikasi berbasis *web* tanpa harus membangunnya dari awal.

Kelebihan dari *Framework CodeIgniter* dibanding dengan *framework* lainnya adalah sebagai berikut :

1. Berukuran kecil ukuran yang kecil merupakan keunggulan tersendiri jika dibandingkan *framework* lain yang berukuran besar dan membutuhkan *resource* yang besar dan juga dalam eksekusi maupun penyimpanannya.
2. Menggunakan konsep M-V-C *codeigniter* merupakan konsep M-V-C (*Model - View - Controller*) yang memungkinkan pemisahan antara *layer application-logic* dan *presentation*. Konsep ini memungkinkan kode PHP, query MySQL, Javascript, dan CSS dipisahkan satu sama lain, mengurangi ukuran file dan membuatnya lebih mudah untuk diperbaiki di masa mendatang.



Gambar 2. 1 Konsep Aliran M-V-C

### 2.2.6 PHP

PHP atau *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Dapat dikatakan bahwa bahasa PHP menggambarkan beberapa bahasa pemrograman seperti C, Java dan Perl dan mudah untuk dipelajari. PHP adalah bahasa skrip sisi server di mana pemrosesan data dilakukan di sisi server. Pengertian lain dari PHP adalah singkatan dari *Hypertext Preprocessor*, yaitu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah data dan mengembalikannya ke browser sebagai kode HTML.



Gambar 2. 2 PHP

### 2.2.7 Bootstrap

Bootstrap adalah *framework* untuk membuat desain *web* yang *responsif*. Artinya tampilan *web* yang dibuat oleh bootstrap akan menyelesaikan ukuran layar dari browser yang digunakan baik di desktop, tablet ataupun *mobile device*. Sehingga kita bisa membuat *web* untuk tampilan desktop saja dan apabila dirender oleh *mobile browser* maka tampilan dari *web* yang kita buat tidak bisa beradaptasi sesuai layar. Dengan bootstrap kita juga bisa membangun *web* dinamis atau statis. Keuntungan menggunakan Bootstrap adalah dibangun dengan *Less*, teknik CSS yang sederhana dan mudah digunakan. Kurang juga menawarkan lebih banyak kekuatan dan fleksibilitas daripada CSS biasa.



Gambar 2. 3 BOOTSTRAP

### 2.2.8 JavaScript

JavaScript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang dilihat di browser lebih interaktif, tidak hanya cantik. JavaScript menyediakan beberapa *fungsi* ke dalam halaman *web*, sehingga bisa menjadi sebuah program yang disajikan melalui antarmuka *web*. JavaScript adalah bahasa *script* yaitu bahasa yang tidak memerlukan *compiler* untuk

dapat menjalankannya, tetapi cukup dengan *Interpreter*. Proses penerjemahan tidak diperlukan agar program dapat bekerja. Browser *web Netscape Navigator* dan *Internet Explorer* adalah contoh dari banyak *interpreter*, karena kedua browser ini telah dilengkapi dengan *Interpreter JavaScript*. Tetapi tidak semua browser *web* bisa menjadi *interpreter JavaScript* karena belum tentu browser tersebut dilengkapi dengan *interpreter JavaScript*. JavaScript merupakan bahasa pemrograman yang ringan dan mudah untuk digunakan. Dengan JavaScript ini, situs *web* bukan hanya halaman data dan informasi, tetapi juga program aplikasi dengan antarmuka *web*. JavaScript adalah bahasa pemrograman yang tidak memerlukan lisensi untuk digunakan. Jika *web browser* yang kita gunakan mendukung JavaScript, maka kita bisa langsung membuat aplikasi berbasis *web* menggunakan JavaScript.

### **2.2.9 DataBase Mysql**

*Database* merupakan kumpulan tabel-tabel yang berisi sekumpulan fakta sebagai sumber data yang disimpan dengan media memori yang dikontrol secara digital dengan bantuan program komputer, berguna untuk memudahkan operasi layanan informasi. *MySQL* adalah aplikasi yang digunakan untuk mengolah *database*, sering digunakan untuk membangun aplikasi yang menggunakan *database*.

### 2.2.10 APACHE WEBSERVER

*Apache* adalah server *web* yang tersebar di internet. Program ini pertama kalinya dikembangkan pada sistem operasi lingkungan UNIX. Namun dalam versi *Apache* berikutnya merilis perangkat lunak yang bisa dijalankan pada *Windows NT*. *Apache* mempunyai sejumlah program penunjang yang sangat banyak. Hal ini membagikan layanan yang cukup lengkap untuk para konsumen. *Apache* merupakan nama server *web* yang bertanggung jawab pada *request-response* HTTP dan *log* informasi. Selain itu, *Apache* juga didefinisikan sebagai *web* server yang kompak, modular, mengikuti standar protokol HTTP, dan tentunya sangat populer.

### 2.2.11 HTML

HTML, yang merupakan singkatan dari *Hyper Text Markup Language*, adalah sekumpulan kode program yang menjadi dasar dari representasi visual dari sebuah halaman *web*. Berisi kumpulan data yang disimpan dalam pengenalan tertentu, menggunakan pengenalan tersebut untuk memformat data tersebut. Kode HTML telah dikembangkan dengan berbagai cara dan teknologi baru dalam dunia pemrograman *web*. Secara umum, sebagian besar situs *web* di Internet masih menggunakan HTML sebagai teknologi utamanya

### 2.2.12 UML (Unified Modelling Language)

UML (*Unified Modeling Language*) adalah metode pemodelan visual untuk merancang sistem berorientasi objek atau

untuk menentukan UML, yaitu bahasa yang telah menjadi standar untuk memvisualisasikan sistem perangkat lunak.

#### **a) Tujuan atau Fungsi UML**

Berikut adalah beberapa tujuan atau fungsi UML, antara lain :

- a. Memberi pengguna bahasa pemodelan visual untuk berbagai proses pemrograman dan desain.
- b. Mampu menyusun *best modeling practice*.
- c. Mampu menyediakan model siap pakai dan bahasa pemodelan visual yang ekspresif untuk mengembangkan sistem dan memodifikasi model dengan mudah.
- d. Dapat digunakan untuk rencana dikarenakan sangat komprehensif dan detail dengan informasi detail tentang pengkodean program.
- e. Mengetahui cara memodelkan sistem menggunakan konsep berorientasi objek sehingga tidak hanya digunakan untuk pemodelan perangkat lunak.
- f. Mengetahui cara membuat bahasa pola yang kemudian dapat digunakan oleh manusia dan mesin.


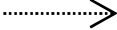

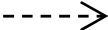



#### **b) Jenis-Jenis UML dan Simbolnya**

##### **1. Use Case Diagram**



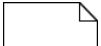
*Use case diagram* adalah salah satu jenis diagram dalam UML yang menggambarkan interaksi antara sistem

dengan aktor, use case diagram juga dapat menggambarkan cara interaksi antara pengguna sistem dengan sistem.

Tabel 2. 1 Simbol Use Case Diagram

Gambar	Nama	Keterangan
	Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
	Generalization	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancertor</i> ).
	Include	Menspesifikasikan bahwa use case sumber secara eksplisit.
	Extend	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.









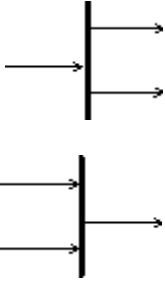

<b>Gambar</b>	<b>Nama</b>	<b>Keterangan</b>
	Use Case	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	Collaboration	Interaksi aturan – aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen – elemennya (sinergi).
	Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

## 2. Activity Diagram

*Activity diagram* atau diagram aksi adalah salah satu jenis diagram dalam UML yang dapat memodelkan proses-proses yang terjadi dalam suatu sistem.

Tabel 2. 2 Simbol Activity Diagram



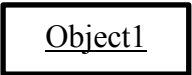
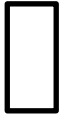
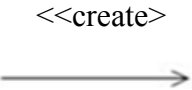
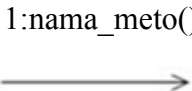
<b>Gambar</b>	<b>Nama</b>	<b>Keterangan</b>
	Activity	Memperlihatkan bagaimana masing - masing kelas antarmuka saling berinteraksi satu sama lain.
	Action	State dari sistem yang mencerminkan eksekusi suatu aksi.
	Initial Node	Bagaimana objek dibentuk atau diawali.



Gambar	Nama	Keterangan
	Final Node	Bagaimana objek dibentuk dan dihancurkan.
	Fork Node	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.
	Decision	Pilihan untuk mengambil keputusan
	Fork/Join	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
	Rake	Menunjukkan adanya dekomposisi

### 3. Sequence Diagram

*Sequence diagram* adalah jenis diagram di UML yang menggambarkan interaksi objek selama periode waktu tertentu. Sequence diagram juga dapat menggambarkan urutan atau langkah-langkah yang harus dilakukan untuk membuat sesuatu seperti use case diagram.

Tabel 2. 3 Simbol Squence Diagram


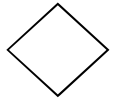
Nama	Simbol	Keterangan
Garis Hidup		Menyatakan kehidupan suatu objek
<i>Actor</i>		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi sendiri.
Object		Menyatakan object yang berinteraksi pesan
Waktu Aktif		Menyatakan object dalam keadaan aktif dan berinteraksi pesan
Pesan tipe create		Menyatakan suatu object membuat object lain, arah panah mengarah pada object yang dibuat
Pesan tipe call		Menyatakan suatu object memanggil oprasi atau metode yang ada pada object lain atau dirinya

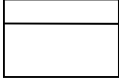

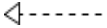
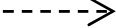

Nama	Simbol	Keterangan
Pesan return	1: keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau suatu metode kembalian ke objek tertentu.
Pesan destroy	destory() 	Suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destory.

#### 4. Class Diagram

*Class diagram* adalah jenis diagram di UML yang digunakan untuk mewakili kelas dan paket di sistem selanjutnya. Sehingga diagram ini dapat memberikan gambaran tentang sistem dan hubungannya.

Tabel 2. 4 Simbol Class Diagram

Gambar	Nama	Keterangan
	Generalization	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> )
	Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

Gambar	Nama	Keterangan
	Class	Himpunan dari objek - objek yang berbagi atribut serta operasi yang sama.
	Collaboration	Deskripsi dari urutan aksi - aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	Dependency	Operasi yang benar - benar dilakukan oleh suatu objek.
	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.

### 2.2.13 Visual Studio Code

Visual Studio Code adalah aplikasi lintas platform yang tersedia untuk beberapa sistem operasi seperti *Windows*, *Linux*, dan *Mac Os*. Visual Studio Code menyertakan editor kode sumber yang ringan dan kuat di desktop. Penggunaan berbagai bahasa pemrograman seperti *Java*, *JavaScript*, *Go*, *C++*. Komponen Visual Studio Code identic dengan yang digunakan di *Azure DevOps*.

Visual Studio memiliki editor kode lintas platform sederhana yang dapat digunakan siapa saja untuk menulis atau membangun aplikasi web.

Komponen pada *Visual Studio Code* :

- a. *customize* : digunakan untuk menambahkan ekstensi bahasa pemrograman. Tambahkan ekstensi bahasa pemrograman untuk mengkonfigurasi dan menyesuaikan template sehingga tidak harus mengingat fitur bahasa.
- b. *command palette* : command palette menyediakan banyak akses perintah, kita bisa memberikan perintah editor membuka file, mencari file, dan sebagainya dengan cepat dan mudah. Untuk membuka command palette bisa dengan tekan Ctrl+Shift+P.
- c. *integrated terminal* : integrated terminal digunakan untuk mengeksekusi skrip di editor. Kita bisa mengeksekusi skrip editor secara langsung di terminal tanpa harus membuka terminal tambahan, ini adalah salah satu kelebihan dari visual studio code.
- d. *extention* : extention adalah fungsi tambahan yang dirancang untuk memperluas fungsional editor dan membantu pengembang dalam pemrograman.
- e. *search* : fitur pencarian visual studio code juga sangat cepat, dan selain kecepatan pencarian data, fitur ini memiliki keuntungan untuk dapat mencari hingga ke tingkat konten.

- f. *grid editor layout* : juga mudah dalam manajemen tata letak visual studio code, setiap tata letak memungkinkan mengatur grup editor secara vertical atau horizontal.
- g. *color themes* : color themes digunakan untuk mengatur warna antarmuka visual studio code sesuai keinginan dengan memilih *File>Prefences>Color Themes* dan memilih tema pilihan dengan menyeret kursor ke atas atau ke bawah.
- h. *cloud enviroment* : kita juga bisa melakukan sesuatu di lingkungan cloud melalui visual studio code seperti membuat *database*, menjalankan perintah, *insert*, *update*, *delete*, dan sebagainya di *cloud*.

#### 2.2.6 Xampp MYSQL

Xampp MySQL adalah program perangkat lunak gratis yang mendukung beberapa sistem operasi, merupakan gabungan dari beberapa program server-independen (localhost), yang terdiri dari program *database* MySQL, server HTTP Apache, dan compiler yang ditulis dalam bahasa pemrograman PHP dan Perl. ditulis dalam bahasa yang tugasnya menjalankan website berbasis PHP dengan program database berbasis MySQL di komputer lokal. Xampp juga dikenal sebagai server Cpanel yang dapat menampilkan situs web tanpa internet atau koneksi internet. [6]

Xampp adalah singkatan di mana setiap hurufnya adalah :

X : Program ini dapat berjalan di banyak sistem operasi seperti *Windows, Linux, Mac OS* dan *Solaris*.

A : Apache adalah aplikasi server web. Tugas utama Apache adalah membuat halaman web yang tepat untuk pengguna berdasarkan kode PHP yang ditulis oleh penulis situs web, jika perlu juga berdasarkan kode PHP yang ditulis, setelah itu dukungan database (misalnya di Mysql) situs web yang dihasilkan dapat mendukung.

M : MySQL adalah aplikasi server basis data. Pengembangannya disebut SQL, yang merupakan singkatan dari *Structured Query Language*. SQL adalah bahasa terstruktur untuk menangani database. MySQL dapat digunakan untuk membuat dan mengelola database dan isinya. MySQL dapat digunakan untuk menambah, memodifikasi dan menghapus data dari *database*.

P : PHP, bahasa pemrograman web. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat web yang bersifat *server-side scripting*. PHP memungkinkan pembuatan situs web dinamis. Sistem manajemen basis data yang biasa digunakan dengan PHP adalah MySQL, tetapi PHP juga mendukung sistem manajemen *database Oracle, Microsoft Acces, Interbase, D-base, PostgreSQL*, dan sebagainya.

P : Perl, bahasa pemrograman tujuan utama, pertama kali dikembangkan oleh Larry Wall di mesin *Unix*. Perl pertama kali dirilis pada tanggal 8 Desember 1987 ditandai dengan keluarnya Perl



1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian *Unix* (*SunOs, Linux, BSD, HP-UX*), juga tersedia untuk sistem operasi seperti *DOS, Windows, PowerPC, BeOS, VMS, EBCDIC, dan PocketPC*.

## **BAB III**

### **METODOLOGI PENELITIAN**

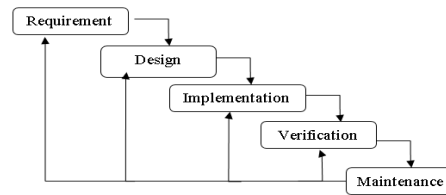
#### **3.1 Alat Penelitian**

Alat dan perangkat lunak yang dibutuhkan pada penelitian ini, antara lain :

- a. *Visual Studio Code*
- b. *Xampp*
- c. *Database Mysql*
- d. *Codeigniter 3*
- e. *Javascript*
- f. *Css*
- g. *Html*
- h. *Apache webserver*

#### **3.2 Prosedur Penelitian**

Metode penelitian yang digunakan dalam penelitian ini adalah metode *SDLC (waterfall)*. *Waterfall* adalah metode kerja yang menekankan fase berurutan dan sistematis. Disebut air terjun karena prosesnya mengalir "ke bawah" dalam satu arah seperti air terjun. Metode waterfall ini harus dilakukan secara berurutan sesuai tahapan yang ada. [10]



Gambar 3. 1 Prosedur Penelitian

### 1. Requirements

Mengumpulkan kebutuhan secara lengkap untuk dianalisis dan mendefinisikan kebutuhan apa saja yang harus dicapai oleh program. Informasi dapat diperoleh melalui wawancara, diskusi, atau survey.

### 2. Design

Melakukan perancangan desain perangkat lunak sebagai perkiraan sebelum dibuatnya kode. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi *system* dasar perangkat lunak dan hubungannya.

### 3. Implementasi

Implementasi ini merupakan tahap dimana semua desain yang telah dilakukan sebelumnya diubah menjadi kode program. Kode yang dihasilkan masih berupa modul-modul yang harus digabungkan pada tahap selanjutnya.

### 4. Verification

Penggabungan modul-modul yang sudah dibuat sebelumnya dilakukan di tahap ini dan melakukan pengujian untuk mengetahui

baik perangkat lunak yang dibuat telah sesuai dengan desain dan fungsinya atau tidak.

## **5. Maintenance**

Langkah ini merupakan langkah terakhir dari model waterfall. Sistem telah dijalankan dan diuji. Tidak ada kesalahan yang ditemukan pada langkah sebelumnya.

### **3.3 Metode Pengumpulan Data**

#### **1. Observasi**

Penelitian lapangan (*observasi*) adalah teknik pengumpulan data yang melibatkan terjun langsung ke lapangan untuk mengamati secara langsung masalah, Tingkah laku, benda yang dilihat, dan hal-hal lain yang diperlukan untuk mendukung berlangsungnya penyidikan di tempat. Pada penelitian ini peneliti melakukan observasi langsung pada tempat-tempat yang dianggap perlu untuk penelitian ini, seperti mengunjungi lokasi khususnya di desa Brebes Luwunragi.

#### **2. Wawancara**

Wawancara adalah teknik pengumpulan data yang dilakukan secara tatap muka dan melalui tanya jawab langsung antara peneliti dengan sumber data. Sumber informasi peneliti adalah ahli pengolahan air.

#### **3. Studi Pustaka**

Dalam metode ini pengumpulan data dilakukan dengan cara meneliti buku-buku referensi, termasuk literatur tertulis, dan topik-topik

yang mendukung pembuatan sistem web. Selain itu, menggunakan internet sebagai sumber informasi untuk menambah pengetahuan tentang proses produksi.

### 3.4 Tempat dan Waktu Pelaksanaan

#### 1. Tempat Penelitian

Tempat pelaksanaan penelitian adalah di kediaman bapak Ahmad Jaeni di Jln. K.H Syarkawi desa Luwungragi Brebes.



Gambar 3. 2 Maps Penelitian

#### 2. Waktu Penelitian

Waktu penelitian ini dilaksanakan sejak bulan Maret 2023. Pengumpulan data dan pengolahan data meliputi penyajian dalam bentuk laporan dan proses bimbingan.

## **BAB IV**

### **ANALISA DAN PERANCANGAN SISTEM**

#### **4.1 Analisa Permasalahan**

Analisis permasalahan adalah untuk mengontrol kecepatan debit air dan mendeteksi apakah pipa air bocor. Memantau kecepatan debit air dan mendeteksi kebocoran secara manual adalah pekerjaan yang sulit, jadi solusi yang efektif adalah menggunakan teknologi IoT untuk pemantauan otomatis. Di Desa Luwungragi, metode yang digunakan untuk mendeteksi kebocoran pada jaringan distribusi air umumnya masih dilakukan secara manual, baik secara langsung ke wilayah distribusi air untuk diperiksa atau berdasarkan laporan masyarakat tentang kebocoran jaringan distribusi air. Efek pendeteksian metode ini buruk, karena butuh waktu lama untuk mendeteksi kebocoran pipa jaringan distribusi. Oleh karena itu, pada penelitian ini dibuat sebuah sistem monitoring berbasis web untuk memonitor laju debit air dan mendeteksi kebocoran pipa air.

Berdasarkan hal tersebut maka dapat dipecahkan suatu permasalahan yaitu bagaimana membuat sistem pengukuran kecepatan debit air dan sistem deteksi kebocoran pipa air, sehingga memudahkan petugas dalam melakukan pengecekan kecepatan debit air dan kebocoran pipa air.

## 4.2 Analisa Kebutuhan Sistem

Perangkat keras (*Hardware*) dan perangkat lunak (*Software*) yang diperlukan untuk membangun alat pemantauan kualitas air dan pendeteksi kebocoran pipa air berbasis *Internet Of Things* sebagai berikut :

### 4.2.1 Perangkat Lunak (*Software*)

Perangkat lunak diprogram secara digital dan menyimpan data yang tidak terlihat secara fisik tetapi berada di dalam komputer. *Software* atau perangkat lunak dapat berupa program atau menjalankan suatu perintah atau perintah yang memungkinkan perangkat lunak komputer untuk menjalankan atau mengeksekusi suatu perintah. Dapat dikatakan perangkat lunak bekerja didalam perangkat keras. *Software* yang digunakan dalam pembuatan alat monitoring kualitas air dan pendeteksi kebocoran pipa air berbasis *Internet Of Things* berintegasi *website* adalah :

- a. *Visual Studio Code*
- b. *Xampp*
- c. *Astah (UML)*

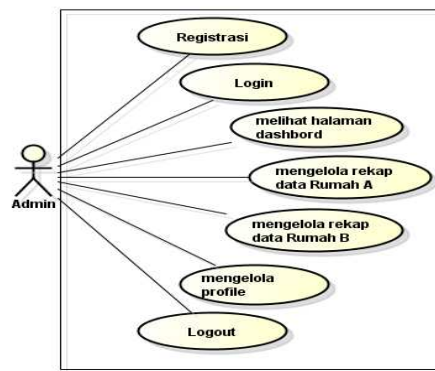
## 4.3 Perancangan Sistem

### 4.3.1 Perancangan Perangkat Lunak

Perancangan perangkat lunak merupakan rancangan dari *website* yang digunakan untuk memonitoring kualitas air dan pendeteksi kebocoran pipa air. Perancangan yang digunakan sebagai berikut :

#### 4. Use Case Diagram

Merupakan gambaran umum dari sistem yang dibuat. *Use Case Diagram* menggambarkan interaksi antara aktor dan sistem. *Use Case Diagram* menunjukkan apa yang biasanya dilakukan dengan sistem dan siapa yang bisa mengakses sistem tersebut.



Gambar 4. 1 Use Case Diagram

Pada gambar 4.1 terdapat admin sebagai aktor yang dapat melakukan registrer (apabila belum mempunyai akun), melakukan login, dapat mengakses halaman dashboard, rekap data rumah A, rekap data rumah B, dan *profile*.

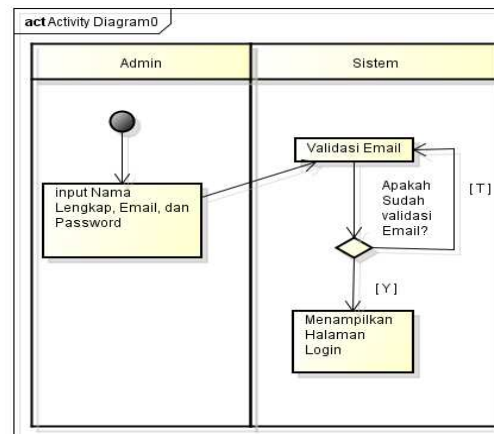
#### 5. Activity Diagram

*Activity Diagram* menjelaskan berbagai alur aktifitas yang ada dalam sistem yang telah dirancang, yang digunakan untuk menggambarkan fungsi yang dibentuk dalam suatu rancangan seperti pada *use case*. *Activity diagram* juga



menjelaskan bagaimana alur berawal dan *decision* yang akan terjadi, selain itu *activity diagram* menjelaskan proses *parallel* yang akan terjadi pada beberapa eksekusi yang telah dibuat.

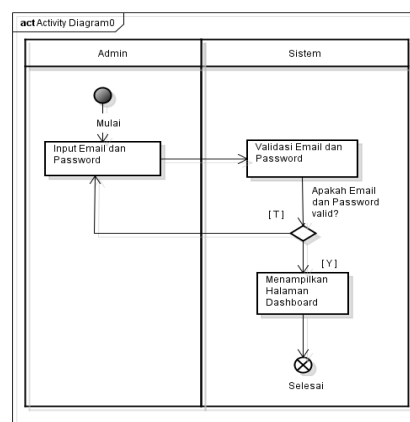
a. Activity Registrasi



Gambar 4. 2 Activity Registrasi

Admin pada gambar 4.2 membuat akun dengan memasukkan nama lengkap, Email, dan Password. Selanjutnya akan ada notifikasi Email untuk di validasi.

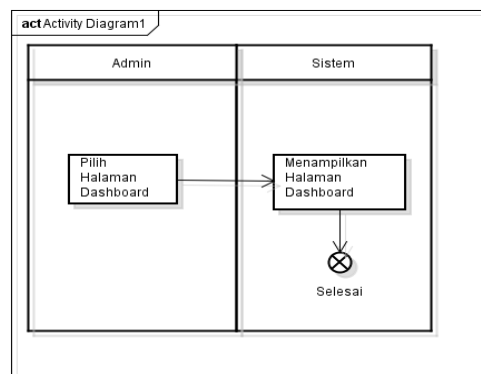
b. Activity Login



Gambar 4. 3 Activity Login

Admin pada gambar 4.3 dapat mengisi *Email* dan *Password* kemudian setelah klik *login*, sistem akan melakukan validasi *Email* dan *Password*, apabila benar maka sistem menampilkan dashboard, apabila ada kesalahan input *Email* dan *Password* maka admin harus input kembali *Email* dan *Password* yang benar.

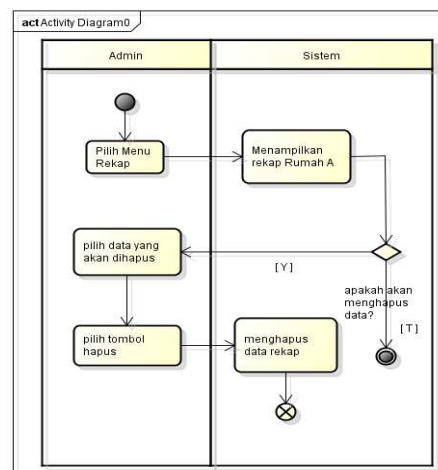
c. *Activity* halaman *Dashboard*



Gambar 4. 4 Activity Dashboard

Pada gambar 4.4 setelah admin berhasil *login* sistem akan menampilkan halaman *dashboard*.

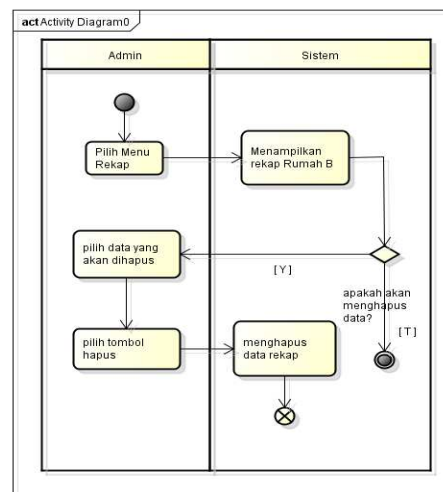
d. *Activity* rekap data rumah A



Gambar 4. 5 Activity rekap data rumah A

Pada gambar 4.5 setelah *login* admin dapat memilih bagian menu rekap data rumah A maka sistem akan menampilkan halaman rekap data debit air dan informasi kebocoran pada pipa. Kemudian jika admin ingin menghapus data, pilih data lalu klik hapus maka sistem akan menghapus data yang dipilih.

e. *Activity* rekap data rumah B

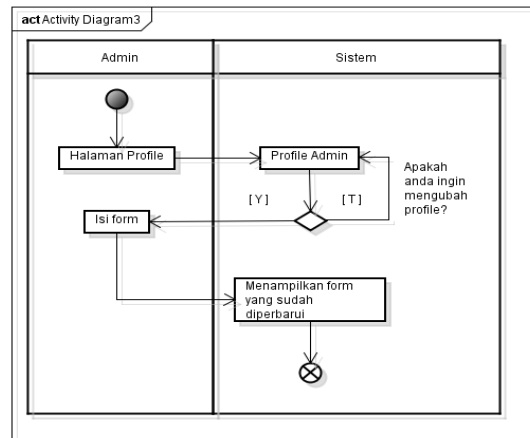


Gambar 4. 6 Activity rekap data rumah B

Pada gambar 4.6 setelah *login* admin dapat memilih bagian menu rekap data rumah B maka sistem akan menampilkan halaman rekap data debit air dan informasi kebocoran pada pipa. Kemudian jika admin ingin menghapus data, pilih data lalu klik

hapus maka sistem akan menghapus data yang dipilih.

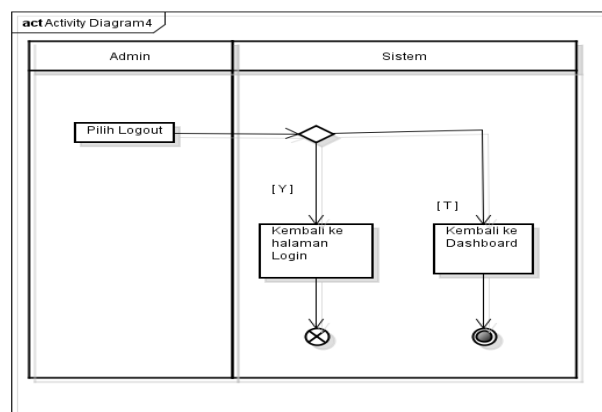
f. *Activity profile admin*



Gambar 4. 7 Activity Profile Admin

Pada gambar 4.7 setelah *login* admin dapat memilih bagian menu *profile*. Kemudian sistem akan menampilkan halaman *profile* dan memberikan pilihan akan mengedit *profile* atau tidak. Jika yam aka admin pilih edit *profile* lalu sistem menampilkan form edit *profile*, kemudian admin klik *submit*.

g. *Activity logout*



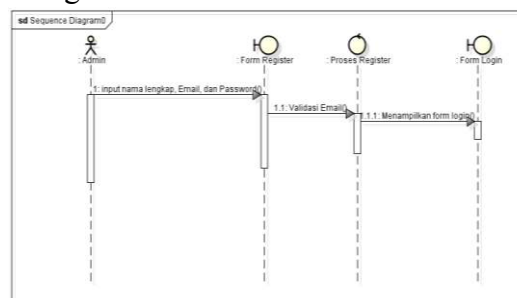
Gambar 4. 8 Activity Logout

Pada gambar 4.8 sebagai admin ketika klik *logout* sistem akan melakukan proses *logout* lalu kembali ke tampilan *login*. Jika tidak *logout* sistem akan kembali kehalaman *dashboard*.

## 6. Sequence Diagram

*Sequence diagram* menjelaskan mengenai interaksi antar objek didalam dan sekitar sistem. *Sequence diagram* terdiri antara dimensi *vertical* (waktu) dan dimensi *horizontal* (objek-objek yang terkait) pada perancangan sistem monitoring kualitas air dan pendeteksi kebocoran pipa air berbasis *Internet Of Things*. Beberapa *sequence diagram* dibuat untuk mempermudah perancangan sehingga dapat diketahui interaksi admin dan sistem dalam pengolahan data sistem monitoring kualitas air dan pendeteksi kebocoran pipa air berbasis *Internet Of Things* seperti berikut :

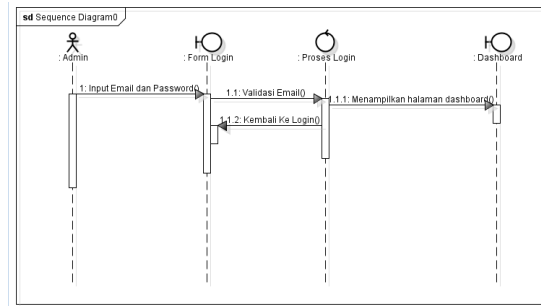
### a. Sequence registrasi



Gambar 4. 9 Sequence Registrasi

Gambar 4.9 membuat akun dengan memasukkan nama lengkap, *Email*, dan *Password*. Selanjutnya akan ada notifikasi Email untuk di validasi.

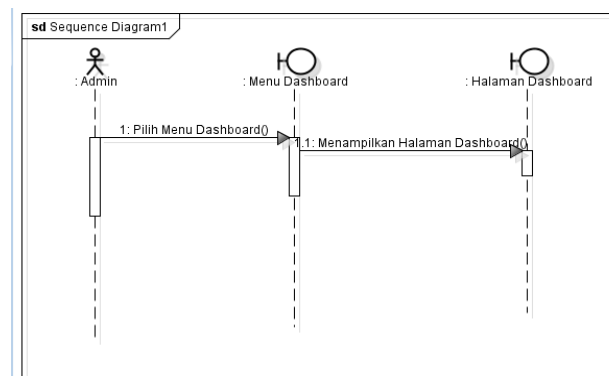
b. *Sequence login*



Gambar 4. 10 Sequence Login

Pada gambar 4.10 dapat mengisi *Email* dan *Password* kemudian setelah klik *login*, sistem akan melakukan validasi *Email* dan *Password*, apabila benar maka sistem menampilkan dashboard, apabila ada kesalahan input *Email* dan *Password* maka admin harus input kembali *Email* dan *Password* yang benar.

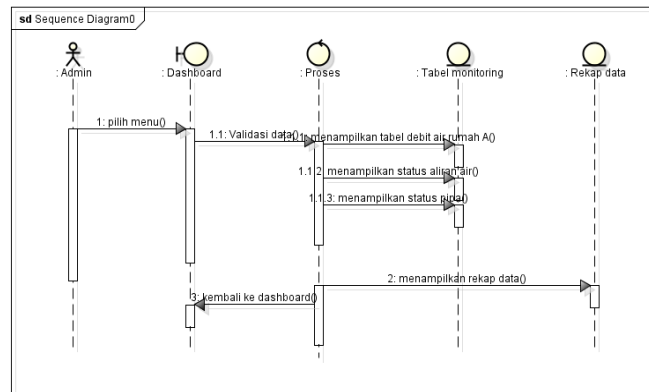
c. *Sequence halaman dashboard*



Gambar 4. 11 Sequence Dashboard

Pada gambar 4.11 setelah admin berhasil *login* sistem akan menampilkan halaman *dashboard*.

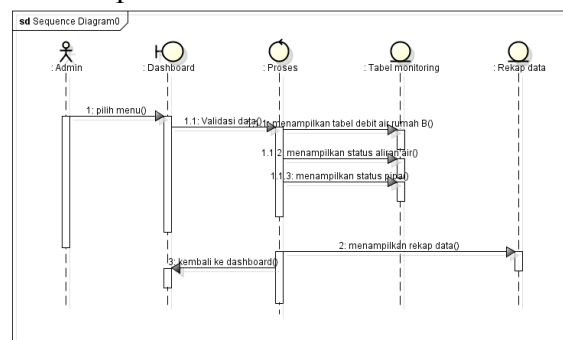
c. *Sequence* rekap data rumah A



Gambar 4. 12 *Sequence* rekap data rumah A

Pada gambar 4. 12 setelah *login* admin memasuki halaman *dashboard* lalu validasi data setelah itu masuk kebagian proses setelah proses akan memasuki monitoring tabel rekap data dari proses, sistem dapat menampilkan halaman rekap data rumah A.

d. *Sequence* rekap data rumah B

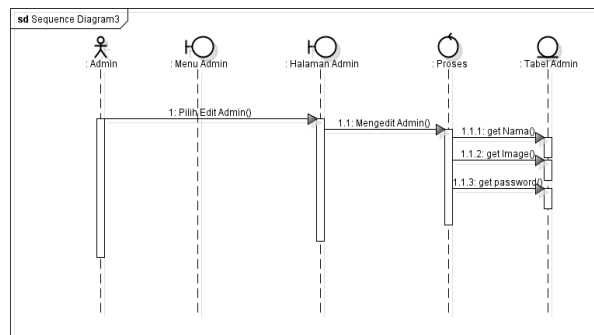


Gambar 4. 13 *Sequence* rekap data rumah B

Pada gambar 4. 13 setelah *login* admin memasuki halaman *dashboard* lalu validasi data setelah itu masuk kebagian proses setelah proses akan memasuki monitoring tabel rekap data dari proses, sistem dapat menampilkan halaman rekap data rumah B.

monitoring tabel rekap data dari proses, sistem dapat menampilkan halaman rekap data rumah B

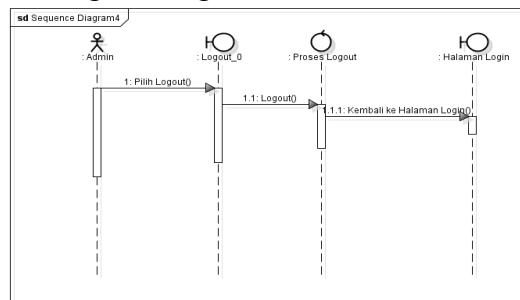
e. *Sequence Profile admin*



Gambar 4. 14 Sequence Profile Admin

Pada gambar 4.14 setelah *login* admin dapat memilih bagian menu *profile*. Kemudian sistem akan menampilkan halaman *profile* dan memberikan pilihan akan mengedit *profile* atau tidak. Jika yam aka admin pilih edit *profile* lalu sistem menampilkan form edit *profile*, kemudian admin klik submit.

f. *Sequence diagram Logout*



Gambar 4. 15 Squence Logout

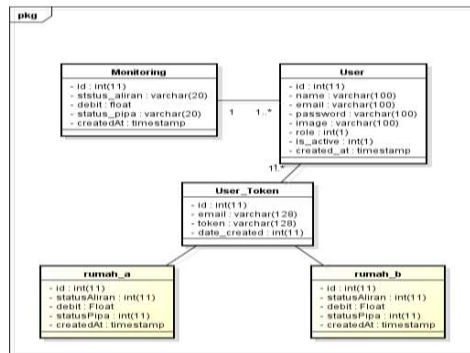
Pada gambar 4.15 sebagai admin ketika klik *logout* sistem akan melakukan proses *logout* lalu



kembali ke tampilan *login*. Jika tidak *logout* sistem akan kembali kehalaman *dashboard*.

#### 4. Class Diagram

Class diagram digunakan untuk hubungan antar *class*, atribut-atribut dan operasi-operasi yang ada. Perancangan *class* diagram dari sistem monitoring kualitas air dan pendeteksi kebocoran pipa air dapat dilihat sebagai berikut:



Gambar 4. 16 Class Diagram

*Class* diagram menggambarkan pendeskripsian atau gambaran dari *class*, operasi serta hubungan antar objek.

#### 4.3.2 Perancangan Database

Dalam perancangan database sistem monitoring kualitas air dan pendeteksi kebocoran pipa air. Ada 3 (tiga) hal yang akan digunakan dalam perancangan ini, database dengan nama *db\_monitoring* dan tabelnya yaitu “*monitoring*”, “*user*” dan “*token user*”. Berikut penjelasan semua tabel dari database “*db\_monitoring*”.

## 1. Tabel monitoring

Nama Database : db\_monitoring

Nama Tabel : monitoring

Tabel 4. 1 Tabel Monitoring

<b>Field</b>	<b>Type</b>	<b>Keterangan</b>
Id	Int(11)	Auto_Increment
Status_aliran	Varchar(20)	
Debit	Float	
Status_pipa	Varchar(20)	
createdAt	timestamp	

## 2. Tabel rumah A

Tabel 4. 2 Tabel rumah A

<b>Field</b>	<b>Type</b>	<b>Keterangan</b>
Id	Int(11)	
StatusAliran	Int(11)	
Debit	Float	
statusPipa	Int(11)	
createdAt	timestamp	

## 3. Tabel rumah B

Tabel 4. 3 Tabel rumah B

<b>Field</b>	<b>Type</b>	<b>Keterangan</b>
Id	Int(11)	
StatusAliran	Int(11)	

<b>Field</b>	<b>Type</b>	<b>Keterangan</b>
Debit	Float	
statusPipa	Int(11)	
createdAt	timestamp	

## 4. Tabel user

Tabel 4. 4 Tabel User

<b>Field</b>	<b>Type</b>	<b>Keterangan</b>
Id	Int(11)	Primary key
Name	Varchar(100)	
Email	Varchar(100)	
Password	Varchar(100)	
Image	Varchar(100)	
role	Int(1)	
Is_active	Int(1)	
Created_at	timestamp	Current_timestamp()

## 5. Tabel token\_user

Tabel 4. 5 Tabel Token User

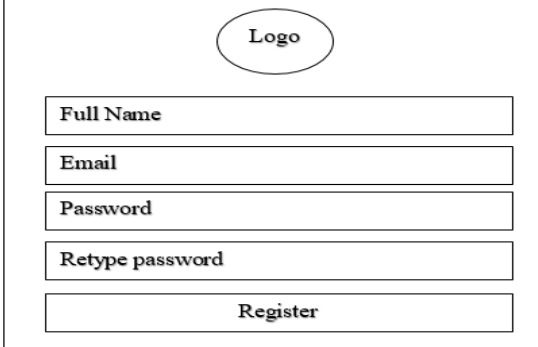
<b>Field</b>	<b>Type</b>	<b>Keterangan</b>
Id	Int(11)	Primary key
Email	Varchar(128)	
Token	Varchar(128)	
Date_created	Int(11)	

#### 4.4 Desain Input/Output

Berikut ini adalah perancangan dari perangkat lunak untuk *website* monitoring kualitas air dan pendeteksi kebocoran pipa didesain menggunakan bahasa pemrograman PHP dan aplikasi Visual Studio Code.

##### 1. Desain *View Register dan Login*

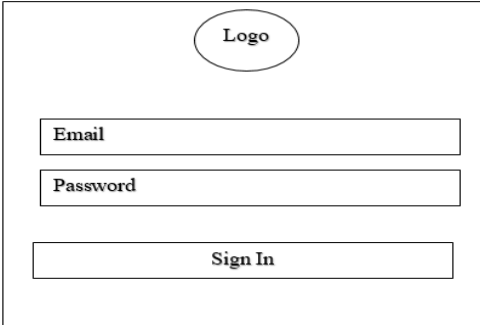
Pada tampilan awala, admin akan melakukan registrasi akun selanjutnya akan memasukkan Full Name, *Email* dan Password.



The diagram shows a registration form layout within a rectangular border. At the top center is an oval labeled "Logo". Below it are four stacked rectangular input fields, each with a label: "Full Name", "Email", "Password", and "Retype password". At the bottom of the form is a wide rectangular button labeled "Register".

Gambar 4. 17 Halaman Registrasi akun

Pada gambar 4.17 merupakan halaman registrasi atau daftar akun dengan memasukkan full name, *email* dan juga *password*.

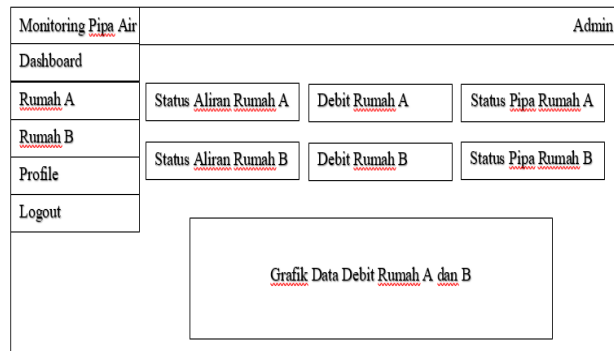


The diagram shows a login form layout within a rectangular border. At the top center is an oval labeled "Logo". Below it are two stacked rectangular input fields, each with a label: "Email" and "Password". At the bottom of the form is a wide rectangular button labeled "Sign In".

Gambar 4. 18 Halaman Login

Pada gambar 4.18 merupakan halaman *Login* akun dengan memasukkan *email* dan *password*.

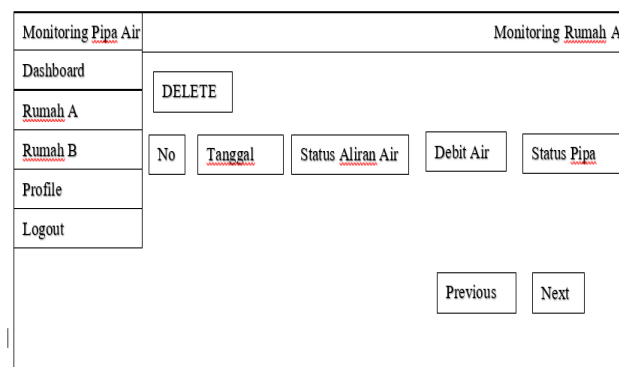
## 2. Desain *View* Dashboard



Gambar 4. 19 Halaman Dashboard

Pada gambar 4.19 merupakan rancangan desain untuk halaman dashboard yang menampilkan nilai debit rumah A dan B, status aliran air dan status pipa air.

## 3. Desain halaman rekap data rumah A



Gambar 4. 20 Desain halaman rekap data rumah A

Pada gambar 4.20 merupakan halaman rekap data rumah A yang dapat menampilkan hasil pengujian pada alat yang berisi tampilan tanggal pengujian, status aliran air, debit air, dan status pipa.

#### 4. Desain halaman rekap data rumah B

Monitoring Pipa Air	Monitoring Rumah B									
Dashboard	<div style="text-align: center;"> <input type="button" value="DELETE"/> </div> <table border="1" style="width: 100%; text-align: center;"> <tr> <td>No</td> <td>Tanggal</td> <td>Status Aliran Air</td> <td>Debit Air</td> <td>Status Pipa</td> </tr> </table> <div style="text-align: right; margin-top: 20px;"> <input type="button" value="Previous"/> <input type="button" value="Next"/> </div>					No	Tanggal	Status Aliran Air	Debit Air	Status Pipa
No						Tanggal	Status Aliran Air	Debit Air	Status Pipa	
Rumah A										
Rumah B										
Profile										
Logout										

Gambar 4. 21 Desain halaman rekap data rumah B

Pada gambar 4.21 merupakan halaman rekap data rumah B yang dapat menampilkan hasil pengujian pada alat yang berisi tampilan tanggal pengujian, status aliran air, debit air, dan status pipa.

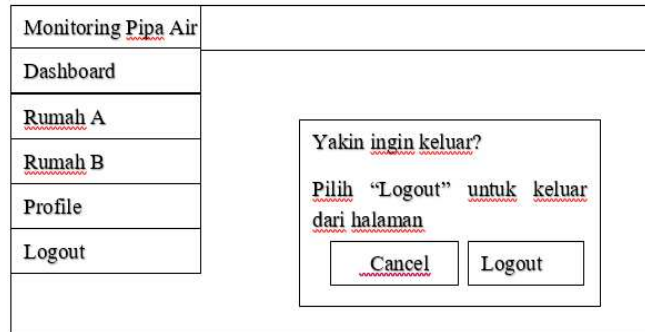
#### 5. Desain *View Profile*

Monitoring Pipa Air			
Dashboard	<div style="text-align: center;"> <input type="button" value="Profile admin"/> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 30%;"> <div style="text-align: center; margin-bottom: 10px;"> </div> <p><u>Name :</u> <input type="text"/></p> <p><u>Email :</u> <input type="text"/></p> </div> <div style="width: 30%; border: 1px solid black; padding: 5px;"> <p><b>Change Profile</b></p> <p><u>Pilih foto</u> <input type="text"/></p> <p><u>Ganti nama</u> <input type="text"/></p> <p style="text-align: center;"><input type="button" value="submit"/></p> </div> <div style="width: 30%; border: 1px solid black; padding: 5px;"> <p><b>Ganti password</b></p> <p><u>Current password</u> <input type="text"/></p> <p><u>New password</u> <input type="text"/></p> <p><u>retype new password</u> <input type="text"/></p> <p style="text-align: center;"><input type="button" value="submit"/></p> </div> </div>		
Rumah A			
Rumah B			
Profile			
Logout			

Gambar 4. 22 Halaman Profile Admin

Pada gambar 4.22 merupakan halaman *profile* yang dapat mengedit *profile admin*, menambahkan foto admin, dan mengubah *password* pada akun.

## 6. Desain *View Logout*



Gambar 4. 23 Halaman Logout

Pada gambar 4.23 merupakan tampilan informasi *logout*, pada saat kita mengklik "*Logout*" maka akan kembali ke tampilan *login*.

## **BAB V**

### **HASIL DAN PEMBAHASAN**

#### **5.1 Implementasi System**

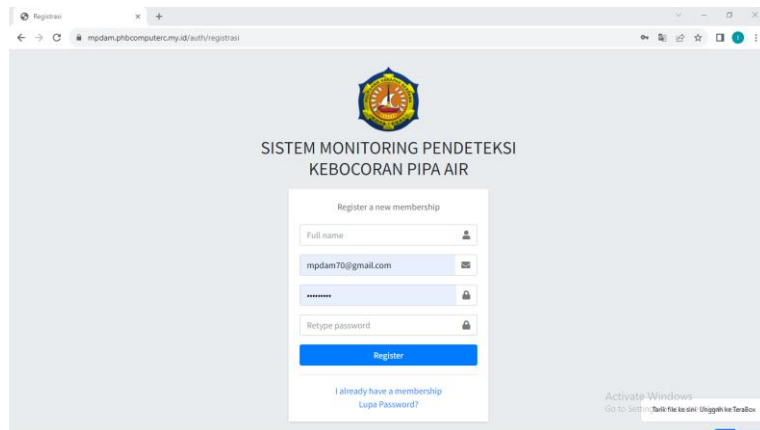
Setelah tahap analisis dan perancangan, langkah selanjutnya dalam membangun sistem informasi adalah menguji apakah sistem sudah siap untuk diimplementasikan. Tahap implementasi sistem adalah kegiatan untuk memperoleh sistem kerja. Selama tahap implementasi sistem dilakukan beberapa kegiatan antara lain: pengujian sistem, pengujian program.

##### **5.1.1 Implementasi Perangkat Lunak**

Implementasi perangkat lunak merupakan penerapan *Website* sebagai media Sistem Monitoring Pendeteksi Kebocoran Pipa Air. Sistem dikembangkan dan diimplementasikan menggunakan bahasa pemrograman PHP (*Hypertext Preprocessor*) dan HTML (*Hypertext Markup Language*) dan database *phpMyAdmin* (perangkat lunak bebas yang ditulis dengan bahasa pemrograman PHP untuk menangani administrasi *MySQL* melalui *website*).



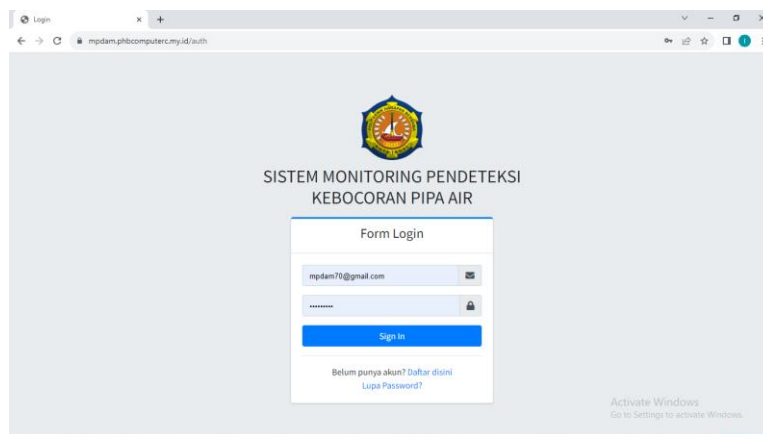
### a. Halaman Registrasi



Gambar 5. 1 Halaman Registrasi

Gambar 5.1 merupakan halaman untuk membuat akun apabila belum memiliki akun dengan menginput nama lengkap, *Email*, dan *Password* selanjutnya akan memvalidasi *Email* melalui notifikasi *Email*.

### b. Halaman Login

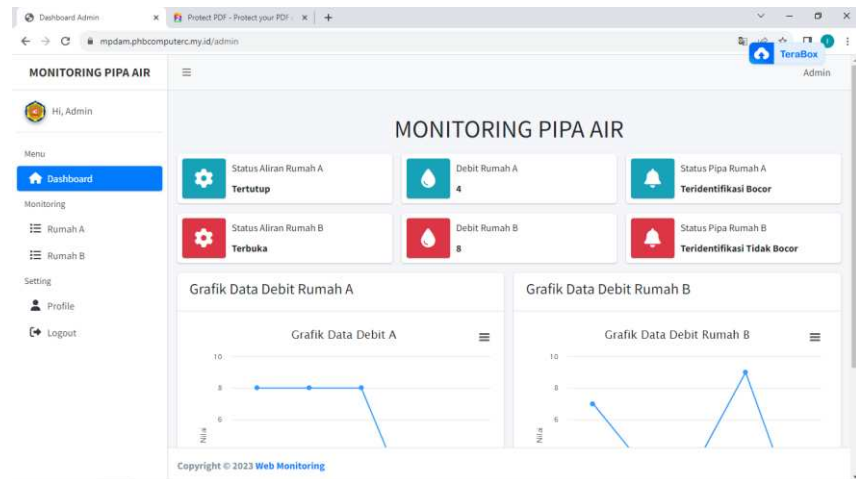


Gambar 5. 2 Halaman Login

Gambar 5.2 merupakan tampilan *login* *website* monitoring pendeteksi kebocoran. Ketika

admin mengakses link dari *website* ini maka akan dilakukan *login* terlebih dahulu.

### c. Halaman Dashboard



Gambar 5. 3 Halaman Dasboard

Gambar 5.3 merupakan tampilan halaman dashboard, halaman ini menampilkan debit air rumah A dan B, status aliran air, status pipa, dan grafik.

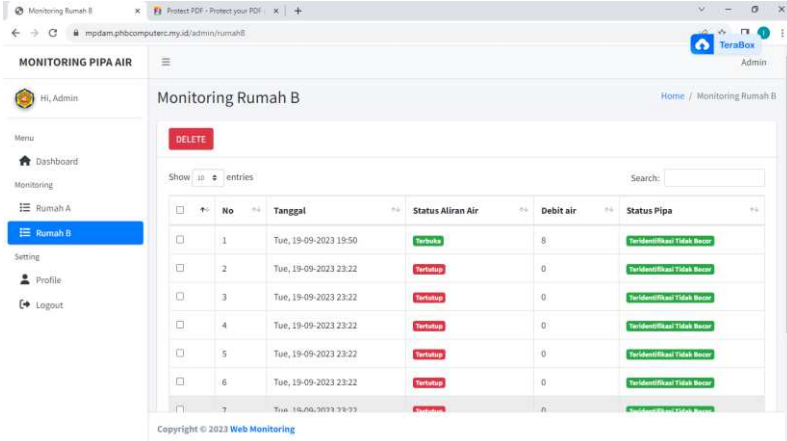
### d. Halaman Rekap Data rumah A

No	Tanggal	Status Aliran Air	Debit air	Status Pipa
1	Thu, 21-09-2023 14:55	Tertutup	4	Teridentifikasi Tidak Bocor
2	Thu, 21-09-2023 14:56	Terbuka	0	Teridentifikasi Tidak Bocor
3	Thu, 21-09-2023 14:56	Terbuka	8	Teridentifikasi Tidak Bocor
4	Thu, 21-09-2023 14:56	Terbuka	8	Teridentifikasi Tidak Bocor
5	Thu, 21-09-2023 14:56	Terbuka	8	Teridentifikasi Tidak Bocor

Gambar 5. 4 Halaman rekap data rumah A

Gambar 5.4 merupakan tampilan halaman rekap data rumah A.

e. Halaman rekap data rumah B



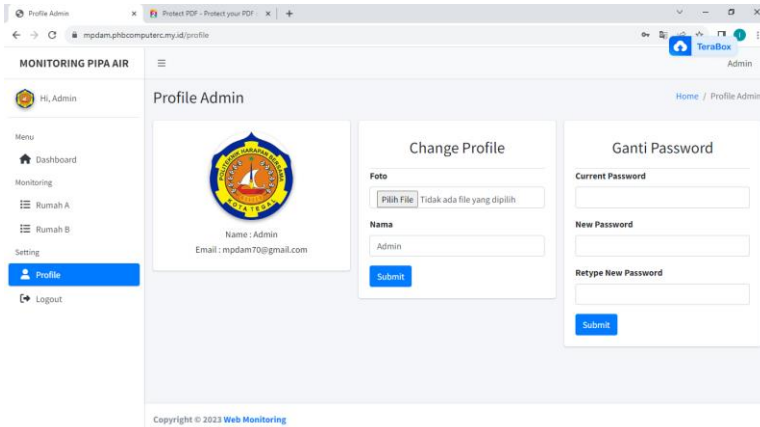
The screenshot shows the 'Monitoring Rumah B' page with a table of data. The table has columns for 'No', 'Tanggal', 'Status Aliran Air', 'Debit air', and 'Status Pipa'. The data rows are as follows:

No	Tanggal	Status Aliran Air	Debit air	Status Pipa
1	Tue, 19-09-2023 19:50	Tertutup	8	Terdeteksi Tidak Bocor
2	Tue, 19-09-2023 23:22	Tertutup	0	Terdeteksi Tidak Bocor
3	Tue, 19-09-2023 23:22	Tertutup	0	Terdeteksi Tidak Bocor
4	Tue, 19-09-2023 23:22	Tertutup	0	Terdeteksi Tidak Bocor
5	Tue, 19-09-2023 23:22	Tertutup	0	Terdeteksi Tidak Bocor
6	Tue, 19-09-2023 23:22	Tertutup	0	Terdeteksi Tidak Bocor
7	Tue, 19-09-2023 23:22	Tertutup	0	Terdeteksi Tidak Bocor

Gambar 5. 5 halaman rekap data rumah B

Gambar 5.5 merupakan tampilan halaman rekap data rumah B.

f. Halaman Profile Admin



The screenshot shows the 'Profile Admin' page. It features a user profile card with a logo, name 'Admin', and email 'mpdam70@gmail.com'. To the right, there are two main sections: 'Change Profile' and 'Ganti Password'. The 'Change Profile' section includes a 'Foto' field with a 'Pilih File' button and a 'Nama' field with the value 'Admin'. The 'Ganti Password' section includes fields for 'Current Password', 'New Password', and 'Retype New Password', each with a 'Submit' button.

Gambar 5. 6 Halaman Profile Admin

Gambar 5.6 merupakan tampilan halaman *profile*, pada halaman ini untuk mengedit profile admin.

## 5.2 Hasil Pengujian

### 5.2.1 Pengujian Sistem

Pengujian sistem ini menggunakan *black box testing* merupakan pengujian yang dilakukan untuk mengamati hasil input dan output dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak. Pengujian sistem informasi ini dimaksudkan untuk menguji apakah seluruh bagian website berfungsi sesuai dengan yang diharapkan.

### 5.2.2 Rencana Pengujian

Program pengujian sistem akan dilakukan secara bertahap, dimulai dengan pengujian kecepatan debit air pada rumah A dan B, pengujian status aliran air, status pipa air apakah ada informasi kebocoran yang akan ditampilkan pada tampilan *website*.

### 5.2.3 Pengujian

Hasil pengujian dari tabel berikut:

1. Hasil pengujian *Login*

Tujuan dari pengujian *login* adalah untuk memastikan bahwa *Email* dan *Password* yang dimasukkan sudah benar, sehingga website dapat diakses dengan baik.

## a. Hasil Pengujian Data True

Tabel 5. 1 Hasil Pengujian Login

<b>Skenario Pengujian</b>	<b>Test Case</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengujian</b>
Menginput data <i>login</i> dengan benar, lalu mengklik <i>login</i>	<i>Email</i> dan <i>Password</i> sesuai <i>role_id</i>	Sistem menerima akses <i>login</i> dan kemudian masuk halaman sesuai <i>role_id</i> yang diharapkan	Valid

## b. Hasil Pengujian Data False

Tabel 5. 2 Hasil Pengujian Login Salah

<b>Skenario Pengujian</b>	<b>Test Case</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengujian</b>
Mengkosongkan semua isian data <i>login</i> , lalu langsung mengklik <i>login</i>	<i>Email</i> : kosong <i>Password</i> : kosong	Sistem akan menolak akses <i>login</i> dan menampilkan pesan " <i>please fill in your Email atau Password</i> "	Valid

<b>Skenario Pengujian</b>	<b>Test Case</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengujian</b>
Hanya mengisi data <i>email</i> dan mengkosongkan <i>password</i> , lalu mengklik login.	<i>Email</i> : @gmail. com <i>Password</i> d : kosong	Sistem akan menolak akses <i>login</i> dan menampilkan pesan “ <i>please fill in your Email atau Password</i> ”	Valid
Hanya mengisi data <i>password</i> dan mengkosongkan <i>email</i> , lalu mengklik login.	<i>Email</i> : kosong <i>Password</i> d : 1234	Sistem akan menolak akses <i>login</i> dan menampilkan pesan “ <i>please fill in your Email atau Password</i> ”	Valid

## 2. Hasil Pengujian Rekap Data

Admin yang berhasil masuk dengan Email dan Password yang sesuai akan dialihkan ke halaman berikutnya. Halaman ini berisi tentang data kecepatan debit air, status aliran air dan adanya kebocoran atau tidak pada pipa yang didapat dari sensor dan data yang terkirim.

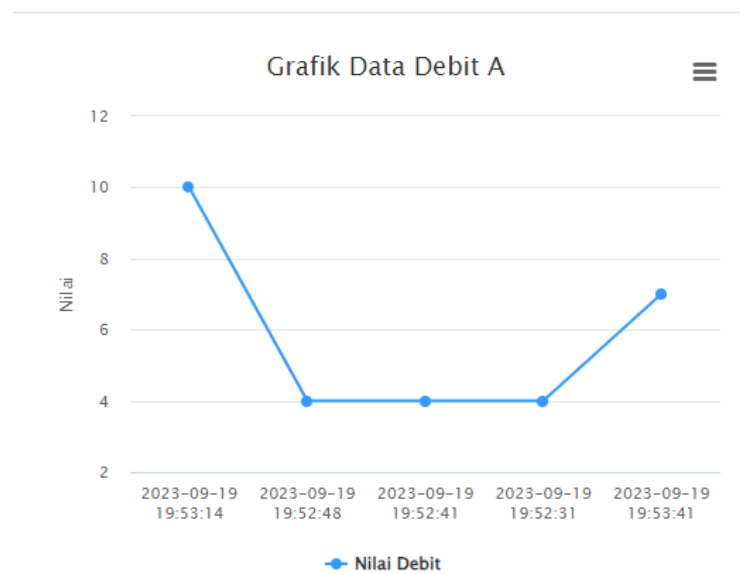
Tabel 5. 3 Hasil Pengujian Rekap Data

<b>Aktifitas Pengujian</b>	<b>Realisasi yang diharapkan</b>	<b>Hasil Pengujian</b>	<b>Kesimpulan</b>
Tampilan dashboard	Menampilkan data monitoring debit air, status aliran air, dan status pipa air	Monitoring dapat berfungsi sesuai yang diharapkan	valid
Rekap data pada Rumah A	Menampilkan rekpa data rumah A	Berhasil menampilkan rekap data rumah A	Valid

<b>Aktifitas Pengujian</b>	<b>Realisasi yang diharapkan</b>	<b>Hasil Pengujian</b>	<b>Kesimpulan</b>
Rekap data pada Rumah B	Menampilkan rekpa data rumah B	Berhasil menampilkan rekap data rumah B	Valid

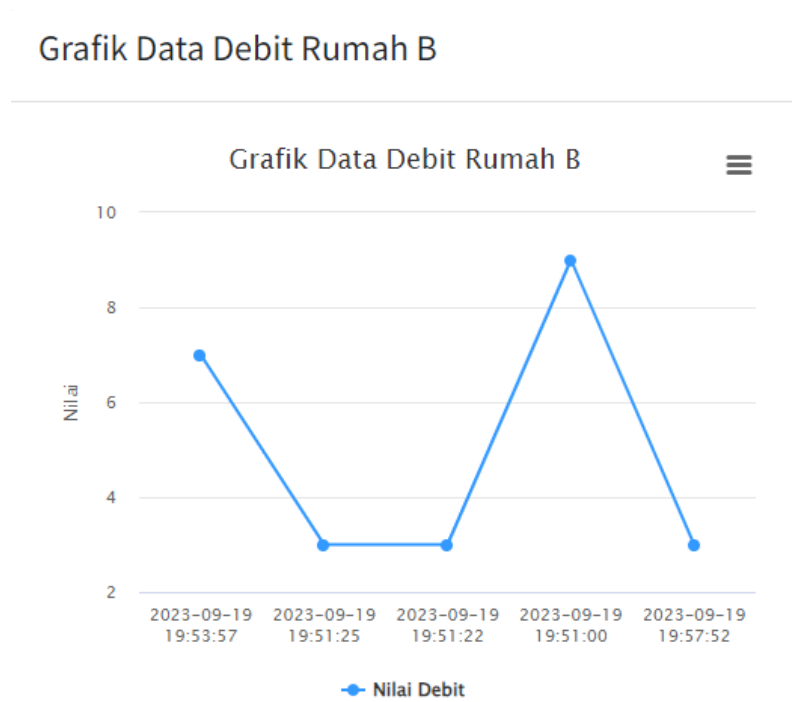
3. Hasil pengujian perbandingan debit rumah A dan rumah B pada Grafik

Grafik Data Debit Rumah A



Gambar 5. 7 Grafik Data Debit Rumah A





Gambar 5. 8 Grafik Data Debit Rumah B

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **6.1 Kesimpulan**

Berdasarkan hasil pengujian yang telah dilakukan maka dapat diambil beberapa kesimpulan yaitu hasil diperoleh dari penelitian ini adalah sistem dapat memonitoring kecepatan debit air rumah A dan B, status pipa, dan status aliran air melalui *website* berbasis monitoring.

#### **6.2 Saran**

Berdasarkan kesimpulan dari penelitian ini, maka ada beberapa saran yang peneliti rekomendasikan selain memonitoring kecepatan debit air rumah A dan B, status pipa, status aliran air diharapkan *website* ini untuk kedepannya bisa dikembangkan lebih luas lagi seperti memberikan informasi yang lebih lengkap lagi, lebih menarik, sistem dapat secara *realtime* dan akurat.

## DAFTAR PUSTAKA

- [1] A. G. R. Ilma and F. M. Amrulloh, "Implementasi Sensor Waterflow Untuk Sistem Monitoring Pemakaian Debit Air HIPPAM Berbasis Android," *Jurnal Krisnadana*, pp. 344-353, 2023.
- [2] R. Peletz, J. Kisiangani and M. Bonham, "Mengapa Program Pemantauan Kualitas Air Berhasil atau Gagal? Analisis Komparatif Kualitatif Terhadap Sistem Pengujian yang Diatur di Afrika sub-Sahara," *International Journal of Hygiene and Environmental Health*, pp. 907-920, 2018.
- [3] M. Galina, M. W. Ramadhani and J. W. Simaatupang, "Prototipe Sistem Pemantauan Penggunaan Listrik dan Air Pascabayar," pp. 304-308, 2019.
- [4] S. R. Bagas D, U. A. Ahmad and J. S. Wicaksana, "Aplikasi Mobile Survei Kualitas Dan Debit Air Berbasis Firebase Studi Kasus Pdam Kabupaten Madiun Mobile Survey," *e-Proceeding of Engineering*, vol. 9, pp. 936-944, 2022.
- [5] T. Sundana, F. A. Johari and F. A. Ariiq, "Prototype Sistem Monitoring Kebocoran Pipad Distribusi Air Berbasis Scada," *Jurnal Indonesia Sosial Teknologi*, pp. 74-86, 2022.
- [6] N. Afthiroh, N. Hidayah and S. , "Sistem Monitoring Penggunaan Air Pdam Berbasis Website," *Jurnal Teknik Komputer*, pp. 1-104, 2019.
- [7] J. I. Dimu, "Perancangan Alat Deteksi Letak Kebocoran Pipa Menggunakan Sensor Flow Meter Berbasis Arduino," *Teknik Elektro*, pp. 1-54, 2022.
- [8] H. Kusuma, F. Ramadhan, A. A. Alawi, R. Nauval and J. Setiawan, "Prototype Pendeteksi Kebocoran Pipa Berbasis IoT Menggunakan Nodemcu ESP8266 Melalui Dashboard Adafruit.Io," *Teknologi dan Informasi Bisnis*, pp. 327-333, 2021.
- [9] A. M. Fauzi and R. Hermawan, "Sistem Pendeteksi Tekanan Air Berbasis IoT di Pdam Tirta Galuh Kabupaten Ciamis," *Jurnal Saintesa*, vol. 2, pp. 1-8, 2022.
- [10] "Bab III Metodologi Penelitian," pp. 29-36, 2019.

# LAMPIRAN

## Lampiran 1 Source Code Halaman Dashboard

```
<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
  <!-- Content Header (Page header) -->
  <div class="content-header">
    <div class="container-fluid">
      <div class="row mb-2">
        <div class="col-sm-6">
          <h1 class="m-0"></h1>
        </div><!-- /.col -->
      </div><!-- /.row -->
    </div><!-- /.container-fluid -->
  </div>
<!-- /.content-header -->

<!-- Main content -->
<section class="content">
  <div class="container-fluid">
    <h1 class="text-center mb-3">Monitoring Pipa Air</h1>
    <!-- Info boxes -->
    <div class="row">
      <div class="col-md-4">
        <div class="info-box">
          <span class="info-box-icon bg-info elevation-1"><i class="fas fa-
cog"></i></span>

          <div class="info-box-content">
            <span class="info-box-text">Status Aliran Rumah A</span>
```

```

<span class="info-box-number" id="aliranA">
    </span>
</div>
<!-- /.info-box-content -->
</div>
<!-- /.info-box -->
</div>
<div class="col-md-4">
    <div class="info-box">
        <span class="info-box-icon bg-info elevation-1"><i class="fas fa-
tint"></i></span>

        <div class="info-box-content">
            <span class="info-box-text">Debit Rumah A</span>
            <span class="info-box-number" id="debitA">
                </span>
            </div>
        <!-- /.info-box-content -->
    </div>
    <!-- /.info-box -->
</div>
<div class="col-md-4">
    <div class="info-box">
        <span class="info-box-icon bg-info elevation-1"><i class="fas fa-
bell"></i></span>

```

```

    <div class="info-box-content">
      <span class="info-box-text">Status Pipa Rumah A</span>
      <span class="info-box-number" id="pipaA">
        </span>
      </div>
    <!-- /.info-box-content -->
  </div>
  <!-- /.info-box -->
</div>
</div>

```

```

<div class="row">
  <div class="col-md-4">
    <div class="info-box">
      <span class="info-box-icon bg-danger elevation-1"><i class="fas
fa-cog"></i></span>

```

```

    <div class="info-box-content">
      <span class="info-box-text">Status Aliran Rumah B</span>
      <span class="info-box-number" id="aliranB">
        </span>
      </div>
    <!-- /.info-box-content -->
  </div>
  <!-- /.info-box -->
</div>
<div class="col-md-4">

```

```

<div class="info-box">
    <span class="info-box-icon bg-danger elevation-1"><i class="fas
fa-tint"></i></span>

    <div class="info-box-content">
        <span class="info-box-text">Debit Rumah B</span>
        <span class="info-box-number" id="debitB">
            </span>
        </div>
    <!-- /.info-box-content -->
</div>
<!-- /.info-box -->
</div>
<div class="col-md-4">
    <div class="info-box">
        <span class="info-box-icon bg-danger elevation-1"><i class="fas
fa-bell"></i></span>

        <div class="info-box-content">
            <span class="info-box-text">Status Pipa Rumah B</span>
            <span class="info-box-number" id="pipaB">
                </span>
            </div>
        <!-- /.info-box-content -->
    </div>
<!-- /.info-box -->
</div>
</div>

```



```
<div class="row">
  <div class="col-lg-6 col-md-12 col-sm-12 col-xl-6">
    <div class="card">
      <div class="card-header">
        <h4>Grafik Data Debit Rumah A</h4>
      </div>
      <div class="card-body">
        <div id="chart-debit-a"></div>
      </div>
    </div>
  </div>
  <div class="col-lg-6 col-md-12 col-sm-12 col-xl-6">
    <div class="card">
      <div class="card-header">
        <h4>Grafik Data Debit Rumah B</h4>
      </div>
      <div class="card-body">
        <div id="chart-debit-b"></div>
      </div>
    </div>
  </div>
</div>
</section>
</div>
```

```
<script src="https://code.jquery.com/jquery-3.7.0.min.js" integrity="sha256-2PmVv0kuTBOenSvLm6bvfbSShRrUJ+3A7x6P5Ebd07/g=" crossorigin="anonymous"></script>
```

```
<script src="<?= base_url('assets/highcharts/highcharts.js'); ?>"></script>
```

```
<script src="<?= base_url('assets/highcharts/exporting.js'); ?>"></script>
```

```
<script src="<?= base_url('assets/highcharts/export-data.js'); ?>"></script>
```

```
<script src="<?= base_url('assets/highcharts/accessibility.js'); ?>"></script>
```

```
<script>
```

```
function get_realtime() {
```

```
    $.ajax({
```

```
        url: "<?= base_url('admin/getRealtime'); ?>",
```

```
        dataType: "json",
```

```
        success: function(response) {
```

```
            $('#aliranA').text((response.rumahA.statusAliran == 1) ? "Terbuka" : "Tertutup");
```

```
            $('#debitA').text(response.rumahA.debit);
```

```
            $('#pipaA').text((response.rumahA.statusPipa == 1) ? "Teridentifikasi Bocor" : "Teridentifikasi Tidak Bocor");
```

```
            $('#aliranB').text((response.rumahB.statusAliran == 1) ? "Terbuka" : "Tertutup");
```

```
            $('#debitB').text(response.rumahB.debit);
```

```
            $('#pipaB').text((response.rumahB.statusPipa == 1) ? "Teridentifikasi Bocor" : "Teridentifikasi Tidak Bocor");
```

```
            setTimeout(get_realtime, 2000)
```

```
        }
```

```
    });
```

```
}
```

```
var chartDebitA;
```

```
var chartDebitB;
```

```
var totalA = 0;
```

```
var totalB = 0;
```

```
function getGrafikDebitA() {
```

```
    $.ajax({
```

```
        url: '<?php echo base_url('admin/getGrafik') ?>',
```

```
        dataType: 'json',
```

```
        success: function(result) {
```

```
            if (result.rumahA.length > totalA) {
```

```
                totalA = result.rumahA.length;
```

```
                var i;
```

```
                var debitA = [];
```

```
                var dateA = [];
```

```
                chartDebitA.series[0].update({
```

```
                    color: "#3399ff"
```

```
                });
```

```
                for (i = 0; i < result.rumahA.length; i++) {
```

```
                    debitA[i] = Number(result.rumahA[i].debit);
```

```
                    dateA[i] = result.rumahA[i].createdAt;
```

```

chartDebitA.series[0].setData(debitA);

        chartDebitA.xAxis[0].setCategories(dateA);
    }
} else if (result.rumahA.length <= totalA) {
    var i;
    var debitA = [];
    var dateA = [];

    chartDebitA.series[0].update({
        color: "#3399ff"
    });

    for (i = 0; i < result.rumahA.length; i++) {
        debitA[i] = Number(result.rumahA[i].debit);

        dateA[i] = result.rumahA[i].createdAt;

        chartDebitA.series[0].setData(debitA);

        chartDebitA.xAxis[0].setCategories(dateA);
    }
}

setTimeout(getGrafikDebitA, 5000);
}
});

```

```
}
```

```
function getGrafikDebitB() {  
    $.ajax({  
        url: '<?php echo base_url('admin/getGrafik') ?>',  
        dataType: 'json',  
        success: function(result) {  
            if (result.rumahB.length > totalB) {  
                totalB = result.rumahB.length;  
  
                var i;  
                var debitB = [];  
                var dateB = [];  
  
                chartDebitB.series[0].update({  
                    color: "#3399ff"  
                });  
  
                for (i = 0; i < result.rumahB.length; i++) {  
                    debitB[i] = Number(result.rumahB[i].debit);  
  
                    dateB[i] = result.rumahB[i].createdAt;  
  
                    chartDebitB.series[0].setData(debitB);  
  
                    chartDebitB.xAxis[0].setCategories(dateB);  
                }  
            }  
        }  
    });  
}
```

```

} else if (result.rumahB.length <= totalB) {
    var i;
    var debitB = [];
    var dateB = [];

    chartDebitB.series[0].update({
        color: "#3399ff"
    });

    for (i = 0; i < result.rumahB.length; i++) {
        debitB[i] = Number(result.rumahB[i].debit);

        dateB[i] = result.rumahB[i].createdAt;

        chartDebitB.series[0].setData(debitB);

        chartDebitB.xAxis[0].setCategories(dateB);
    }
}

setTimeout(getGrafikDebitB, 5000);
}
});

```

```
}
```

```
document.addEventListener('DOMContentLoaded', function() {  
  get_realtime();
```

```
  
  chartDebitA = Highcharts.chart('chart-debit-a', {  
    chart: {  
      type: 'line',  
      events: {  
        load: getGrafikDebitA  
      }  
    },  
    title: {  
      text: 'Grafik Data Debit A'  
    },  
    yAxis: {  
      title: {  
        text: 'Nilai'  
      }  
    },  
    xAxis: {  
  
    },  
    series: [{  
      name: "Nilai Debit"  
    }]  
  });
```

```
chartDebitB = Highcharts.chart('chart-debit-b', {
  chart: {
    type: 'line',
    events: {
      load: getGrafikDebitB
    }
  },
  title: {
    text: 'Grafik Data Debit Rumah B'
  },
  yAxis: {
    title: {
      text: 'Nilai'
    }
  },
  xAxis: {
  },
  series: [{
    name: "Nilai Debit"
  }]
});
});
</script>
```



## SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Ida Afriliana.ST, M.Kom  
NIDN : 0624047703  
NIPY : 12.013.168  
Jabatan Struktural : Ka. Prodi DIII Teknik Komputer  
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut :

Nama : Erni Azizah  
NIM : 20040141  
Program Studi : Teknik Komputer

Judul TA : SISTEM MONITORING PENDETEKSI KEBOCORAN  
PIPA AIR PDAM BERBASIS IOT

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

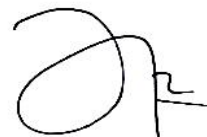
Tegal, 26 Juni 2023

Mengetahui,  
Ka. Prodi DIII Teknik Komputer



Ida Afriliana.ST, M.Kom  
NIPY. 12.013.168

Dosen Pembimbing I,



Ida Afriliana.ST, M.Kom  
NIPY. 12.013.168

## SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Abdul Basit, S.Kom., MT  
NIDN : 0608129106  
NIPY : 01.015.198  
Jabatan Struktural : Sekretaris Prodi DIII Teknik Komputer  
Jabatan Fungsional : Asisten Ahli

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut :

Nama : Erni Azizah  
NIM : 20040141  
Program Studi : Teknik Komputer

Judul TA : SISTEM MONITORING PENDETEKSI KEBOCORAN  
PIPA AIR PDAM BERBASIS IOT

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 26 Juni 2023

Mengetahui,

Ka. Prodi DIII Teknik Komputer

  
Ida Afriliana, ST, M.Kom  
NIPY. 12.013.168

Dosen Pembimbing II,



Abdul Basit, S.Kom., MT  
NIPY. 01.015.198



No. : 006.03/KMP.PHB/IV/2023  
Lampiran : -  
Perihal : Permohonan Izin Observasi Tugas Akhir (TA)

Kepada Yth.

Kepala PDAM DESA LUWUNGRAGI

Jl. K.H. Syarkawi, Kabupaten Brebes, Jawa Tengah, Desa Luwungragi


Dengan Hormat,

Sehubungan dengan tugas mata kuliah Tugas Akhir (TA) yang akan diselenggarakan di semester VI (Genap) Program Studi D III Teknik Komputer Politeknik Harapan Bersama Tegal, Maka dengan ini kami mengajukan izin observasi pengambilan data di PDAM DESA LUWUNGRAGI yang Bapak / Ibu Pimpin, untuk kepentingan dalam pembuatan produk Tugas Akhir, dengan Mahasiswa sebagai berikut:

No.	NIM	Nama	No. HP
1	20040131	NURUL HIKMAH HARTAMI	085921166028
2	20040141	ERNI AZIZAH	081258520150

Demikian surat permohonan ini kami sampaikan atas izin dan kerjasamanya kami sampaikan terima kasih.

Tegal, 04 April 2023  
Ka. Prodi DIII Teknik Komputer  
Politeknik Harapan Bersama Tegal

  
**Ida Afriliana, ST, M.Kom**  
NIPY. 12.013.168