

SISTEM MONITORING IRIGASI DAN PREDIKSI DEBIT AIR BERBASIS IOT DAN SUPPORT VECTOR MACHINE(SVM)

Toqa Aldila Cinderatama¹, Rinanza Zulmy Alhamri², Yoppy Yunhasnawa³, Fery Sofian Efendi⁴, Rudy Ariyanto⁵

^{1,2,4}D3 Manajemen Informatika Kampus Kediri, Jurusan Teknologi Informasi, Politeknik Negeri Malang, Indonesia, ^{3,5}D4 Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, Indonesia
¹ toqa.aldila@polinema.ac.id, ² rinanza.zulmy@polinema.ac.id, ³ yunhasnawa@polinema.ac.id, ⁴ fery.sofian@polinema.ac.id, ⁵ rudy.ariyanto@polinema.ac.id

Abstrak

Menurut data BPS yang dirilis pada Oktober 2023 mengenai data pertanian, jika dilihat dari luasan lahan pertanian, luas panen padi pada 2023 diperkirakan sekitar 10,20 juta hectare mengalami penurunan sebanyak 255,79 ribu hektare atau 2,45 persen dibandingkan luas panen padi di 2022 yang sebesar 10,45 juta hektare. Dengan penurunan luasan lahan panen padi tersebut perlu dipikirkan berbagai cara untuk lebih meningkatkan hasil pertanian dan mencegah terjadinya kesenjangan yang tinggi antara tingkat kebutuhan dan tingkat pemenuhan bahan makanan. Salah satu cara yang dapat dilakukan adalah mengoptimalkan lahan pertanian yang tersedia dengan panca usaha tani dan pengairan yang optimal, sehingga diperlukan jaringan irigasi beserta bangunan irigasi dalam keadaan baik. Penelitian ini memanfaatkan teknologi IoT untuk monitoring debit serta memprediksinya menggunakan metode Supervised learning-Support Vector Machine (SVM). Metode penelitian yang akan dilakukan meliputi : Analisis Kebutuhan, perancangan, implementasi, pengujian dan pemeliharaan. Hasil dari penelitian ini adalah telah berhasil Merancang dan membangun sistem monitoring debit air pada irigasi pertanian berbasis platform Internet of Things menggunakan sensor water flow, sensor ultrasonic HCSR04 dan sensor suhu ds18b20, dilengkapi dengan aplikasi monitoring berbasis android untuk memonitor dan mengambil data history debit air. Aplikasi prediksi debit air pada irigasi pertanian juga telah diimplementasikan menggunakan support vector machine terutama menggunakan linier regression dan diperoleh hasil prediksi debit air dengan kondisi saat pemantauan berada pada kisaran 1500 liter/menit atau 25 liter per detik.

Kata kunci : IoT, SVM, Irigasi, Mobile apps, Sensor Water Flow, Sensor Ultrasonic HCSR04

1. Pendahuluan

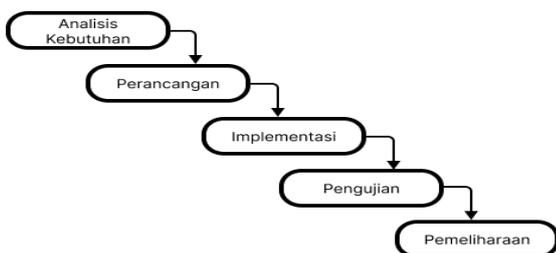
Indonesia merupakan salah satu negara agraris dimana sektor pertaniannya sangat bergantung pada curah hujan (Krismianto, 2013). Menurut data BPS yang dirilis pada Oktober 2023 luas panen padi diperkirakan sebesar 10,20 juta hektare dengan produksi padi sekitar 53,63 juta ton gabah kering giling (GKG). Jika dikonversikan menjadi beras untuk konsumsi pangan penduduk, produksi beras pada 2023 diperkirakan sebesar 30,90 juta ton. (BPS, 2023). Jika dilihat dari luasan lahan pertanian, luas panen padi pada 2023 diperkirakan sekitar 10,20 juta hectare tersebut mengalami penurunan sebanyak 255,79 ribu hektare atau 2,45 persen dibandingkan luas panen padi di 2022 yang sebesar 10,45 juta hektare. Dengan penurunan luasan lahan panen padi tersebut perlu dipikirkan berbagai cara untuk lebih meningkatkan hasil pertanian dan mencegah terjadinya kesenjangan yang tinggi antara tingkat kebutuhan dan tingkat pemenuhan bahan makanan. Salah satu cara yang dapat dilakukan adalah mengoptimalkan lahan pertanian yang tersedia dengan panca usaha tani dan pengairan yang optimal, sehingga diperlukan jaringan irigasi beserta bangunan irigasi dalam keadaan baik. (Sri Astutik, 2021). Disisi yang lain perkembangan Internet of Things (IoT)

telah mengalami pertumbuhan yang pesat dalam beberapa tahun terakhir. Beberapa aspek utama perkembangannya meliputi peningkatan koneksi, pengembangan sensor yang lebih canggih, dan penerapan teknologi nirkabel yang semakin canggih. Diperkirakan bahwa jumlah perangkat IoT akan terus bertambah, dengan perkiraan mencapai sekitar 100 miliar di tahun 2030. Selain peningkatan koneksi, pengembangan sensor, dan teknologi nirkabel, IoT juga mengalami perluasan target penggunaan. Berbagai sektor mulai dari industri hingga pertanian sudah banyak yang menggunakan sistem IoT sebagai alat untuk membantu kegiatan. Beberapa penelitian terdahulu terkait monitoring berbasis IoT meliputi penelitian yang dilakukan oleh Gunawan et. al. dimana dirancang sebuah *prototype* sistem monitoring bendungan air yang dapat bekerja dengan proses sistem monitoring air secara otomatis dan buka tutup pintu air yang di kontrol dari jarak jauh. Sistem ini dirancang menggunakan mikrokontroler NodeMCU ESP32 sebagai pusat kontrol sistem dan *software blynk* sebagai sistem control monitoring jarak jauh. Pada penelitian ini telah berhasil dirancang suatu *prototype* bendungan air dengan sistem buka tutup pintu air secara otomatis dengan akurasi 99,275% (Gugun Gunawan, 2022). Selanjutnya penelitian yang dilakukan oleh Muh.

Maulana Syahaddan dan Anita Fira Waluyo memiliki tujuan untuk membangun teknologi mobile yang terintegrasi dengan perangkat Arduino dan ESP8266 berbasis Internet of Things. Tujuan dari aplikasi monitoring sawah antara lain memperoleh informasi kelembaban tanah, ketinggian air irigasi, ketinggian air pompa, dan otomatisasi pengusir hama menggunakan speaker yang mengeluarkan suara berfrekuensi. Aplikasi ini juga memudahkan petani untuk memantau kondisi sawahnya dari jarak jauh dan kapan saja, sehingga mengurangi biaya bahan bakar yang harus dikeluarkan setiap kali melakukan perjalanan ke sawah. Aplikasi ini juga sangat efektif bagi masyarakat yang memiliki sawah namun terkendala pekerjaan utama. Keunggulan pembuatan aplikasi terletak pada penggunaan database yaitu database Firebase, karena menyimpan informasi kelembaban tanah, ketinggian air irigasi, dan pengusiran hama dengan cepat tersampaikan dan terupdate secara real time dengan mengirimkan data dari tools Internet of Things ke aplikasi seluler (Muhamad Maulana Syahaddan, 2023). Sedangkan penelitian terdahulu terkait Support Vector Machine (SVM) membuktikan bahwa metode SVM cukup baik dalam studi kasus prediksi. Penelitian yang dilakukan oleh (Aulia Putri, 2023) dalam memprediksi kelulusan mahasiswa tingkat akhir menghasilkan akurasi 84%. Penelitian Emy Haryatmi dan Sheila Pramita Hervianti model prediksi kelulusan mahasiswa tepat waktu menghasilkan nilai akurasi dari data testing sebesar 94,4% (Emy Haryatmi, 2021) Selanjutnya, Habibah dan Rahmadden dalam penelitiannya melakukan prediksi harga cryptocurrency menggunakan Algoritma Support vector machine dipadukan dengan Particle Swarm Optimization mendapatkan akurasi kinerja maksimal mencapai 95% (Habibah br Lumbantobing, 2023). Berdasarkan beberapa penelitian tersebut Support Vector Machine merupakan metode yang dapat menghasilkan akurasi cukup tinggi dalam melakukan prediksi. Penelitian ini memanfaatkan metode *Supervised learning Support Vector Machine (SVM)* untuk prediksi debit air pada irigasi pertanian berdasarkan data monitoring yang didapat dari penerapan sensor-sensor IoT.

2. Metode

Dalam penelitian ini menerapkan *Waterfall Model* sebagai kerangka kerja untuk pengembangan sistem. Untuk tahapannya dapat dilihat pada Gambar 1 berikut.



Gambar 1. Metode *Waterfall*

Gambar 1 merupakan *Waterfall Model* dalam sebuah sistem. Uraian masing-masing tahapan dapat dijelaskan sebagai berikut:

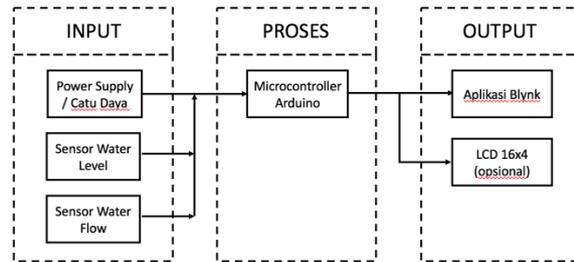
1. Analisis Kebutuhan

Tahap ini mencakup identifikasi dan pemahaman kebutuhan sistem. Analisis dilakukan untuk menentukan fitur dan fungsi yang diperlukan dalam penelitian ini. Dari apa saja alat yang dibutuhkan dan sistem apa yang akan digunakan. Fokus pada pemahaman mendalam terhadap kondisi irigasi aspek pengelolaan dan kebutuhan monitoring irigasi pertanian.

2. Perancangan

Pada tahap perancangan, konsep dan struktur sistem diatur secara rinci. Ini mencakup perancangan arsitektur, skema sensor, rancangan antarmuka mobile untuk monitoring. Tujuannya adalah menciptakan kerangka kerja yang efisien untuk memudahkan pembuatan sistem nantinya melakukan implementasi.

Blok Diagram Proses Sistem



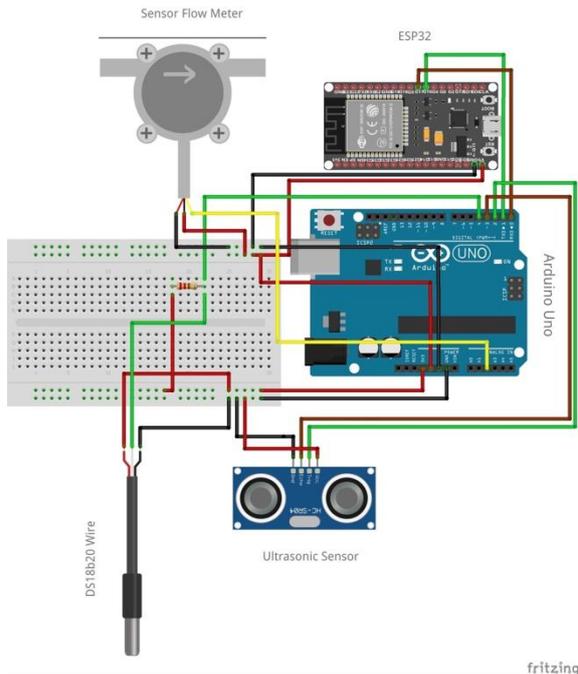
Gambar 1 Blok Diagram Proses

Dari blok diagram proses sistem yang di tunjukkan pada Gambar 2, dapat dijelaskan sebagai berikut:

1. Catu Daya berfungsi untuk memberi tegangan masukan sebesar 5 Volt DC terhadap mikrokontroller, sensor water flow, sensor water level, LCD 16x4.
2. Mikrokontroller Arduino berfungsi untuk memproses data yang diterima sensor sebagai input.
3. LCD 16x4 digunakan untuk menampilkan hasil pembacaan sensor.
4. Blynk berfungsi sebagai wadah hasil pemantauan jarak jauh yang menampilkan hasil pembacaan sensor, status level air dan disertai dengan diagram berdasarkan kolektif waktu secara real time.

Perancangan Perangkat Keras

Ada beberapa tahapan untuk pembuatan sebuah rangkaian yaitu membuat sebuah schematic rangkaian menggunakan software fritzing, membuat sebuah desain alat, penentuan tata letak pemasangan masing-masing komponen, pengkabelan dan penyolderan pada kabel-kabel penghubung, pemasangan rangkaian di sebuah box panel, seperti power supply, mikrokontroller, dan sensor.



Gambar 2. Schematic Rangkaian Alat

Gambar 3 menunjukkan schematic rangkaian alat, dimana alat ini bekerja dengan tiga bagian penting yang saling berhubungan satu sama lain, yaitu hardware, software, dan mekanik. Dalam kondisi aktif, mikrokontroler bertugas sebagai pusat kontrol yang mengendalikan seluruh komponen hardware, pada saat mikrokontroler terhubung dengan jaringan internet hotspot/wifi, sensor water level akan mendeteksi nilai level air dengan prinsip Sensor ini bekerja berdasarkan prinsip resistansi variabel. Sensor ini terdiri dari serangkaian konduktor terbuka paralel. Bersama-sama rangkaian ini bertindak sebagai resistor variabel, yang resistansinya bervariasi sesuai dengan ketinggian air di tangki air. Semakin banyak sensor air yang terendam, semakin baik konduktivitasnya dan semakin rendah resistansinya. Semakin sedikit sensor air terendam, semakin buruk konduktivitasnya dan semakin tinggi resistansinya. Output dari sensor ketinggian air sesuai dengan ketahanan air yang dihasilkan. yaitu akan menghasilkan tegangan yang sebanding dengan resistansi.

Untuk memastikan sensor memberikan data yang akurat, perlu dilakukan kalibrasi sensor untuk jenis air tertentu yang akan digunakan. Sebelum memulai proyek atau menyimpan data di sensor, harus dilihat pembacaan apa yang sebenarnya didapatkan dari sensor tersebut. Catat nilai yang tepat dari keluaran sensor saat sensor benar-benar kering, saat terendam sebagian di dalam air, dan saat terendam seluruhnya. Dalam kalibrasi ini, perlu dilakukan beberapa trial and error. Akan sulit untuk mulai mengkalibrasi sensor ini dengan nilai yang akurat. setelah didapatkan nilai yang akurat, kemudian diharapkan dapat menangani proyek tersebut dengan sangat baik.

Kemudian untuk sensor water flow akan bekerja dengan prinsip air yang mengalir akan melewati katup dan akan membuat rotor magnet berputar dengan kecepatan tertentu sesuai dengan tingkat aliran yang mengalir. Medan magnet yang terdapat pada rotor akan memberikan efek pada sensor efek hall dan itu akan menghasilkan sebuah sinyal pulsa yang berupa tegangan (Pulse Width Modulator). Output dari pulsa tegangan memiliki tingkat tegangan yang sama dengan input dengan frekuensi laju aliran air. Sinyal tersebut dapat diolah menjadi data digital melalui pengendali atau mikrokontroler.

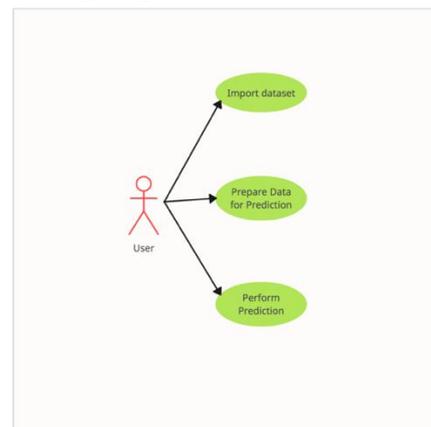
Data dari water level dan water flow sensor tersebut kemudian akan ditampilkan pada LCD 16x4 yang berada pada box panel (opsional) maupun pada aplikasi blynk.

Arduino akan mengirimkan hasil real-time pembacaan data jarak level air dan debit air yang diperoleh dari hasil pembacaan sensor water level dan water flow, aplikasi blynk akan terus-menerus memperbarui data secara real-time ketika aplikasi terhubung dengan Arduino melalui jaringan internet.

Data dari sensor ini kemudian juga akan disimpan ke sistem backend yang nantinya akan dijadikan sebagai dataset untuk prediksi debit air dan dipadukan juga dengan data curah hujan.

Perancangan Aplikasi Prediksi berbasis Web

Pada aplikasi prediksi berbasis web terdapat 3 persyaratan sistem utama yang digambarkan dalam use case diagram pada Gambar 4.



Gambar 4. Use Case Diagram pada Aplikasi Prediksi

Sebagaimana bisa dilihat pada diagram tersebut, aplikasi yang dibuat ini nantinya harus bisa menerima input berupa data set yang diunggah oleh pengguna atau mengambil langsung data dari firebase. Data set ini yang diupload/import ini berupa file CSV yang setelah diterima di server akan disimpan di database. Dataset inilah yang kemudian akan dijadikan untuk prediksi. Selain data dari sensor data input lain yang akan dijadikan parameter adalah data curah hujan. Setelah data disiapkan pada akhirnya sistem harus dapat melakukan prediksi secara otomatis dan menampilkan visualisasi hasilnya.



Gambar 5. Arsitektur Diagram Sistem IoT

Gambar 5 mengilustrasikan secara umum pada sistem monitoring menggunakan platform IoT, alat IoT akan mengirim data dari sensor-sensor melalui internet dan disimpan ke dalam database firebase untuk kemudian hasil monitoring tersebut akan ditampilkan ke user melalui aplikasi mobile.



Gambar 6. Arsitektur Diagram Aplikasi Prediksi

Arsitektur diagram aplikasi prediksi berjalan pada backend dan ditunjukkan pada Gambar 6, dimana alur pengguna dimulai dari interaksi mereka dengan website sebagai antarmuka. Melalui website, pengguna dapat mengakses berbagai fitur dan layanan yang disediakan oleh sistem. Pengguna dapat melakukan tindakan melihat informasi pada perangkat IoT yang terhubung. Data yang dihasilkan dari interaksi pengguna dengan website akan disimpan atau diakses dari database. Database bertindak sebagai tempat penyimpanan data, menyimpan informasi yang diperlukan oleh sistem untuk diproses atau diakses kembali. Untuk memungkinkan komunikasi antara pengguna dan website, serta pengguna dengan perangkat IoT, koneksi internet sangat penting. Koneksi ini memfasilitasi transfer data, memungkinkan pengguna untuk berinteraksi secara real-time dengan sistem, dan memungkinkan perangkat IoT untuk mengirim dan menerima instruksi atau data. Perangkat IoT, seperti sensor atau perangkat kontrol, terhubung ke internet atau sistem melalui protokol IoT. Mereka berperan dalam mengukur, mengirim, atau menerima data yang relevan terkait lingkungan atau fungsi yang dipantau, dan dapat dikendalikan atau diatur dari website.

Perancangan Halaman Monitoring Irigasi

Halaman monitoring irigasi ini didesain untuk menampilkan data penting terkait kondisi irigasi secara langsung. Ada tiga informasi utama ditampilkan, yaitu suhu air, ketinggian air, dan debit air. Halaman ini juga memiliki dua tombol, yaitu "Pilih Tanggal," yang memungkinkan pengguna untuk membuka kalender atau memilih tanggal, serta tombol "Download," yang memungkinkan pengunduhan data debit air berdasarkan tanggal yang dipilih. Di bagian bawah halaman, terdapat dua ikon navigasi, yaitu ikon layar monitor yang mengarah ke halaman monitoring, dan ikon grafik yang membawa pengguna ke halaman history. Rancangan halaman monitoring irigasi ditunjukkan pada Gambar 7 di bawah ini.



Gambar 7. Rancangan Halaman Monitoring Irigasi

Perancangan Halaman History

Halaman ini dirancang untuk menampilkan riwayat data irigasi dalam bentuk grafik, memungkinkan pengguna untuk menganalisis riwayat data dengan memilih tanggal tertentu menggunakan tombol "Pilih Tanggal" atau mengatur ulang tampilan grafik dengan tombol "Reset." Grafik di tengah halaman visualisasikan perubahan datadebit air, berdasarkan tanggal yang dipilih. Navigasi di bagian bawah halaman memungkinkan pengguna untuk kembali ke halaman monitoring atau tetep di halaman history, ditunjukkan pada Gambar 8.



Gambar 8. Rancangan Halaman History

Preprocessing data pada aplikasi monitoring dan aplikasi prediksi

Data yang akan diolah adalah data primer yang diperoleh dari hasil pembacaan sensor-sensor meliputi, hasil dari sensor water flow, sensor ultrasonic, dan sensor suhu. Untuk data dari sensor suhu ditampilkan langsung ke dashboard aplikasi, sedangkan dari sensor ultrasonic digunakan untuk mengukur jarak ketinggian permukaan air, dimana,

$$S = \frac{340 \cdot T}{2} \tag{1}$$

S = jarak (cm)
 T = waktu (detik)
 340 = merupakan nilai sinyal yang dipancarkan dalam gelombang bunyi dengan kecepatan sekitar 340 m/s.

Data jarak yang diperoleh dari sensor ultrasonic yaitu jarak sensor dengan ketinggian permukaan air akan digunakan sebagai pengurang dari ketinggian saluran irigasi setinggi 275 sentimeter sehingga akan diperoleh ketinggian permukaan air pada saluran irigasi.

Kemudian untuk data primer dari hasil pembacaan sensor waterflow perlu dilakukan proses normalisasi karena sering masih terdapat inkonsistensi pembacaan hasil sensor water flow. Gambar 9 menunjukkan hasil raw data pembacaan sensor.

1	Debit	Jam	Tanggal
32	16	8:30:59	10/2/24
33	0	8:31:09	10/2/24
34	0	8:31:19	10/2/24
35	0	8:31:29	10/2/24
36	0	8:31:39	10/2/24
37	0	8:31:49	10/2/24
38	0	8:31:59	10/2/24
39	88	8:32:09	10/2/24
40	1136	8:32:19	10/2/24
41	1480	8:32:28	10/2/24
42	400	8:32:38	10/2/24
43	1184	8:32:49	10/2/24
44	1224	8:32:59	10/2/24
45	272	8:33:09	10/2/24
46	480	8:33:19	10/2/24
47	0	8:33:29	10/2/24
48	0	8:33:39	10/2/24
49	0	8:33:48	10/2/24
50	752	8:33:59	10/2/24
51	1984	8:34:09	10/2/24
52	2048	8:34:19	10/2/24
53	2112	8:34:29	10/2/24
54	2056	8:34:40	10/2/24
55	1448	8:34:49	10/2/24
56	720	8:34:59	10/2/24
57	0	8:35:10	10/2/24
58	0	8:35:20	10/2/24

Gambar 9. Data mentah hasil pembacaan sensor waterflow

Data mentah pembacaan hasil sensor water flow akan dilakukan normalisasi dengan cara menghilangkan data dengan hasil debit air 0 ditunjukkan pada Gambar 10 di bawah ini.

1	Debit	Jam	Tanggal
2	8	8:26:29	10/2/24
3	8	8:27:08	10/2/24
4	8	8:30:11	10/2/24
5	16	8:30:59	10/2/24
6	88	8:32:09	10/2/24
7	1136	8:32:19	10/2/24
8	1480	8:32:28	10/2/24
9	400	8:32:38	10/2/24
10	1184	8:32:49	10/2/24
11	1224	8:32:59	10/2/24
12	272	8:33:09	10/2/24
13	480	8:33:19	10/2/24
14	752	8:33:59	10/2/24
15	1984	8:34:09	10/2/24
16	2048	8:34:19	10/2/24
17	2112	8:34:29	10/2/24
18	2056	8:34:40	10/2/24
19	1448	8:34:49	10/2/24
20	720	8:34:59	10/2/24

Gambar 10. Data input untuk aplikasi prediksi

3. Implementasi

Pada tahap ini instalasi sensor, konfigurasi perangkat keras dan perangkat lunak, serta pengembangan aplikasi mobile monitoring dan aplikasi web untuk prediksi dilakukan dalam tahap ini.

4. Pengujian

Pengujian sistem adalah suatu proses yang dirancang untuk mengevaluasi dan menguji keseluruhan sistem guna memastikan bahwa sistem tersebut berfungsi sesuai dengan spesifikasi dan kebutuhan yang telah ditetapkan sebelumnya. Pengujian meliputi verifikasi sensor, uji koneksi jaringan, dan simulasi kondisi operasional yang beragam. Pengujian juga mengevaluasi respon sistem terhadap perubahan parameter dan kondisi irigasi pertanian. Juga pengujian terhadap aplikasi prediksi menggunakan Support Vector Machine (SVM). Berikut adalah pengujian sistem yang direncanakan dapat dilihat pada Tabel 1.

Tabel 1. Pengujian Sistem IoT

No	Objek Pengujian	Skenario	Hasil yang diharapkan
1	Login sebagai pengguna	Membuka halaman login dan melakukan login dengan memasukkan username dan password pengguna.	Sistem dapat menampilkan halaman login, sistem dapat memverifikasi dan dapat masuk ke dalam dashboard utama.
2	Melihat nilai ketinggian air	Memilih menu ketinggian air dan menampilkan nilainya.	Sistem dapat menampilkan nilai dari ketinggian air.

3	Melihat nilai debit air	Memilih menu debit air dan menampilkan nilainya.	Sistem dapat menampilkan nilai dari debit air.
4	Mencetak laporan	Memilih menu laporan dan dapat mencetak laporan pada setiap kolom	Sistem menampilkan nilai parameter (water level, water flow) pada irigasi, dan beberapa data statistik setiap irigasi yang berfungsi untuk pencetakan laporan.
5	Logout pengguna	Menampilkan halaman logout.	Sistem menampilkan halaman login setelah melakukan proses logout.

Sedangkan rencana pengujian pada aplikasi prediksi debit air pada Tabel 2 berikut

Tabel 2. Pengujian Aplikasi Prediksi

No	Objek Pengujian	Skenario	Hasil yang diharapkan
1	Login sebagai pengguna	Membuka halaman login dan melakukan login dengan memasukkan username dan password pengguna.	Sistem dapat menampilkan halaman login, sistem dapat memverifikasi dan dapat masuk ke dalam dashboard utama.
2	Import / Upload data set	Memilih menu import/upload data set	Sistem dapat import dataset.
3	Menambah dataset curah hujan	Memilih menu tambah dataset curah hujan	Sistem dapat menampilkan menu tambah dataset curah hujan.
4	Mempersiapkan data untuk prediksi	Memilih menu laporan data preparation	Sistem menampilkan menu data preparation
5	Melakukan prediksi	Memilih menu prediksi	Sistem menampilkan menu prediksi dan visualisasinya
6	Logout pengguna	Menampilkan halaman login.	Sistem melakukan proses logout.

3. Hasil dan Pembahasan

Halaman monitoring irigasi ini didesain untuk menampilkan data penting terkait kondisi irigasi secara realtime. Informasi utama yang ditampilkan ada 3 yaitu Suhu, ketinggian air, dan debit air. Suhu saat ini tercatat sebesar 27,0°C, ketinggian air mencapai 217 cm, dan debit air terpantau 0 L/min. Halaman ini juga dilengkapi dengan 2 tombol yaitu, "Pilih Tanggal" yang digunakan untuk menampilkan datepicker dialog dan berfungsi untuk menfilter data download, lalu tombol "Download" yang digunakan untuk mendownload file debit air berdasarkan tanggal

yang dipilih dengan format csv. Di bagian bawah halaman ada menu navigasi dengan 2 pilihan yaitu menuju ke halaman monitoring atau ke halaman history. Pengujian monitoring ditunjukkan pada Gambar 11.



Gambar 11. Halaman Monitoring Irigasi

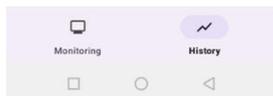
Berikut adalah potongan kode yang digunakan pada halaman monitoring

```
private fun showData() {
    Realtime Database
    DBmonitoring.addValueEventListener(object : ValueEventListener {
        override fun
        onDataChange(dataSnapshot:
        DataSnapshot) {
            val debit =
            dataSnapshot.child("Debit").getValue(Int::class.java)
            val ketinggian =
            dataSnapshot.child("Ketinggian").getValue(Int::class.java)
            val suhu =
            dataSnapshot.child("Suhu").getValue(Double::class.java)
            binding.waterFlowValue.text =
            "${debit ?: 0} L/min"
            binding.waterLevelValue.text =
            "${ketinggian ?: 0} cm"
            binding.temperatureValue.text =
            "${suhu ?: 0.0}°C"
        }
    })
    private fun
    writeDataToCSV(dataList:
    List<Array<String>>) {
        val downloadsDir =
        Environment.getExternalStoragePublic
        lDirectory(Environment.DIRECTOR
        Y_DOWNLOADS)
        val csvFile = File(downloadsDir,
        "irrigation_data.csv")
        try {
            CSVWriter(FileWriter(csvFile)).use { csvWriter ->
```

```

csvWriter.writeAll(dataList)
}
Log.d("CSVWriter", "Data berhasil
disimpan di
${csvFile.absolutePath}")
Toast.makeText(this, "Data
berhasil disimpan di
${csvFile.absolutePath}",
Toast.LENGTH_LONG).show()
} catch (e: IOException) {
Log.e("CSVWriter", "Error menulis
CSV: ${e.message}")
Toast.makeText(this, "Error
menulis CSV: ${e.message}",
Toast.LENGTH_LONG).show()
}
}
    
```

Halaman history pada Gambar 12 merupakan halaman untuk menampilkan data historis terkait kondisi irigasi dalam bentuk grafik. Grafik yang ditampilkan menunjukkan perubahan debit air berdasarkan tanggal tertentu. Halaman ini juga dilengkapi dengan dua tombol, yaitu "Pilih Tanggal" yang digunakan untuk menampilkan datepicker dialog dan berfungsi untuk memfilter data yang akan ditampilkan pada grafik, serta tombol "Reset" yang digunakan untuk mengembalikan tampilan grafik ke pengaturan awal. Di bagian bawah halaman, terdapat menu navigasi dengan dua pilihan, yaitu menuju ke halaman monitoring atau tetap di halaman history.



Gambar 12. Halaman History Irigasi

Berikut adalah source code untuk halaman history.

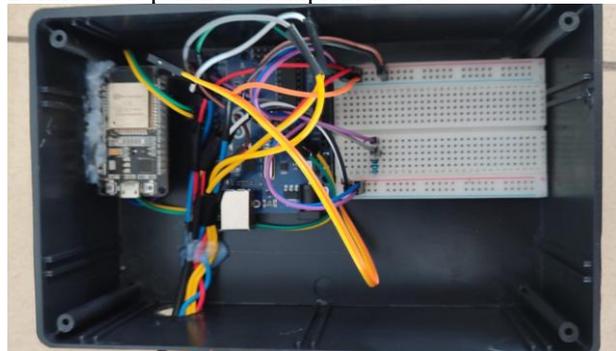
```

private fun fetchDataForDate(date:
String) {
valueEventListener?.let {
database.removeEventListener(it)
}
valueEventListener =
database.child(date).addValueEven
    
```

```

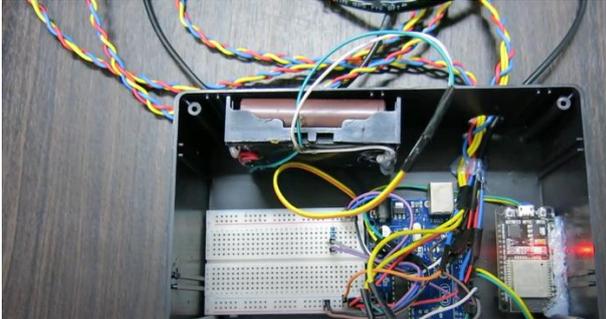
tListener(object :
ValueEventListener {
override fun
onDataChange(snapshot:
DataSnapshot) {
entries.clear()
for (childSnapshot in
snapshot.children) {
val debit =
childSnapshot.child("Debit").getV
alue(Int::class.java) ?: 0
val jam =
childSnapshot.child("Jam").getVal
ue(String::class.java) ?: ""
val timeParts = jam.split(":")
if (timeParts.size == 3) {
try {
val hours = timeParts[0].toFloat()
val minutes =
timeParts[1].toFloat()
val seconds =
timeParts[2].toFloat()
if (hours >= 0 && minutes >= 0 &&
seconds >= 0 && debit >= 0) {
val timeValue = hours + minutes /
60 + seconds / 3600
entries.add(Entry(timeValue,
debit.toFloat()))
} else {
Log.e("HistoryFragment",
"Negative value detected:
hours=$hours, minutes=$minutes,
seconds=$seconds, debit=$debit")
}
} catch (e: NumberFormatException)
{
Log.e("HistoryFragment", "Error
parsing time: $jam", e)
}
} else {
Log.e("HistoryFragment", "Invalid
time format: $jam")
}
}
if (entries.isNotEmpty()) {
updateChart()
}
}
}
    
```

Berikut merupakan hasil implementasi alat



Gambar 13. Tampak atas Alat IoT Monitoring Debit Air Irigasi. Gambar 13, menunjukkan gambar tampak atas dari alat IoT monitoring debit air irigasi yang telah

dikembangkan dengan meliputi microcontroller esp32 dan Arduino, serta 3 sensor yaitu Sensor DS18B20, Sensor Flow Meter YFB10, Sensor Ultrasonik HCSR04. Sedangkan Gambar 14 menunjukkan implementasi alat IoT yang sudah dilengkapi dengan battery.



Gambar 14. Alat Iot Monitoring Debit Air Irigasi yang sudah dilengkapi dengan battery

Berikut merupakan Source kode pada bagian alat-alat IoT untuk mengontrol alat pemantauan irigasi.

Source Kode pada Arduino:

```
// Flow sensor reading
waktuAktual = millis();
if (waktuAktual >= (waktuLoop + 1000)) {
    waktuLoop = waktuAktual;
    literPerjam = (pulsa_sensor * 60 / 7.5);
    pulsa_sensor = 0;
    Serial.print("Flow rate: ");
    Serial.print(literPerjam);
    Serial.println(" L/jam");
}
// Read distance from HC-SR04
long duration, distance;
digitalWrite(TRIGGER_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIGGER_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIGGER_PIN, LOW);
duration = pulseIn(ECHO_PIN, HIGH);
distance = (duration / 2) / 29.1;
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" cm");
// Read temperature from DS18B20
sensor.requestTemperatures();
suhuDS18B20 = sensor.getTempCByIndex(0);
// Send data over SoftwareSerial
String data = String(suhuDS18B20, 1) + "," + String(literPerjam) + "," + String(distance);
mySerial.println(data);
```

Source kode pada esp32

```
def read_uart():
    while True:
        data = ""
        while uart.any():
            data =
            uart.readline().decode('utf-8').strip()
            if data:
                print("Data received:", data)

def process_data(data):
    try:
        temp, flow, dist = data.split(',')
        temp = float(temp)
        flow = int(flow)
        dist = int(dist)
        print("Temperature:", temp, "C")
        print("Flow rate:", flow, "L/m")
        print("Distance:", dist, "cm")
        send_to_firebase(temp, flow, dist)
    except Exception as e:
        print("Failed to process data:", e)

def send_to_firebase(temp, flow, dist):
    url = urlWeb + "irrigation/monitoring.json"
    data = {
        "Debit": flow,
        "Ketinggian": dist,
        "Suhu": temp
    }
    headers = {'Content-Type': 'application/json'}
    try:
        response = urlReq.patch(url, headers=headers, data=ujson.dumps(data))
        print("Data sent to Firebase:", response.json())
        response.close()
    except Exception as e:
        print("Failed to send data to Firebase:", e)
```

Telah dilakukan pengujian fungsionalitas sensor water flow dengan cara dialirkan dengan air dari kran sebelum pengujian di tempat irigasi yang riil, yang ditunjukkan pada Gambar 15.



Gambar 15. Uji Fungsionalitas Sensor Water Flow

Dari hasil pengujian di dapatkan bahwa sensor water flow dapat berjalan dengan baik, dan memberikan

hasil yang kemudian disimpan ke firebase dan ditampilkan dalam aplikasi mobile seperti pada Gambar 16 di bawah ini.



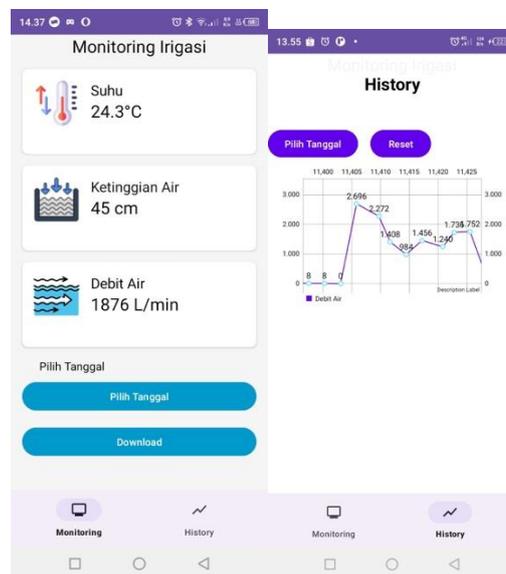
Gambar 16. Hasil Monitoring Fungsionalitas Alat IoT

Pengujian dilakukan di saluran irigasi Bendung Gunting, Sungai Kedak, Kec. Semen, Kabupaten Kediri. Pengujian dilakukan dengan cara menguji sensor secara langsung ke dalam saluran irigasi yang ditunjukkan pada Gambar 17.



Gambar 17. Pengujian Lapangan Alat IoT Pada Saluran Irigasi

Hasil pembacaan sensor water flow ditunjukkan pada Gambar 18 di bawah ini yang ditampilkan ke aplikasi android, meliputi pemantauan suhu, ketinggian air dan debit air.



Gambar 18. Hasil Pembacaan Sensor ditampilkan ke Aplikasi Android

Implementasi Proses Aplikasi Prediksi

Berdasarkan rancangan yang telah dibuat dengan rincian diatas, maka terealisasikan ke dalam sebuah aplikasi perhitungan prediksi dengan bahasa python menggunakan google colaboratory yang berkolaborasi dengan perangkat keras yang dinamakan Internet of Things (IoT)

Library merupakan kumpulan function dan method yang memudahkan dalam melakukan analisis data. Library biasanya mengandung built-in module yang menyediakan fungsi-fungsi yang berbeda yang dapat langsung digunakan.

```
# Library yang Diperlukan
import pandas as pd
from sklearn.model_selection
import train_test_split
from sklearn.linear_model import
LinearRegression
from sklearn.metrics import
mean_absolute_error, r2_score
import matplotlib.pyplot as plt
```

Penjelasan Kode:

1. Library pandas adalah sebuah library python untuk memanipulasi dan analisis data.
2. Scikit-learn adalah library untuk pembelajaran mesin di Python yang menyediakan berbagai algoritma pembelajaran mesin yang sudah siap digunakan, mulai dari regresi, klasifikasi, hingga clustering.
3. Matplotlib adalah library visualisasi data yang memungkinkan pengguna untuk membuat grafik seperti plot garis, histogram, scatter plot, dan lain-lain.

Membuat Data Frame

Memproses data dan debit air agar dapat digunakan dalam model pembelajaran mesin khususnya prediksi. Data yang mencakup waktu (dalam detik) dan debit air agar digunakan dalam model pembelajaran mesin.

```
# Membaca file CSV dan memuatnya ke dalam DataFrame df
df =
pd.read_csv('irrigation_data.csv')

# Mengonversi kolom 'Jam' (yang berformat HH:MM:SS) menjadi total detik
# pd.to_datetime mengubah 'Jam' menjadi objek datetime
# Fungsi lambda digunakan untuk mengonversi jam-menit-detik ke dalam detik total

df['Seconds'] =
pd.to_datetime(df['Jam'],
format='%H:%M:%S').apply(lambda x:
x.hour * 3600 + x.minute * 60 +
x.second)

# Mengambil kolom 'Seconds' sebagai fitur input (X) yang berisi total detik
X = df[['Seconds']]

# Mengambil kolom 'Debit' sebagai target output (y) yang berisi nilai debit air
y = df['Debit']
```

Penjelasan Kode:

1. Membaca data dari file CSV yang berisi kolom waktu (Jam) dan debit air (Debit) ke dalam DataFrame df menggunakan Pandas.
2. Kolom 'Jam' (yang berisi waktu dalam format HH:MM) dikonversi menjadi total detik, sehingga lebih mudah digunakan sebagai input dalam model
3. Mengambil kolom 'Seconds' (total detik) dari DataFrame dan menyimpannya dalam variabel X yang akan digunakan sebagai fitur input untuk model.
4. Mengambil kolom 'Debit' (nilai debit air) dan menyimpannya dalam variabel y, yang akan menjadi target output atau label yang ingin diprediksi oleh model

Pembagian Data Pelatihan dan Pengujian

Kode selanjutnya adalah kode yang digunakan untuk membuat, melatih, dan menguji model dengan menggunakan data yang telah diproses sebelumnya.

```
# Membagi data menjadi set pelatihan dan set pengujian
X_train, X_test, y_train, y_test
= train_test_split(X, y,
test_size=0.2, random_state=42)

# Membuat model regresi linier
model = LinearRegression()
model.fit(X_train, y_train)

# Prediksi nilai pada data pengujian
y_pred = model.predict(X_test)
```

Penjelasan Kode:

1. Membagi data X (input: waktu dalam detik) dan y (output: debit air) menjadi dua bagian, yaitu **set pelatihan** dan **set pengujian**.
 - a. X_train, y_train: Data latih yang akan digunakan untuk melatih model.
 - b. X_test, y_test: Data uji yang akan digunakan untuk menguji performa model setelah dilatih.
 - c. LinearRegression(): Membuat sebuah model **regresi linier** yang digunakan untuk memodelkan hubungan linier antara waktu (detik total) dan debit air.
 - d. model.fit(X_train, y_train): Melatih model menggunakan data pelatihan (X_train, y_train), sehingga model dapat mempelajari pola hubungan antara input (waktu dalam detik) dan output (debit air).
 - e. Menggunakan model yang sudah dilatih untuk **memprediksi** nilai debit air pada **set pengujian** (X_test)
 - f. Hasil prediksi disimpan dalam variabel y_pred, yang berisi nilai debit yang diprediksi oleh model

Parameter:

- a. test_size=0.2 : 20% dari data akan digunakan untuk pengujian (set pengujian), sedangkan 80% digunakan untuk pelatihan (set pelatihan).
- b. random_state=42:memberikan seed untuk pengacakan pembagian data, sehingga hasil pembagian akan konsisten jika kode dijalankan berulang kali.

Evaluasi Model dan Visualisasi Hasil Prediksi

Kode selanjutnya merupakan kode untuk mengevaluasi performa model dan visualisasi hasil prediksi dari model yang membandingkan nilai debit air yang diprediksi dengan nilai debit aktual.

```
# Evaluasi model
mae = mean_absolute_error(y_test,
y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Mean Absolute Error
(MAE): {mae}")
print(f"R-squared: {r2}")

# Plot hasil prediksi vs nilai
aktual
plt.figure(figsize=(10, 6))

# Plot garis untuk nilai aktual
dan prediksi
plt.plot(y_test.values,
label='Actual Debit Air',
marker='o', linestyle='-',
color='blue') # Nilai aktual
plt.plot(y_pred, label='Predicted
Debit Air', marker='x',
linestyle='--',
color='red') # Nilai
prediksi

# Sinkronisasi jam sesuai dengan
y_test
time_labels =
df.loc[y_test.index,
'Jam'].astype(str) # Mengambil
kolom 'Jam' untuk sumbu X

# Menambahkan keterangan jam
sebagai label sumbu x
plt.xticks(ticks=range(len(y_test
)), labels=time_labels,
rotation=45)

# Menambahkan judul dan label
sumbu
plt.title('Actual vs Predicted
Debit Air')
plt.xlabel('Time (Jam)')
plt.ylabel('Debit Air')

# Menampilkan grid, dan plot
plt.legend()
plt.grid(True)
plt.tight_layout() # Agar label
jam tidak terpotong
plt.show()
```

Penjelasan Kode:

1. `mean_absolute_error(y_test, y_pred)` menghitung Mean Absolute Error (MAE), yang merupakan rata-rata perbedaan absolut antara nilai aktual (`y_test`) dan nilai prediksi (`y_pred`)

2. `r2_score(y_test, y_pred)` menghitung R-squared (R^2), yang menunjukkan seberapa baik model regresi mampu menjelaskan variabilitas dalam data. Nilai R^2 berkisar dari 0 hingga 1, dengan nilai mendekati 1 menandakan model yang baik
3. Membuat visualisasi yang menampilkan perbandingan antara nilai debit air aktual (`y_test.values`) dan nilai debit air prediksi (`y_pred`). Mengambil waktu (Jam) yang sesuai dengan data pengujian (`y_test.index`) dari DataFrame, kemudian menggunakannya sebagai label pada sumbu X.

Pengujian Proses

Pengujian proses suatu sistem adalah kegiatan yang dilakukan untuk menilai dan memastikan bahwa semua tahapan dan alur kerja dalam sistem tersebut berfungsi secara optimal sesuai dengan spesifikasi dan tujuan yang telah ditetapkan.

Pengujian Data Frame

Pengujian data frame digunakan untuk menganalisis data dalam bentuk csv menjadi dataframe untuk dijadikan bahan pengujian yang memuat debit, jam, tanggal, second. Pengujian Data Frame ditunjukkan pada Gambar 19.

```
# Mengonversi kolom 'Jam' (yang berformat HH:MM:SS) menjadi total detik
# pd.to_datetime mengubah 'Jam' menjadi objek datetime
# Fungsi lambda digunakan untuk mengonversi jam-menit-detik ke dalam detik total
df['Seconds'] = pd.to_datetime(df['Jam'], format='%H:%M:%S').apply(lambda x: x.hour * 3600 + x.minute * 60 + x.second)

# Mengambil kolom 'Seconds' sebagai fitur input (X) yang berisi total detik
X = df[['Seconds']]

# Mengambil kolom 'Debit' sebagai target output (y) yang berisi nilai debit air
y = df['Debit']
print(df)
```

Debit	Jam	Tanggal	Seconds	
0	1136	8:25:48	18/2/2024	30346
1	1480	8:25:57	18/2/2024	30357
2	400	8:26:07	18/2/2024	30367
3	1184	8:26:19	18/2/2024	30379
4	1224	8:26:29	18/2/2024	30389
...
68	2056	8:53:47	18/2/2024	35627
69	1448	8:53:59	18/2/2024	35639
70	1136	8:54:07	18/2/2024	35647
71	1480	8:54:18	18/2/2024	35658
72	0	8:56:08	18/2/2024	35768

Gambar 19. Pengujian Data Frame

Pengujian Perhitungan Prediksi

Pengujian Perhitungan Prediksi untuk memprediksi nilai debit air pada data pengujian menggunakan model yang telah dilatih (test), dari data pelatihan sekarang menggunakan pengetahuannya untuk memproses data baru (waktu dalam detik, yaitu `X_test`). Hasil dari prediksi ini disimpan dalam variabel `y_pred`, yang berisi perkiraan debit air untuk setiap titik data dalam set pengujian seperti pada Gambar 20.

```
[10] # Membagi data menjadi set pelatihan dan set pengujian
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Membuat model regresi linier
model = LinearRegression()
model.fit(X_train, y_train)

# Prediksi nilai pada data pengujian
y_pred = model.predict(X_test)
print(y_pred)
```

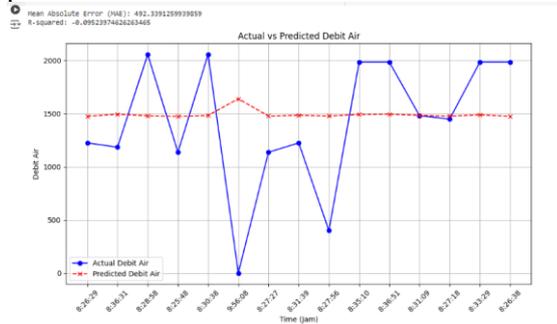
```
[1474.69661258 1493.05761918 1479.24111421 1473.44611213 1482.29111531
1638.75617162 1476.46561321 1484.15161598 1477.35011353 1490.58711829
1493.6676194 1483.23661565 1476.19111311 1487.50661719 1474.97111268]
```

Gambar 20. Pengujian Perhitungan Prediksi

Pengujian Evaluasi Model dan Visualisasi

Pengujian Evaluasi Model dan Visualisasi berguna untuk mengevaluasi model hasil prediksi yang dibandingkan dengan nilai aktual debit air yang dilakukan menggunakan Mean Absolute Error (MAE) dan R-squared (R^2). MAE mengukur rata-rata kesalahan absolut antara nilai aktual (y_{test}) dan nilai yang diprediksi (y_{pred}), dengan nilai yang lebih kecil menunjukkan prediksi yang lebih akurat.

Setelah evaluasi hasil prediksi divisualisasikan dalam bentuk plot, satu mewakili nilai debit air yang sebenarnya (dengan garis biru dan marker bulat), dan satu lagi mewakili nilai prediksi dari model (dengan garis merah dan marker silang). Dari data pembacaan sensor diperoleh debit air yang terbaca antara 400 – 2112 liter/menit, dan diperoleh hasil prediksi di kisaran 1500 liter/menit, ditunjukkan pada Gambar 21 di bawah ini.



Gambar 21. Hasil Prediksi Debit Air

Tabel 3. Hasil Pengujian Sistem IoT

No	Objek Pengujian	Skenario	Hasil
1	Login sebagai pengguna	Membuka halaman login dan melakukan login dengan memasukkan username dan password pengguna.	Sesuai
2	Melihat nilai ketinggian air	Memilih menu ketinggian air dan menampilkan nilainya.	Sesuai
3	Melihat nilai debit air	Memilih menu debit air dan menampilkan nilainya.	Sesuai
4	Mencetak laporan	Memilih menu laporan dan dapat mencetak laporan pada setiap kolam	Sesuai
5	Logout pengguna	Menampilkan halaman logout.	Sesuai

Sedangkan rhasil pengujian pada aplikasi prediksi debit air pada Tabel 4 berikut

Tabel 4. Hasil Pengujian Aplikasi Prediksi

No	Objek Pengujian	Skenario	Hasil
1	Login sebagai pengguna	Membuka halaman login dan melakukan	Sesuai

		login dengan memasukkan username dan password pengguna.	
2	Import / Upload data set	Memilih menu import/upload data set	Sesuai
3	Menambah dataset curah hujan	Memilih menu tambah dataset curah hujan	Sesuai
4	Mempersiapkan data untuk prediksi	Memilih menu laporan data preparation	Sesuai
5	Melakukan prediksi	Memilih menu prediksi	Sesuai
6	Logout pengguna	Menampilkan halaman login.	Sesuai

4. Kesimpulan

Kesimpulan pada penelitian berjudul “Prediksi Debit Air Pada Irigasi Pertanian Menggunakan Support Vector Machine Dan Pemantauan Kondisinya Menggunakan Platform IoT” ini adalah: Telah berhasil Merancang dan membangun sistem monitoring debit air pada irigasi pertanian berbasis platform Internet of Things menggunakan sensor water flow, sensor ultrasonic HCSR04 dan sensor suhu ds18b20, dilengkapi dengan aplikasi monitoring berbasis android untuk memonitor dan mengambil data history debit air. Aplikasi prediksi debit air pada irigasi pertanian juga telah diimplementasikan menggunakan support vector machine terutama menggunakan linier regression dan diperoleh hasil prediksi debit air dengan kondisi saat pemantauan berada pada kisaran 1500 liter/menit atau 25 liter per detik. Namun kondisi saluran irigasi saat beberapa kali pengujian lapangan belum optimal karena volume air irigasi hanya berkisar di 30% kapasitas saluran irigasi dikarenakan musim kemarau. Untuk perbaikan dari penelitian ini diperlukan implementasi metode Deep Learning seperti Long Short Term Memory (LSTM) untuk perbaikan hasil prediksi debit air.

5. Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada P3M Politeknik Negeri Malang yang telah memberikan support terlaksananya penelitian ini. Kegiatan Penelitian ini dibiayai dengan dana DIPA Nomor SP DIPA – 023.18.2.677606/2024, Politeknik Negeri Malang.

Daftar Pustaka:

A. Taufik, A. F. (2023). Sistem Monitoring pH dan Kekeruhan Kolam Ikan Koi Berbasis Internet of Things Menggunakan Aplikasi Blynk. *Jurnal Teknik Elektro*, 56-66.
 A.F. Sallaby, F. U. (2015). Aplikasi Widget Berbasis Java. *Jurnal Media Infotama*, 171-180.
 A.Y. Permana, P. R. (2019). PERANCANGAN SISTEM INFORMASI PENJUALAN

- PERUMAHAN MENGGUNAKAN METODE SDLC PADA PT. MANDIRI LAND PROSPEROUS BERBASIS MOBILE. SIGMA - Jurnal Teknologi Pelita Bangsa.
- Arief Selay, G. D. (2022). Internet of Things. Karimah Tauhid, 860-868.
- Aulia Putri, C. S. (2023). Komparasi Algoritma K-NN, Naive Bayes dan SVM untuk Prediksi Kelulusan Mahasiswa Tingkat Akhir. MALCOM: Indonesian Journal of Machine Learning and Computer Science, 20-26.
- B.P Nugroho, A. S. (2023). SISTEM MONITORING DAN CONTROLLING ALAT PENGURAS AIR OTOMATIS PADA KOLAM LOBSTER BERBASIS IOT (STUDI KASUS GAS FARM BLITAR). Jurnal Mahasiswa Teknik Informatika.
- Barbará, D. (2002). Applications of Data Mining in Computer Security. Kluwer Academic Publishers.
- BPS. (2023). Luas Panen dan Produksi Padi di Indonesia 2023 (Angka Sementara). Jakarta: BPS.
- Chauhan, N. S. (2020, 04 1). An introduction to the DBSCAN algorithm and its Implementation in Python. Retrieved from <https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html>
- Christ Memory Sitorus, A. R. (2020). Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine. Jurnal Teknik Informatika dan Sistem Informasi, 254-265.
- Cynet. (2019, 09 29). EDR Security and Protection for the Enterprise. Retrieved from Cynet: <https://www.cynet.com/endpoint-protection-and-edr/top-6-edr-tools-compared/>
- Education, I. C. (2020, 09 21). Unsupervised Learning. Retrieved from <https://www.ibm.com/cloud/learn/supervised-learning>
- Emy Haryatmi, S. P. (2021). Penerapan Algoritma Support Vector Machine Untuk Model Prediksi Kelulusan Mahasiswa Tepat Waktu. Jurnal RESTI Rekayasa Sistem dan Teknologi Informasi, 386-392.
- Gugun Gunawan, E. S. (2022). SISTEM MONITORING BENDUNGAN AIR BERBASIS INTERNET OF THINGS. PROSIDING SEMINAR NASIONAL NCIET (pp. 103-114). Semarang: Jurusan Teknik Mesin Politeknik Negeri Semarang.
- Habibah br Lumbantobing, R. (2023). Prediksi Harga Cryptocurrency Menggunakan Algoritma Support Vector Machine. Innovative: Journal of Social Science Research, 7348-7355.
- Krismiando. (2013). Pengembangan model estimasi curah hujan di Jawa berbasis data indeks Monsun, SOI, dan DMI. Bandung: CV. Andira.
- M. Kantardzic, J. W. (2003). Data Mining: Concepts, Models, methods, and Algorithms.
- Muhamad Maulana Syahaddan, A. F. (2023). Sistem Pemantauan Sawah Mobile Berbasis Internet of Things . KLIK: KAJIAN ILMIAH INFORMATIKA DAN KOMPUTER, 1369-1380.
- Sri Astutik, D. S. (2021). Rehabilitasi Jaringan Irigasi Untuk Peningkatan Produksi Pertanian. Seminar Keinsinyuran Program Studi Program Profesi Insinyur (PSPPI) Tahun 2021 (pp. 139-146). Malang: Universitas Muhammadiyah Malang.
- T. Ardyanto, A. P. (2017). Pembuatan Game 2D Petualangan Hanoman Berbasis Android. Jurnal Ilmiah Go Infotech, 14-17.

Halaman ini sengaja dikosongkan