

Penerapan Metode Regresi Linear Pada Sistem Peringatan Dini Banjir Berbasis Internet of Things (IoT)

Nugra Zurus Pratama*, Tedy Rismawan, Suhardi

Fakultas MIPA, Program Studi Rekayasa Sistem Komputer, Universitas Tanjungpura, Pontianak, Indonesia
Email: ^{1,*}nugrazurus@student.untan.ac.id, ²tedyrismawan@siskom.untan.ac.id ³suhardi@siskom.untan.ac.id
Email Penulis Korespondensi : nugrazurus@student.untan.ac.id
Submitted 15-09-2022; Accepted 19-10-2022; Published 31-10-2022

Abstrak

Banjir dapat memberikan dampak negatif seperti genangan air yang masuk ke rumah bagi yang terdampak. Oleh karena itu, dibutuhkan suatu sistem untuk memonitor cuaca dan dapat memberikan peringatan dini banjir. Pada penelitian ini dibuat sistem monitoring cuaca dan peringatan dini banjir ini berbasis *Internet of Things* dengan menerapkan metode Regresi Linear. Sistem terdiri dari *node sensor* cuaca, *node sensor* tinggi air, dan perangkat lunak berupa *website*. Sistem mengukur curah hujan, suhu udara, kelembaban udara, dan tinggi air. Proses pengiriman data dari sensor ke server menggunakan ESP32 sebagai mikrokontroler yang terhubung ke jaringan *wifi* dan internet. Sistem akan mengirimkan notifikasi apabila ketinggian air di atas level normal. Berdasarkan hasil pengujian didapatkan sebanyak 45 kejadian curah hujan. Persentase keberhasilan prediksi ketinggian air menggunakan metode regresi linear mendapatkan persentase sebesar 94,4% dengan nilai *error* sebesar 5,6%.

Kata Kunci: Banjir; Hujan; Peringatan Dini; Internet of Things; Regresi Linear

Abstract

Flood is an event when water inundates an area that is usually not flooded for a certain period of time. Floods usually occur because of continuous rainfall and result in overflow of river. Floods can cause negative impact such as puddles of water that enter homes of those affected. Therefore, we need a system to monitor the weather and can provide flood early warning. In this study the weather monitoring system and flood early warning were made based on internet of things by applying linear regression methods. The system consists of a weather sensor node, a water level sensor node and software in the form of a website. The system measures rainfall, air temperature, humidity, and water level. The process of sending data from the sensor to the server uses ESP32 as a microcontroller which is connected to a wifi network and internet. The system will send a notification if the water level is above the normal level. Based on the test results obtained as many as 45 occurrences of rainfall. The percentage of success in predicting water levels using the linear regression method is 94,4% with an error value of 5,6%.

Keywords: Flood; Rainfall; Early Warning; Internet of Things; Linear Regression

1. PENDAHULUAN

Banjir merupakan peristiwa ketika air menggenangi suatu wilayah yang biasanya tidak digenangi air dalam jangka waktu tertentu. Banjir biasanya terjadi karena curah hujan turun terus menerus dan mengakibatkan meluapnya air sungai, danau, laut atau drainase karena jumlah air yang melebihi daya tampung media penopang air dari curah hujan tadi. Kejadian banjir sangat bersifat lokal. Satu daerah bisa terlanda banjir dan daerah lainnya aman. Banjir juga terkadang dapat meluas dan melumpuhkan kehidupan perkotaan seperti yang pernah terjadi di Jakarta [1].

Banjir disebabkan oleh beberapa hal salah satunya karena air pasang. Menurut Wali Kota Pontianak Edi Rusdi Kamtono, air pasang laut yang mengalir ke Sungai Kapuas pada tanggal 2 hingga 3 Desember 2019 mencapai ketinggian 1,7 meter atau di atas normal. Data tersebut diperoleh dari BMKG Maritim Pontianak, pada tanggal 2 hingga 3 Desember 2019. Ia menambahkan sekitar 55 persen daerah Kota Pontianak rawan tergenang di musim hujan yang disertai air pasang tertinggi di Sungai Kapuas [2].

Hujan deras juga dapat mengakibatkan banjir. Menurut kumparan.com terjadi hujan deras yang mengguyur Kota Pontianak pada Kamis 23 Januari 2020 pukul 14.00 WIB hingga pukul 15.00 WIB. Hujan deras ini mengakibatkan beberapa ruas jalan besar yang biasa ramai dilewati oleh pengendara terendam banjir. Rumah warga yang berada dipinggir jalan juga terkena dampak banjir. Salah seorang warga mengatakan rumahnya terendam air dengan ketinggian sekitar 10 sentimeter [3].

Regresi linear merupakan analisis statistika yang memodelkan hubungan beberapa variabel menurut bentuk hubungan persamaan linear eksplisit. Persamaan linear bentuk eksplisit adalah persamaan linear yang menempatkan suatu peubah secara tunggal pada salah satu persamaan. Dalam metode regresi teknik analisis statistika yang digunakan menggambarkan hubungan antara satu variabel dependen dengan satu atau lebih variabel independen. Variabel dependen adalah variabel yang akan diestimasi nilainya, biasanya di plot pada sumbu tegak (sumbu y). Variabel independen adalah variabel yang diasumsikan berpengaruh terhadap variabel terikat, biasanya plot pada sumbu datar (sumbu x). Semua model regresi selalu dituliskan sebagai persamaan yang menghubungkan variabel dependen dan variabel independen [4].

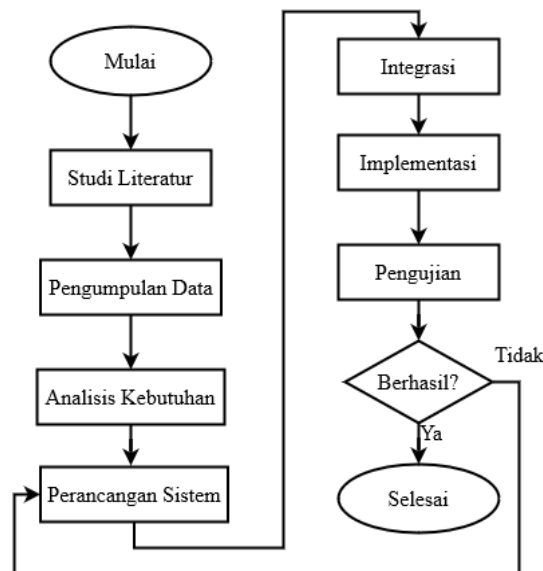
Beberapa penelitian sudah pernah dilakukan untuk monitoring dan peringatan dini banjir, penelitian yang pertama memonitoring cuaca dan mendeteksi banjir menggunakan modul ESP8266 serta dengan beberapa sensor yaitu sensor kecepatan angin, sensor curah hujan, sensor suhu udara, sensor tekanan udara, sensor kelembapan udara, dan sensor jarak. Masukan dari penelitian tersebut berupa sensor Anemometer, sensor *Wind Vane*, suhu dan kelembapan, dan *rotary switch*. Data yang ditampilkan pada penelitian ini melalui android [5]. Penelitian berikutnya mengukur satuan-satuan cuaca yaitu suhu udara, kelembapan, intensitas cahaya, tekanan udara arah angin, dan kecepatan angin. Masukan dari penelitian

adalah sensor intensitas cahaya, sensor kecepatan angin, dan sensor arah angin. Sedangkan keluarannya adalah nilai arah angin, nilai kecepatan angin, dan nilai intensitas cahaya [6]. Penelitian berikutnya mendeteksi dan memonitoring tinggi air dengan menggunakan mikrokontroler arduino mega 2560 dan sensor ultrasonik HC-SR04. Masukan penelitian berupa sensor ultrasonik berbasis *website* dan mempuntai keluaran berupa SMS kondisi ketinggian air, pengiriman peringatan API Facebook [7]. Penelitian berikutnya menggunakan metode Regresi Linear untuk melakukan prediksi kelas yang akan dibuka setiap tahunnya. Variabel yang digunakan untuk prediksi yaitu periode dan jumlah mahasiswa [8].

Dari beberapa penelitian yang telah dipaparkan, maka selanjutnya akan dilakukan penelitian dengan judul “Penerapan Metode Regresi Linear pada Sistem Monitoring Dini Banjir Berbasis *Internet of Things* (IoT)”. Penelitian ini membuat sistem yang dapat memonitoring cuaca dan memberikan peringatan dini banjir dengan menerapkan metode regresi linear. Sistem terdiri dari perangkat keras dan perangkat lunak, perangkat keras akan mengukur satuan-satuan cuaca dan tinggi permukaan air dengan menggunakan beberapa sensor, data yang diperoleh kemudian akan dikirim ke server melalui internet dan selanjutnya akan diproses oleh perangkat lunak. Data yang telah diproses akan ditampilkan melalui media *website*.

2. METODOLOGI PENELITIAN

Penelitian yang berjudul penerapan metode regresi linear pada sistem peringatan dini banjir berbasis *internet of things* (IoT), terdapat tujuh tahapan yaitu studi literatur, metode pengumpulan data, analisis kebutuhan, perancangan sistem, integrasi, implementasi, dan pengujian. Berikut merupakan tahapan metode penelitian terdapat pada Gambar 1.



Gambar 1. Diagram Alir Penelitian

2.1 Studi Literatur

Pada tahapan ini dilakukan pengumpulan literatur atau referensi untuk menunjang penelitian. Literatur yang digunakan dapat berupa buku, jurnal ilmiah, *datasheet*, dan *website*.

2.2 Metode Pengumpulan Data

Dalam penelitian ini digunakan metode observasi sebagai metode pengumpulan data. Metode observasi merupakan suatu metode yang digunakan untuk mengamati dan mencatat secara sistematis terhadap gejala yang tampak pada objek penelitian.

2.3 Analisis Kebutuhan

Tahap analisis kebutuhan merupakan tahap untuk menganalisis kebutuhan dari penelitian atau sistem yang akan dibangun. Dalam penelitian ini analisis kebutuhan dibagi menjadi dua yaitu kebutuhan perangkat keras dan kebutuhan perangkat lunak.

2.3.1 Kebutuhan Perangkat Keras

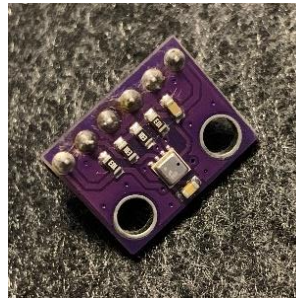
- ESP32 adalah *system on chip* berdaya rendah pada mikrokontroler yang terintegrasi dengan Wi-Fi dan dual-mode Bluetooth. ESP32 menggunakan mikroprosesor Tensilica Stensa LX6 baik dalam variasi dual-core dan single-core dilengkapi dengan antena built-in, RF balun, penguat daya, filter, dan modul manajemen daya. ESP32 dibuat dan dikembangkan oleh Espressif Systems, sebuah perusahaan Cina yang berbasis di Shanghai, dan diproduksi oleh TSMC menggunakan proses 40nm. ESP32 memiliki 18 ADC (Analog to Digital Converter), 2 DAC, 16 PWM, 10

Sensor sentuh, 2 jalur antarmuka UART, pin antarmuka I2C, I2S, dan SPI [9]. Pada penelitian ini ESP32 digunakan sebagai pengolah data yang diperoleh dari sensor. Selain itu ESP32 juga berfungsi untuk mengirimkan data yang diperoleh dari sensor ke *database* melalui internet.



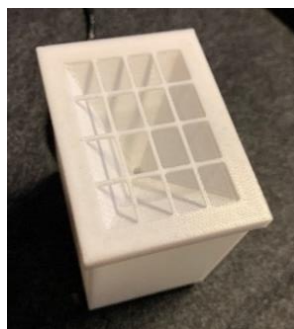
Gambar 1. ESP32

- b. Sensor BME280: Sensor BME280 adalah sensor gabungan kelembaban, tekanan dan sensor suhu berdasarkan prinsip penginderaan. Sensor ini mengeluarkan data penginderaan berupa data digital. Dimensinya yang kecil dan konsumsi daya yang rendah memungkinkan untuk diimplementasikan pada perangkat yang digerakkan dengan baterai handset, modul GPS, atau jam tangan [10]. Pada penelitian ini sensor BME280 digunakan sebagai pengindra suhu udara, kelembaban udara, dan tekanan udara.



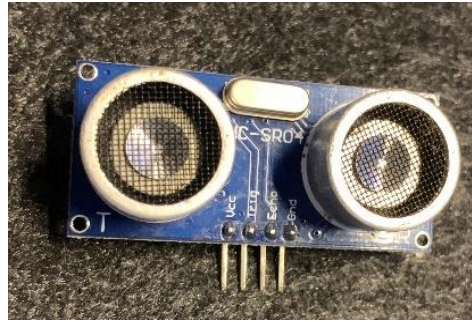
Gambar 2. Sensor BME280

- c. Sensor Curah Hujan (*Tipping Bucket*): Sensor curah hujan adalah sensor yang berfungsi sebagai pengukur curah hujan. Sensor curah hujan ini bertipe *tipping bucket*. Sensor jenis *tipping bucket* bekerja dengan cara menghitung pulsa persatuan waktu yang ditentukan dari banyaknya air yang masuk ke dalam corong sensor tersebut. Di dalam sensor hujan terdapat dua buah bejana yang saling bergantian menampung air hujan. Tiap gerakan bejana berjungkit secara mekanis bergerak melewati sensor *hall* sehingga menghasilkan pulsa digital. Sensor ini menggunakan suplai daya DC 5 volt. Sensor ini memiliki keluaran data digital *low* atau *high* [11]. Pada penelitian ini sensor curah hujan digunakan untuk mengukur debit curah hujan.



Gambar 3. Sensor Curah Hujan

- d. Sensor Ultrasonik HC-SR04: Sensor ultrasonik adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Pada sensor ultrasonik, gelombang ultrasonik dibangkitkan melalui sebuah alat yang disebut dengan piezoelektrik dengan frekuensi tertentu. Piezoelektrik ini akan menghasilkan gelombang ultrasonik (umumnya berfrekuensi 40kHz) ketika sebuah osilator diterapkan pada benda tersebut. Secara umum, alat ini akan menembakkan gelombang ultrasonik menuju suatu area atau suatu target. Setelah gelombang menyentuh permukaan target, maka target akan memantulkan kembali gelombang tersebut. Gelombang pantulan dari target akan ditangkap oleh sensor, kemudian sensor menghitung selisih antara waktu pengiriman gelombang dan waktu gelombang pantul diterima [12]. Pada penelitian ini sensor ultrasonik digunakan untuk mengukur jarak antara sensor dengan permukaan air.



Gambar 4. Sensor HC-SR04

2.3.2 Kebutuhan Perangkat Lunak

- Firestore Realtime Database* adalah *database* yang di-host di *cloud*. Data disimpan sebagai JSON dan disinkronkan secara *realtime* ke setiap klien yang terhubung. Ketika mem-build aplikasi lintas platform dengan SDK iOS, Android, dan JavaScript, semua klien akan berbagi sebuah *instance Realtime Database* dan menerima *update* data terbaru secara otomatis [13]. *Realtime Database* adalah *database* NoSQL, sehingga memiliki pengoptimalan dan fungsionalitas yang berbeda dengan *database* terkait. *API Realtime Database* dirancang agar hanya mengizinkan operasi yang dapat dijalankan dengan cepat. Hal ini memungkinkan Anda untuk mem-build pengalaman *realtime* yang luar biasa dan dapat melayani jutaan pengguna tanpa mengorbankan kemampuan respons. Oleh karena itu, perlu dipikirkan bagaimana pengguna mengakses data, kemudian buat struktur data sesuai dengan kebutuhan tersebut.
- Arduino IDE adalah *software* buatan Arduino yang berisi editor teks untuk menulis kode, area pesan, konsol teks, bilah alat dengan tombol untuk fungsi umum dan serangkaian menu. Arduino IDE terhubung ke perangkat keras Arduino untuk mengunggah program dan berkomunikasi dengan perangkat keras tersebut. Program yang ditulis menggunakan Arduino IDE disebut sketsa. Sketsa ini ditulis dalam editor teks dan disimpan dengan ekstensi file .ino. Editor memiliki fitur untuk memotong/menempel dan untuk mencari/mengganti teks. Area pesan memberikan umpan balik saat menyimpan dan mengeksport dan juga menampilkan kesalahan. Konsol menampilkan output teks oleh Arduino IDE, termasuk pesan kesalahan lengkap dan informasi lainnya. Sudut kanan bawah jendela menampilkan papan yang dikonfigurasi dan port serial. Tombol toolbar memungkinkan Anda untuk memverifikasi dan mengunggah program, membuat, membuka, dan menyimpan sketsa, dan membuka monitor serial [14]. Pada penelitian ini Arduino IDE digunakan untuk menulis dan mengunggah program ke ESP32.
- Visual Studio Code (VS Code) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code (seperti C++, C#, Python, Go, Java, dst) [15]. Pada penelitian ini Visual Studio Code digunakan untuk menulis kode program website.
- Nodejs adalah open-source, cross-platform runtime environment yang digunakan untuk pengembangan aplikasi web server-side. Aplikasi Nodejs ditulis dalam JavaScript dan dapat dijalankan pada berbagai macam sistem operasi. Nodejs didasarkan pada arsitektur event-driven dan non-blocking Input/Output API yang dirancang untuk mengoptimalkan skalabilitas untuk aplikasi web realtime. Selama ini, kerangka kerja yang tersedia untuk pengembangan web semuanya didasarkan pada model stateless. Model stateless adalah di mana data yang dihasilkan dalam satu sesi tidak dipertahankan untuk penggunaan di sesi berikutnya dengan pengguna tersebut. Banyak pekerjaan yang harus dilakukan untuk menjaga informasi sesi antara permintaan untuk pengguna. Tetapi dengan Nodejs akhirnya ada cara bagi aplikasi web untuk memiliki koneksi dua arah secara realtime, di mana klien dan server menjalankan komunikasi secara bersamaan, memungkinkan mereka untuk bertukar data secara bebas [16]. Pada penelitian ini NodeJS digunakan untuk menjalankan aplikasi website.

2.4 Implementasi

Pada tahap implementasi akan merealisasikan hal-hal yang dilakukan pada tahap perancangan sistem, sehingga menghasilkan keluaran yang diinginkan. Proses dari implementasi sistem dimulai dari perakitan komponen-komponen yang digunakan pada *node* sensor cuaca dan *node* sensor tinggi air, kemudian membuat kode program untuk *node* sensor cuaca dan *node* sensor tinggi air, selanjutnya membuat *realtime database* di Firebase, dan membuat antarmuka *website*

2.5 Internet of Things (IoT)

Internet of Things (IoT) menurut Rekomendasi ITU-T Y.2060 didefinisikan sebagai sebuah penemuan yang mampu menyelesaikan permasalahan yang ada melalui penggabungan teknologi dan dampak sosial. Jika ditinjau dari standarisasi secara teknik, IoT dapat digambarkan sebagai infrastruktur global untuk memenuhi kebutuhan informasi masyarakat, memungkinkan layanan canggih dengan interkoneksi baik secara fisik dan virtual berdasarkan pada yang telah ada dan perkembangan informasi serta teknologi komunikasi (ICT) [17].

2.6 Regresi Linear

Regresi linear merupakan analisis statistika yang memodelkan hubungan beberapa variabel menurut bentuk hubungan persamaan linear eksplisit. Dalam metode regresi teknik analisis statistika yang digunakan menggambarkan hubungan antara satu variabel dependen dengan satu atau lebih variabel independen [4]. Terdapat variabel dependen yang dilambangkan sebagai Y dan variabel independen dilambangkan sebagai X . *intercept* (konstanta) dilambangkan dengan β_0 , sedangkan *error* dilambangkan dengan e . Berikut model matematis regresi linear dinyatakan pada Persamaan 1 [18].

$$Y = \beta_0 + \beta X_1 + e \quad (1)$$

Keterangan:

Y = variabel dependen

β_0 = *intercept*

X = variabel independen

e = *error*

Regresi linear atau dapat disebut dengan regresi linear sederhana mempunyai variabel dependen yang dilambangkan sebagai Y dan variabel independen dilambangkan sebagai x . kemudian konstanta (*intersep*) dilambangkan sebagai a dan konstanta regresi dilambangkan dengan b yang terdapat bentuk umum yang dinyatakan pada Persamaan 2 [4].

$$Y = a + bx \quad (2)$$

Keterangan:

Y = variabel dependen

a = konstanta (*intersep*)

b = konstanta regresi (*slope*)

x = variabel independen

2.7 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) merupakan persentase *error* rata-rata mutlak. MAPE digunakan untuk menunjukkan rata-rata kesalahan peramalan dalam bentuk persentase terhadap data aktual [19]. Pada perhitungan ini, nilai *Absolute Percentage Error* (APE) dicari terlebih dahulu menggunakan nilai aktual dilambangkan dengan X_i dan nilai perkiraan dilambangkan dengan F_i seperti pada Persamaan 3.

$$APE = \frac{|X_i - F_i|}{X_i} \times 100\% \quad (3)$$

Setelah nilai APE diperoleh, maka akan dilakukan perhitungan nilai MAPE seperti pada Persamaan 4.

$$MAPE = \frac{\sum APE_i}{n} \quad (4)$$

Keterangan:

APE = *Absolute Percentage Error* atau nilai kesalahan absolut

X_i = nilai aktual

F_i = nilai perkiraan

n = banyaknya data

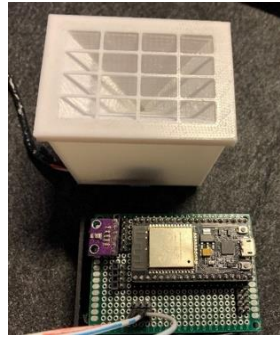
2.8 Pengujian

Tahap pengujian dilakukan untuk menguji keberhasilan dari tahap perancangan sistem yang telah dilakukan sebelumnya. Pengujian dilakukan terhadap perangkat keras dan perangkat lunak. Tahapan pengujian dilakukan mulai dari menguji komponen pemrosesan utama dengan masing-masing sensor, kemudian menguji proses pengiriman dari *node* sensor cuaca dan *node* sensor tinggi air ke *realtime database* Firebase, selanjutnya menguji proses penerimaan dari *realtime database* Firebase ke *website*, dan menguji keseluruhan sistem yang dibuat.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Node Sensor Cuaca

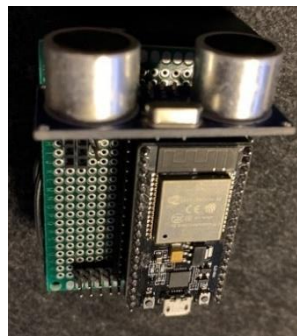
Node sensor cuaca merupakan bagian dari sistem yang berfungsi untuk melakukan pemantauan terhadap kondisi cuaca yang terdiri dari suhu udara, kelembaban udara, tekanan udara, dan curah hujan. Komponen pada *node* sensor cuaca merupakan gabungan dari ESP32, sensor BME280, dan *sensor* curah hujan. ESP32 berfungsi sebagai komponen yang melakukan pengolahan data yang didapat dari *sensor*. Hasil pemantauan dari *node* sensor cuaca bersifat *realtime*. Komponen-komponen *node* sensor cuaca dapat dilihat pada gambar 5.



Gambar 5. Implementasi Node Sensor Cuaca

3.2 Implementasi Node Sensor Tinggi Air

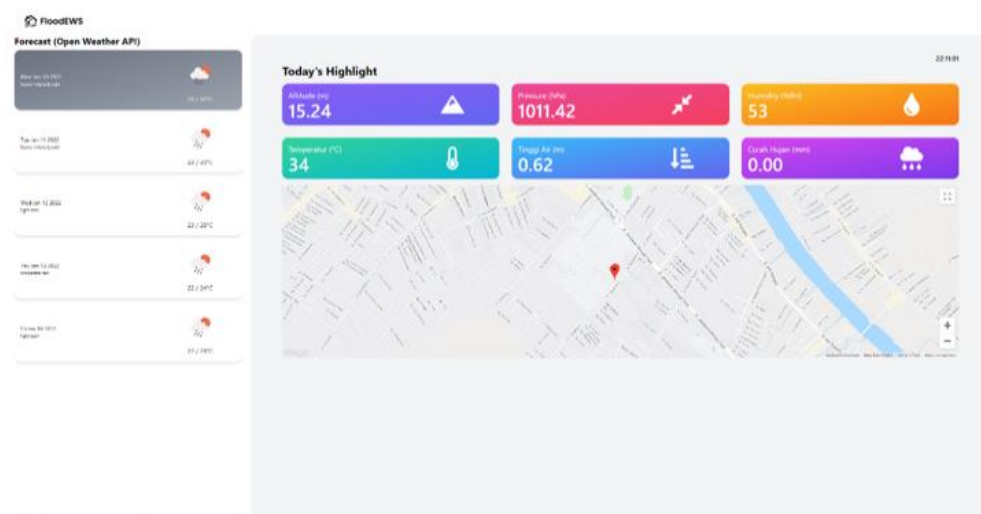
Node sensor tinggi air merupakan bagian dari sistem yang berfungsi untuk melakukan pemantauan terhadap kondisi tinggi air. *Node* sensor tinggi air merupakan gabungan dari ESP32 dan sensor HC-SR04. Sensor HC-SR04 terhubung ke ESP32 berfungsi untuk mendeteksi ketinggian air. Hasil pemantauan dari *node* sensor tinggi air bersifat *realtime*. Hasil implementasi *node* sensor tinggi air dapat dilihat pada Gambar 6.



Gambar 6. Implementasi Node Sensor Tinggi Air

3.3 Implementasi Antarmuka Website

Antarmuka *website* bertujuan untuk memberikan informasi kepada pengguna dengan antarmuka yang interaktif. Halaman pertama yang tampil adalah halaman beranda. Tampilan pada halaman beranda memperlihatkan nilai-nilai dari *node* sensor cuaca dan *node* sensor tinggi air. Halaman ini juga memperlihatkan letak dari *sensor* yang digunakan dalam peta. Tampilan dari halaman beranda dapat dilihat pada Gambar 7.



Gambar 7. Halaman Beranda

Halaman Beranda menampilkan sejumlah informasi mengenai cuaca dan tinggi air yang diambil oleh sensor, prakiraan cuaca yang diambil melalui *Open Weather API* dan lokasi letak sensor. Informasi cuaca yang ditampilkan yaitu tekanan udara, kelembaban udara, suhu udara, tinggi air, dan curah hujan. Jika pengguna menekan salah satu tab info dari sensor maka pengguna akan masuk ke halaman grafik dan tabel. Pada halaman ini pengguna dapat melihat data historis dari sensor yang dipilih. Tampilan dari halaman grafik dan tabel dapat dilihat pada Gambar 8.



Gambar 8. Halaman Tabel dan Grafik

Halaman Tabel dan Grafik menampilkan informasi yang direkam oleh sensor per 30 menit. Data yang disajikan dalam bentuk tabel dan grafik hanya curah hujan dan tinggi air. Pada tabel dan grafik data yang ditampilkan berupa waktu dan ketinggian air atau debit curah hujan.

3.4 Perhitungan Regresi Linear

Perhitungan ini dilakukan menggunakan metode Regresi Linear dan diimplementasikan dalam aplikasi untuk memprediksi kenaikan permukaan air pada saat terjadi hujan. Perhitungan ini menggunakan data yang diambil dari *node* sensor cuaca dan *node* sensor tinggi air sehingga didapat sebanyak 67 kejadian hujan yang terdapat pada Tabel 1.

Tabel 1. Data Curah Hujan dan Tinggi

No.	Curah Hujan (X)	Tinggi Air (Y)	X.Y	X ²	Y ²
1.	0,53	41	21,73	0,2809	1681
2.	0,053	2	0,106	0,002809	4
3.	0,053	12	0,636	0,002809	144
4.	0,053	2	0,106	0,002809	4
5.	0,053	3	0,159	0,002809	9
...
63.	0,318	47	14,946	0,101124	2209
64.	0,159	9	1,431	0,025281	81
65.	0,159	6	0,954	0,025281	36
66.	0,053	2	0,106	0,002809	4
67.	0,159	11	1,749	0,025281	121
n = 67	ΣX = 8,215	ΣY = 552	ΣXY = 140,662	ΣX ² = 1,845	ΣY ² = 12128

Tabel ini merupakan data latih yang akan digunakan untuk mencari nilai dari konstanta a dan b pada persamaan 2. Variabel-variabel yang digunakan dalam perhitungan yaitu curah hujan (X) dan Tinggi Air (Y). Dari tabel di atas didapat hasil perhitungan ΣX sebesar 8,125, ΣY sebesar 552, ΣXY sebesar 140,662, ΣX² sebesar 1,845, dan ΣY² sebesar 12128.

3.5 Pengujian Sistem

Pengujian ini merupakan gabungan dari keseluruhan pengujian yang berguna untuk memastikan apakah sistem dapat bekerja sesuai dengan yang diharapkan. Pada pengujian ini menguji apakah data dari *node* sensor cuaca dan *node* sensor tinggi air terhubung dengan *website* serta dapat menampilkan *push notification*. Hasil dari pengujian sistem ini dapat dilihat pada Tabel 2.

Tabel 2. Pengujian Keseluruhan Sistem

No.	Curah Hujan (Inch)	Kelembaban (%)	Suhu (°C)	Tinggi Air (cm)	Hasil Prediksi (cm)	Nilai Aktual (cm)	APE (%)	Kategori	Notifikasi	Tanggal dan Waktu
1.	0,106	44,87	36,46	87	93,84	88	6,64	Waspada	Terkirim	2/4/2022, 10:30
2.	0,106	61,65	31,15	84	90,84	89	2,07	Waspada	Terkirim	2/4/2022, 16:00
3.	0,106	62,1	30,3	89	95,84	92	4,17	Waspada	Terkirim	2/4/2022, 16:30
4.	0,106	51,02	33,66	55	61,84	55	12,44	Rendah	Tidak Terkirim	2/9/2022, 15:00
5.	0,106	55,38	33,17	53	59,84	54	10,81	Rendah	Tidak Terkirim	2/9/2022, 17:00
6.	0,106	46,87	34,71	56	62,84	61	3,02	Rendah	Tidak Terkirim	2/11/2022, 17:00
7.	0,053	53,11	34,21	56	58,22	56	3,96	Rendah	Tidak Terkirim	2/12/2022, 8:30
8.	0,159	49,02	35,64	56	67,46	59	14,34	Rendah	Tidak Terkirim	2/12/2022, 9:00

9.	0,106	51,41	34,39	55	61,84	56	10,43	Rendah	Tidak Terkirim	2/12/2022, 16:00
10.	0,106	63,29	29,57	55	61,84	55	12,44	Rendah	Tidak Terkirim	2/15/2022, 3:00
...
48.	0,106	57,75	30,87	84	87	90,84	4,41	Waspada	Terkirim	2/2/2022,16:30:01
49.	0,106	41,26	38,85	167	170	173,84	2,26	Bahaya	Terkirim	1/31/2022,17:00:02
50.	0,106	48,49	33,56	187	189	193,84	2,56	Bahaya	Terkirim	2/1/2022, 18:00:01
51.	0,106	42,95	36,59	170	172	176,84	2,81	Bahaya	Terkirim	2/2/2022, 11:00:02

Dari tabel tersebut telah didapatkan hasil APE, maka data aktual tinggi air dibandingkan dengan data prediksi tinggi air untuk mengetahui hasil akurasi sistem dan dihitung nilai *error* dengan menggunakan MAPE. Berikut perhitungan MAPE untuk mengetahui *error* pada akurasi prediksi sebagai berikut.

$$\begin{aligned}
 &6,64 + 2,07 + 4,17 + 12,44 + 10,81 + 3,02 + \\
 &3,96 + 14,34 + 10,43 + 12,44 + 1,34 + \\
 &5,91 + 8,46 + 1,3 + 2,22 + 1,5 + 5,26 + \\
 &8,42 + 0,42 + 7,11 + 11,23 + 9,13 + 0,42 + \\
 &11,23 + 3,84 + 1,47 + 0,25 + 11,23 + 3,24 + \\
 &10,81 + 8,46 + 0,32 + 8,46 + 5,82 + 0,24 + \\
 &7,33 + 2,97 + 1,26 + 8,07 + 0,37 + 10,25 + \\
 &2,18 + 15,96 + 2,3 + 11,23 + 3,23 + 0,18 + 4,41 + \\
 &2,26 + 2,56 + 2,81 \\
 \text{MAPE} = &\frac{\quad}{51}
 \end{aligned}$$

$$\text{MAPE} = 5,6\%$$

$$\text{Persentase Keberhasilan Prediksi} = 100\% - 5,6\% = 94,4\%$$

3.6 Pembahasan

Proses pengiriman data hasil pemantauan baik dari *node* sensor cuaca ataupun *node* sensor tinggi air ke firebase dilakukan oleh ESP32 melalui jaringan internet. Proses pengiriman data dari *node* sensor cuaca dan *node* sensor tinggi air ke firebase dilakukan oleh ESP32 dengan mengeksekusi baris program yang telah dibuat. Hasil pembacaan setiap sensor oleh ESP32 akan dikirim ke firebase ketika ESP32 menjalankan program yang telah ditanam. Data yang didapat dari hasil pemantauan masing-masing sensor akan dikirimkan ke firebase setiap 30 menit sekali. Sebelum mengirim data ke firebase data pemantauan dari masing-masing sensor ditampung terlebih dahulu pada variabel. Tipe data variabel yang digunakan untuk masing-masing sensor disesuaikan dengan keluaran masing-masing sensor.

Proses prediksi ketinggian air menggunakan metode Regresi Linear sederhana. Untuk dapat menentukan nilai prediksi dilakukan perhitungan nilai a dan b terlebih dahulu. Nilai a dan b didapatkan dari hasil perhitungan menggunakan data latih yang telah diambil oleh sensor sebanyak 67 data. Setelah nilai a dan b didapatkan maka dapat dilakukan proses prediksi ketinggian air. Proses prediksi terjadi apabila terjadi hujan. Sistem akan menghitung nilai ketinggian air menggunakan nilai dari sensor curah hujan sebagai variabel x. Nilai yang didapat dari hasil perhitungan tinggi air kemudian ditambahkan dengan nilai tinggi air terkini. Apabila hasil perhitungan ditambah dengan nilai tinggi air terkini memenuhi kondisi pada sistem maka akan dikirim notifikasi peringatan dini banjir kepada pengguna.

Pada proses pengujian didapatkan data uji sebanyak 51 data kejadian hujan. Dari data tersebut dapat dihitung persentase *error* absolut (APE). Setelah APE didapatkan maka dilanjutkan dengan perhitungan MAPE. Nilai MAPE yang dihasilkan sebesar 5,6% dengan itu juga diperoleh akurasi sebesar 94,4%. Pada penelitian ini dibuat beberapa level ketinggian air yaitu rendah, normal, siaga, waspada, dan bahaya. Ketinggian air berada pada level rendah ketika < 70cm, level normal ketika di antara 70cm dan 80cm, level siaga ketika di antara 80cm dan 90cm, level waspada ketika di antara 90cm dan 100cm, dan level bahaya ketika >100cm. Notifikasi juga akan terkirim jika hasil prediksi menunjukkan level siaga, waspada, dan bahaya. Notifikasi peringatan dini banjir akan dikirimkan oleh sistem secara otomatis ketika ketinggian air berada pada level siaga, waspada, dan bahaya.

Untuk mendapatkan notifikasi dari sistem pengguna perlu mengizinkan penerimaan notifikasi terlebih dahulu yang akan tampil saat mengunjungi *website*. Setelah pengguna mengizinkan untuk menerima notifikasi maka pengguna dianggap telah berlangganan untuk menerima notifikasi dari sistem. Pesan dari notifikasi akan menyesuaikan dengan nilai yang didapatkan oleh sensor sesuai dengan kondisi yang telah dideklarasikan. Jika kondisi pada sistem terpenuhi maka akan terkirim notifikasi ke pengguna yang telah berlangganan.

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, sistem monitoring cuaca dan peringatan dini banjir dibangun dengan membuat *node* sensor cuaca dan *node* sensor tinggi air yang terintegrasi dengan aplikasi *website*. Sistem ini dapat memonitor cuaca dan memberikan peringatan dini bahaya banjir secara *realtime*. Peringatan dini banjir dikirimkan dalam bentuk *push notification*. Notifikasi yang akan dikirimkan oleh sistem secara otomatis ketika ketinggian air berada pada level siaga (tinggi air di antara 80cm dan 90cm), waspada (tinggi air di antara 90cm dan 100cm), dan bahaya (tinggi air ketika >100cm). Untuk mendapatkan notifikasi dari sistem pengguna perlu mengizinkan penerimaan notifikasi terlebih

dahulu yang akan tampil saat mengunjungi *website*. Setelah pengguna mengizinkan untuk menerima notifikasi maka pengguna dianggap telah berlangganan untuk menerima notifikasi dari sistem. Metode regresi linear diterapkan untuk memprediksi ketinggian air dengan menghitung kenaikan ketinggian air menggunakan curah hujan sebagai variabel independen. Pengujian sistem menggunakan metode MAPE dengan 51 data kejadian hujan mendapatkan nilai *error* sebesar 5,6% dan nilai akurasi sebesar 94,4%. Saran untuk pengembangan selanjutnya, yaitu dapat menerapkan metode prediksi lainnya agar membedakan hasil yang telah diperoleh dan menambahkan parameter yang digunakan pada penelitian agar mendapatkan akurasi yang lebih baik.

REFERENCES

- [1] Badan Nasional Penanggulangan Bencana, *Buku Saku Tanggap Tangkas Tangguh Menghadapi Bencana*, Jakarta: Pusat data Informasi dan Humas BNPB, 2019.
- [2] Andilala, “Air pasang laut di Pontianak capai ketinggian di atas normal,” 5 Desember 2019. [Online]. Available: <https://www.antaranews.com/berita/1194848/air-pasang-laut-di-pontianak-capai-ketinggian-di-atas-normal>.
- [3] Hi!Pontianak, “Hujan Deras di Pontianak, Sejumlah Ruas Jalan Terendam Banjir,” 23 Januari 2020. [Online]. Available: <https://kumparan.com/hipontianak/hujan-deras-di-pontianak-sejumlah-ruas-jalan-terendam-banjir-1shPq8pYbNB/full>.
- [4] S. C. V. E. Spyros Makridakis, *Metode dan Aplikasi Peramalan*, Jakarta: Erlangga, 2000.
- [5] T. Rahajoeningoem, “Sistem Monitoring Cuaca dan Deteksi Banjir pada Android Berbasis Internet of Things (IoT),” *Prosiding SAINTIKS FTIK UNIKOM*, vol. 2, pp. 33-40, 2017.
- [6] F. Erwan, “Rancang Bangun Sistem Pengukur Cuaca Otomatis Menggunakan Arduino dan Terintegrasi dengan Website,” *Jurnal Coding, Sistem Komputer Untan*, pp. 255-264, 2018.
- [7] H. Kurniawan, “Rancang Bangun Sistem Pendeteksi dan Monitoring Banjir Menggunakan Arduino dan Website,” *Coding : Jurnal Komputer dan Aplikasi*, pp. 11-22, 2019.
- [8] M. Zain, “Implementasi Forecasting Pada Perancangan Sistem Pembukaan Kelas di STIKOM Bali dengan Menggunakan Metode Regresi Linear,” *EKSPLORA INFORMATIKA*, pp. 17-28, 2013.
- [9] L. Shenzhen Ai-Thinker Technology Co., “Nodemcu-32s Datasheet Version V1,” 2019.
- [10] Bosch Sensortec GmbH, *BME280 Integrated Environmental Unit*, Reutlingen: Bosch Sensortec GmbH, 2015.
- [11] D. Pangestu, “PURWARUPA SISTEM INFORMASI TITIK LOKASI DAN INTENSITAS CURAH HUJAN DI KOTA PONTIANAK BERBASIS WEBSITE,” *Coding Jurnal Komputer dan Aplikasi*, pp. 247-254, 2018.
- [12] A. Sandriyadi, *ALAT BANTU TUNANETRA BERBASIS ARDUINO*, Yogyakarta: STMIK Akakom Yogyakarta, 2017.
- [13] Google, “Firebase Realtime Database,” 18 Desember 2020. [Online]. Available: <https://firebase.google.com/docs/database?hl=id>.
- [14] Arduino, “Arduino Integrated Development Environment,” 23 Februari 2022. [Online]. Available: <https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics>. [Diakses 20 04 2022].
- [15] S. M. Umyy Gusti Salamah, *Tutorial Visual Studio Code*, Media Sains Indonesia, 2021.
- [16] A. Nordeen, *Learn NodeJS in 24 Hours: Complete Node JS Guide with Examples*, Guru99, 2020.
- [17] Y. Yudhanto dan A. Azis, *Pengantar Teknologi Internet of Things (IoT)*, Surakarta: UNSPress, 2019.
- [18] S. Yamin, *Tutorial Statistik SPSS, Lisrel, WARPPLS, & JASP (Mudah & Aplikatif)*, Bekasi: Dewangga Energi Internasional Publishing, 2021.
- [19] Q. A'yunin, A. D. Sulistyono, A. Syawli, A. Rahmawati, C. A. Intyas, D. Aliviyanti, E. N. Wiratno, F. O. Setyawan, F. E. Supriatin, H. Djamaludin, J. E. Tambunan, M. K. Rihm, N. H. Wardani, S. S. Rijal, S. Anitasari, T. N. Ma'rifat dan W. K. Sari, *Perikanan Berkelanjutan*, Malang: Tim UB Press, 2021.