

PROTOTYPE PERINGATAN DINI KETINGGIAN AIR BENDUNGAN MENGUNAKAN ESP32-CAM

Eko Nurfitriyono

**Pembimbing Utama: Andi Yusika Rangan, S.Kom., M.Kom Pembimbing Pendamping: H. Pajar Pahrudin,
S.Kom., M.H**

Program Studi Teknik Informatika, STMIK Widya Cipta Dharma
Jl. Prof. M. Yamin No. 25 Samarinda Kalimantan Timur 75123
Telp: (0541) 736071, Fax: (0541) 203492
E-mail: ekonurfitriyono@gmail.com

ABSTRAK

Bendungan sebagai wadah penampungan air, yang banyak digunakan oleh masyarakat untuk kebutuhan sehari-hari namun jika air pada bendungan dibiarkan tanpa pengawasan, maka akan menjadi suatu masalah yang dapat menimbulkan bencana banjir. Debit air yang tidak terkontrol akibat curah hujan yang ekstrem dapat membuat suatu bendungan meluap dikarenakan kurangnya informasi ke pengawas, sehingga diperlukan alat yang dapat memberikan peringatan secara otomatis kepada pengawas, sehingga pengawas dapat melakukan penanganan dengan cepat untuk membuka pintu air. Metode dalam penelitian ini menggunakan metode waterfall. Dalam penelitian ini, teknik pengumpulan data yang digunakan adalah studi lapangan, observasi dan studi pustaka. Pengujian alat dilakukan dengan menggunakan pengujian blackbox testing dan whitebox testing. Serta perangkat lunak pendukung yang digunakan Arduino Integrated Development Environment, dan Blynk. Hasil dari penelitian ini adalah dibuatnya alat pendeteksi dan peringatan dini bendungan dengan menggunakan ESP32-Cam dan sensor Ultrasonik, dimana aplikasi Blynk digunakan sebagai user interface yang dapat menampilkan notifikasi, menampilkan data ketinggian dan live streaming video ketinggian air bendungan. Penelitian ini dilakukan di Bendungan Benanga, Kota Samarinda.

Kata Kunci: *Prototype*, Peringatan, Ketinggian, Air, Bendungan, ESP32-Cam, Sensor, Ultrasonik, Blynk, Notifikasi

1. PENDAHULUAN

Seiring dengan perkembangan zaman, air adalah salah satu elemen penting di bumi, tanpa air tidak akan ada ekosistem lingkungan. Air juga mempunyai peranan penting dalam kehidupan manusia. Tanpa air manusia tidak dapat hidup. Air digunakan dalam kehidupan sehari-hari, untuk keperluan rumah tangga sampai dengan industri.

Demikian pula halnya dengan bendungan sebagai wadah penampungan air, yang banyak digunakan oleh masyarakat untuk kebutuhan sehari-hari namun jika air pada bendungan dibiarkan tanpa pengawasan, maka akan menjadi suatu masalah yang dapat menimbulkan bencana banjir.

Pengawasan terhadap ketinggian air pada bendungan merupakan pekerjaan yang tidak terlalu berat, namun jika terjadi kelalaian dalam pengawasan akibatnya sangat merugikan karena menyangkut keselamatan warga disekitarnya.

Debit air yang tidak terkontrol akibat curah hujan yang ekstrem dapat membuat suatu bendungan meluap dikarenakan kurangnya informasi ke pengawas, sehingga diperlukan alat yang dapat memberikan peringatan secara otomatis kepada pengawas, sehingga pengawas dapat melakukan penanganan dengan cepat untuk membuka pintu air.

Sesuai latar belakang yang telah diuraikan di atas, maka penulis membuat suatu Prototype Peringatan Dini Ketinggian Air Bendungan Menggunakan ESP32-CAM dan Sensor Ultrasonik yang berfungsi menggantikan peran manusia dalam mengawasi aktifitas ketinggian air.

2. RUANG LINGKUP

Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan suatu masalah yaitu “Bagaimana cara membangun Prototype Peringatan Dini Ketinggian Air Bendungan Menggunakan ESP32-CAM Dan Sensor Ultrasonik?”

Batasan Masalah

Pembuatan Prototype Peringatan Dini Ketinggian Air Bendungan Menggunakan ESP32-CAM Dan Sensor Ultrasonik memiliki batasan-batasan masalah yakni sebagai berikut:

1. Peringatan dini ketinggian air bendungan menggunakan aplikasi Blynk.
2. Menggunakan sensor Ultrasonik.
3. Menggunakan mikrokontroler Esp32 Cam
4. Informasi ketinggian air mempunyai 4 level, yaitu: Aman (0 – 5 CM), Siaga (6 – 10 CM), Awas(11 -15 CM), Waspada(16 – 20 CM).
5. Peringatan dikirimkan ketika ketinggian air sudah mencapai level tertentu, yaitu: Siaga 30 detik sekali, Awas 20 detik sekali, Waspada 10 detik sekali.
6. Informasi peringatan Blynk berisi tentang ukuran dan level ketinggian air.
7. Sensor dapat di ON/OFF.
8. Monitor informasi ketinggian air ditampilkan pada aplikasi Blynk.
9. Sensor diletakkan di mulut saluran pelimpah bendungan.
10. Menggunakan kamera untuk dapat melihat ketinggian air secara langsung melalui aplikasi Blynk.

3. BAHAN DAN METODE

Dalam metode penelitian ini terdapat beberapa hal yang harus diperhatikan, antara lain sebagai berikut:

3.1 Sistem Minimum

Menurut Ekojono (2018), Sistem Minimum adalah penerapan mikroprosesor pada suatu rangkaian digital, dengan komponen minimal sehingga sistem mikroprosesor dapat bekerja. Dalam sistem minimum diperlukan perangkat-perangkat seperti, unit memori, unit masukan dan keluaran, dan rangkaian pendukung lain.

Menurut Immersa Lab (2014), Sistem Minimum Mikrokontroler adalah sebuah rangkaian paling sederhana dari sebuah mikrokontroler agar IC mikrokontroler tersebut bisa beroperasi dan deprogram. Dalam aplikasinya sistem minimum sering dihubungkan dengan rangkaian elektronik dan rangkaian lain untuk tujuan tertentu.

Berdasarkan definisi di atas dapat disimpulkan bahwa sistem minimum adalah rangkaian digital sederhana yang minimal berisi unit memori, unit masukan dan keluaran.

3.2 Mikrokontroler

Menurut Ekojono (2018), Mikrokontroler adalah sebuah sistem mikroprosesor lengkap yang terkandung di dalam sebuah chip. Mikrokontroler berbeda dari mikroprosesor serba guna yang digunakan dalam sebuah PC, karena sebuah mikrokontroler umumnya telah berisi komponen pendukung sistem minimal mikroprosesor, yakni memori dan antarmuka I/O.

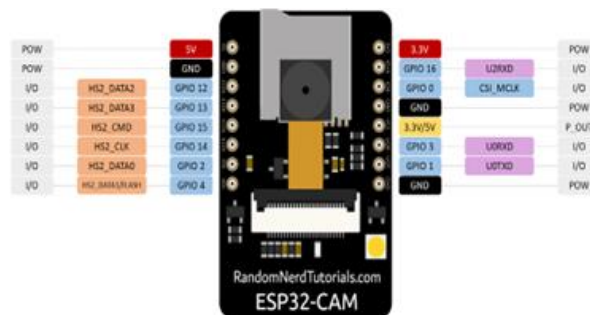
Sumardi (2015), Menyatakan bahwa mikrokontroler adalah mikroprosesor yang dikhususkan untuk instrumentasi dan kendali. Mikroprosesor merupakan suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus. Mikrokontroler merupakan komputer didalam chip yang digunakan untuk mengontrol peralatan elektronik, yang

menekankan efisiensi dan efektivitas biaya. Secara harfiah disebut “pengendali kecil” dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan pendukung seperti IC TTL dan CMOS dapat direduksi/diperkecil dan akhirnya terpusat serta dikendalikan oleh mikrokontroler ini.

Berdasarkan definisi di atas dapat disimpulkan bahwa Mikrokontroler adalah suatu instrumen elektronik yang digunakan untuk sistem kendali, dimana dalam sistemnya terdapat *input* dan *output*.

3.3 ESP32-CAM

Menurut Serrhini (2020) ESP32 adalah mikrokontroler SoC yang diproduksi oleh perusahaan China Espressif System. Ini adalah mikrokontroler murah dengan kemampuan terintegrasi Bluetooth dan Wi-Fi. Seri ESP32 menggunakan mikroprosesor Tensilica Xtensa LX6. Dalam tulisan ini, ESP32 dengan modul kamera digunakan

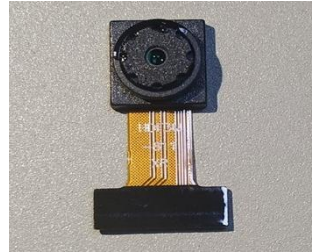


Gambar 1. ESP32-CAM

3.4 Kamera OV2640

Menurut Liang (2016), OV2640 adalah sensor gambar CMOS, sensor video pertama dengan ukuran 0,25 inci dan 2-juta piksel, merupakan sensor gambar baru dengan arsitektur 2,2 mikron OmniPixel2. Itu dapat dirakit menjadi produk soket skala mm, yang saat ini menjadi salah satu modul kamera miniatur. OV2640 menggunakan arsitektur OmniPixel2 berteknologi tinggi dan canggih, dengan warna yang lebih kaya dan struktur lensa mikro celah nol, untuk meningkatkan faktor pengisian dan kuantum, dan membentuk piksel 2,2 mikron yang sensitif, meningkatkan efisiensi dan efek tampilan. Selain manfaat yang dibawa oleh arsitektur OmniPixel2, OmniVision mengintegrasikan modul pemrosesan sinyal gambar tingkat lanjut (OmniQSP), yang sangat meningkatkan kinerja gambar. OV2640 memiliki lima unit, termasuk unit pemrosesan sinyal analog, konverter A/D 10-bit, unit pemrosesan sinyal digital, unit spesifikasi format output, dan unit kompresi, ia juga memiliki mikrokontroler, antarmuka periferifal SCCB, dan antarmuka output video digital.

Berdasarkan definisi di atas dapat disimpulkan bahwa kamera OV2640 adalah miniatur kamera yang mana menggunakan arsitektur Omnipixel2, berukuran 0.25 inci dan 2-juta piksel.



Gambar 2. Kamera OV2640

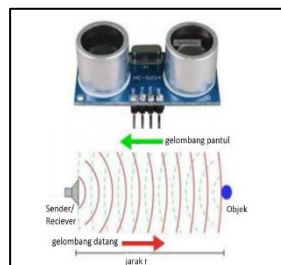
3.5 Sensor HC-SR04

Menurut Fitri (2019), Sensor ultrasonik tipe HCSR04 merupakan perangkat yang digunakan untuk mengukur jarak dari suatu objek. Kisaran jarak yang dapat diukur sekitar 2-450 cm. Perangkat ini menggunakan dua pin digital untuk mengkomunikasikan jarak yang terbaca. Prinsip kerja sensor ultrasonik ini bekerja dengan mengirimkan pulsa ultrasonik sekitar 40 KHz, kemudian dapat memantulkan pulsa echo kembali, dan menghitung waktu yang diambil dalam mikrodetik.

$$\text{Jarak} = \text{Kecepatan_suara} \times \text{waktu_pantul} / 2$$

Sensor Ultrasonik Devantech SR04 dengan spesifikasi sebagai berikut :

1. Bekerja pada tegangan DC 5 volt
2. Beban arus sebesar 30 mA-50 mA
3. Menghasilkan gelombang dengan frekuensi 40 KHz
4. Jangkauan jarak yang dapat dideteksi 3 cm-450 cm
5. Membutuhkan trigger input minimal sebesar 10 uS
6. Dapat digunakan dalam dua pilihan mode yaitu input trigger dan output echo terpasang pada pin yang berbeda atau input trigger dan output echo terpasang dalam satu pin yang sama.



Gambar 3. Sensor HC-SR04

3.6 Blynk

Menurut Pradeeka (2018), Blynk disebut sebagai platform IoT yang paling ramah pengguna, menyediakan cara untuk membangun aplikasi seluler dalam hitungan menit. Dengan pembuat aplikasi seluler seret dan lepas Blynk, siapa pun dapat membangun aplikasi IoT yang luar biasa dengan sumber daya dan upaya minimal. Blynk mendukung lebih dari 400 platform perangkat keras dan jenis konektivitas utama. Perangkat kerasnya bisa berupa platform prototyping, seperti Arduino dan Raspberry Pi, hingga ESP8266 kelas industri, Intel, Sierra Wireless, Particle, dan penawaran Texas Instruments.

3.7 Arduino IDE

Menurut Jody dkk (2017), Integrated Development Environment (IDE) adalah aplikasi perangkat lunak yang memungkinkan anda menulis kode dan menguji kode tersebut dalam bahasa pemrograman yang didukung IDE.



Gambar 4. Flowchart Alur Kerja Arduino IDE

3.8 Metode Pengembangan Sistem

Menurut Sri (2016), Sistem bisa diartikan sebagai sekumpulan subsistem, komponen ataupun elemen yang saling bekerjasama dengan tujuan yang sama untuk menghasilkan output yang sudah ditentukan sebelumnya.

Sistem yang sedang berjalan atau sedang digunakan organisasi atau perusahaan akan terus dikembangkan untuk memperbaiki kekurangan-kekurangan pada sistem tersebut. Untuk melakukan pengembangan sistem, metode yang digunakan adalah SDLC. Metode adalah tahap-tahap ataupun aturan untuk melakukan sesuatu.

SDLC adalah sebuah metode yang digunakan untuk mengembangkan sebuah sistem. SDLC adalah sebuah proses logika yang digunakan oleh seorang system analyst untuk mengembangkan sebuah sistem informasi yang melibatkan requirements, validation, training dan pemilik sistem.

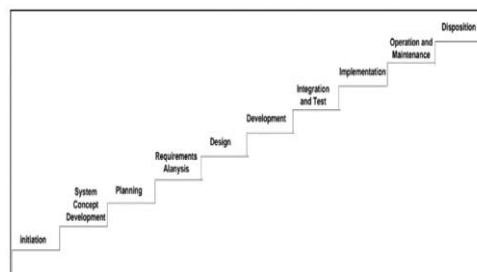
SDLC identik dengan teknik pengembangan sistem waterfall, karena tahapannya menurun dari atas kebawah.

3.9 Model Waterfall

Menurut Putra (2020), Metode Waterfall merupakan metode pengembangan perangkat lunak yang paling sederhana dan terkadang disebut sebagai classic life cycle (siklus hidup klasik) yang menyarankan pendekatan sistematis dan sekuensial untuk pengembangan perangkat lunak.

Menurut Sri (2016), Seperti contoh diatas yang memberikan gambaran kepada kita sebanyak 10 tahapan dalam melakukan pengembangan sistem, namun pada kondisi-kondisi tertentu ada tahapan-tahapan yang tidak perlu kita lakukan atau ada tahapan yang harus kita tambahkan.

1. Initiation/Planning, merupakan tahap dimana sistem digambarkan secara global beserta tujuan yang akan direncanakan terhadap sistem yang akan dikembangkan. Tahap ini identik dengan tahap analisis..
2. Requirement Gathering and Analysis, pada tahap ini analis mencoba untuk menguraikan permasalahan sistem dan menggambarkannya kedalam beberapa diagram untuk menggambarkan situasi yang sedang berjalan, kemudian pada tahap ini juga analis mencoba mendesain sebuah solusi yang akan diberikan kepada user.
3. Design, pada tahap ini solusi-solusi yang sudah digambarkan secara global pada tahap requirement gathering and analysis diuraikan secara detail baik dalam bentuk diagram, layouts, business rules, dan dokumentasi-dokumentasi lain yang dibutuhkan.
4. Build or Coding, pada tahap ini sistem mulai di bangun atau dikembangkan. Tahap ini identik dengan pembuatan program aplikasi untuk mendukung sistem.
5. Testing, pada tahap ini sistem yang sudah dibangun atau dikembangkan dicoba oleh tim tester ataupun oleh user.



Gambar 5. Tahapan Pengembangan Sistem SDLC Waterfall

3.10 Flowchart

Menurut Wahyudi (2021), Flowchart (Diagram Alir) adalah sebuah jenis diagram yang mewakili algoritma, alir kerja atau proses, yang menampilkan langkah-langkah dalam bentuk simbol-simbol grafis, dan urut-urutannya dihubungkan dengan panah (Myers, 1986). Flowchart digunakan untuk dokumentasi yang menggambarkan proses dan urutan yang dirancang. Karena dirancang dalam bentuk visual, maka flowchart

memiliki keunggulan dalam merepresentasikan algoritma. Karena itu, standar internasional untuk penyusunan algoritma tidak dengan kalimat, akan tetapi dibuat secara visual dengan bantuan simbol-simbol baku yang memiliki arti, yaitu dalam bentuk flowchart.

Menurut Sitorus (2015), untuk menggambarkan sebuah algoritma yang terstruktur dan mudah dipahami oleh orang lain (khususnya programmer yang bertugas mengimplementasikan program), maka dibutuhkan alat bantu yang berbentuk diagram alir (flowchart). Flowchart menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga flowchart merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. Diagram alir ini selain dibutuhkan sebagai alat komunikasi, juga diperlukan sebagai dokumentasi.

3.11 Blok Diagram

Menurut Sugianto (2015), Diagram dari sebuah sistem, dimana bagian utama atau fungsi yang diwakili oleh blok dihubungkan dengan garis, yang menunjukkan hubungan dari blok. Banyak dalam dunia rekayasa dalam desain hardware, desain elektronik, software desain dan proses aliran diagram.



Gambar 6. Blok Diagram

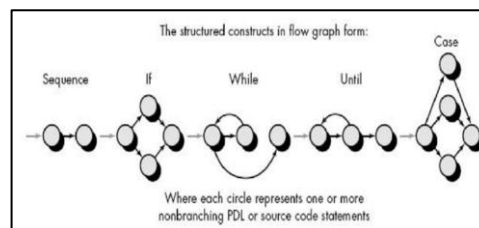
3.12 Pengujian Black Box

Menurut Sari (2021), Pengujian black box digunakan untuk memperlihatkan bahwa fungsi-fungsi perangkat lunak dapat beroperasi, bahwa input diterima dengan baik dan output dihasilkan dengan tepat, dan integritas informasi eksternal (seperti file data) dipelihara. Pengujian black box menguji beberapa aspek dasar suatu sistem dengan memperhatikan sedikit struktur logika internal perangkat lunak tersebut.

3.13 Pengujian White Box

Menurut Sari (2021), White-Box Testing (kadang-kadang disebut sebagai glass-box testing) adalah metode desain kasus uji yang menggunakan struktur kontrol desain prosedural untuk memperoleh kasus uji.

Menurut Kholiq (2022), Pengujian kotak putih adalah pendekatan desain kasus yang memperoleh kasus uji melalui struktur proses kontrol demi desain.



Gambar 7. Flow Graph

4. PEMBAHASAN

4.1 Gambaran Umum

Alat ini dibangun untuk memudahkan pemilik rumah burung walet dalam memantau suhu dan kelembaban ruangan pada rumah walet sehingga bisa meningkatkan kualitas sarang walet dan mengurangi persentase gagal panen.

Dengan memanfaatkan kemajuan teknologi, proses pemantauan suhu dan kelembaban ruangan pada rumah burung walet ini, dapat dilakukan dengan alat *monitoring* suhu dan kelembaban berbasis Android dan disertai dengan sistem kontrol pendingin ruang pada rumah burung walet. Pada alat *monitoring* suhu dan kelembaban ruangan pada rumah burung walet berbasis Android ini, data hasil pengukuran suhu dan kelembaban ruangan pada rumah burung walet dapat ditampilkan di sebuah smartphone melalui jarak jauh, serta adanya sistem kontrol pendingin ruang pada rumah burung walet yang memanfaatkan smartphone sebagai remote kontrol.

4.2 Pembahasan

1. Analisis

Analisis merupakan teknik pemecahan masalah yang menguraikan bagian komponen dengan mempelajari seberapa bagus bagian komponen tersebut bekerja.

1. Analisis Data

Analisis data yang digunakan dalam membangun sistem ini antarlain :

- 1) Datasheet ESP32-Cam

- 2) Datasheet Wemos D1 Mini
- 3) Datasheet Sensor HC-SR 04
- 4) Datasheet Blynk

2. Analisis Sistem

Pada analisis sistem terdapat beberapa kondisi khusus untuk mengukur suhu dan kelembaban ruangan pada rumah walet berbasis Android ini antara lain alat ini menghasilkan keluaran yaitu sebagai alat pendeteksi ketinggian air bendungan yang mana nantinya dapat dimonitoring jarak jauh melalui handphone menggunakan aplikasi Blynk. Pengguna aplikasi dapat melihat indikator ketinggian air bendungan secara realtime dan juga bisa melihat keadaan air bendungan berupa video.

3. Analisis Teknologi

Pada pengoperasian alat ini diperlukan perangkat keras dan perangkat lunak, yang dibutuhkan antara lain :

1. Perangkat Keras

- 1) ESP32-Cam
- 2) Sensor HC-SR04
- 3) Relay
- 4) Kabel Jumper
- 5) Breadboard
- 6) Wemos D1 Mini

2. Perangkat Lunak

- a) *Software Arduino Integrated Developmont Environment*
- b) Blynk

4. Analisis User

Pada analisis user tahapan yang dilakukan adalah menentukan siapa yang akan menggunakan sistem ini. Target pengguna user adalah :

1. Penjaga bendungan.
2. Mahasiswa yang akan melakukan penelitian yang berkaitan dengan alat pendeteksi ketinggian air bendungan.

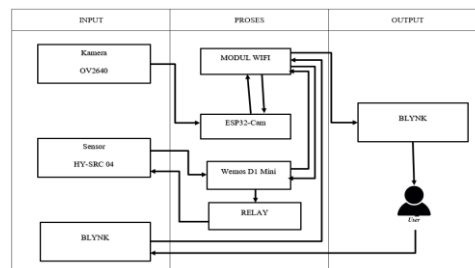
2. Desain Sistem

Desain merupakan sebuah proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta di dalam melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami proses pengerjaan.

1. Desain Perangkat Keras

Perancangan alat merupakan langkah yang amat penting dalam merealisasikan sistem yang sesungguhnya. Tujuan perancangan ini adalah untuk merencanakan perangkat keras sesuai dengan spesifikasi, cara kerja dari sistem yang dibuat, sehingga diharapkan dapat mengefisienkan waktu, biaya dan tenaga.

1. Blok Diagram

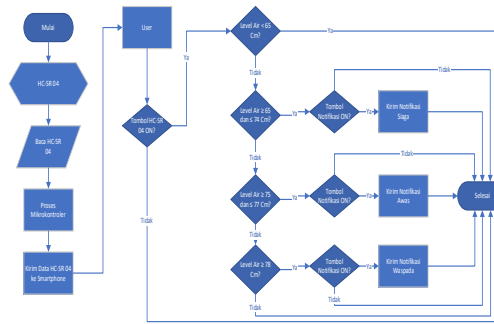


Gambar 8. Blok Diagram

Berikut merupakan penjelasan rancangan blok diagram prototype peringatan dini ketinggian air bendungan menggunakan ESP32-cam dan sensor ultrasonik, Kamera OV2640 menangkap gambar berupa video dan Sensor HC-SR 04 mendeteksi ketinggian air bendungan lalu mikrokontroler memproses data yang diberikan untuk kemudian dikirim ke handphone pengguna melalui wifi sebagai antar muka Blynk, dimana akan ada UI yang menampilkan indikator ketinggian air bendungan dan juga video streaming melalui kamera yang terdapat pada ESP32-Cam. Pada input Blynk terdapat menu pilihan untuk mengaktifkan atau menonaktifkan sensor, notifikasi dan flash:

- 1) ESP32 Cam : Sebagai Pengendali dari semua komponen dan kamera.
- 2) HC-SR04 : Sensor Jarak.
- 3) Relay: Saklar untuk ON/OFF sensor HC-SR04.
- 4) Android : sebagai remot kontrol, menampilkan hasil pengukuran HC-SR04 dan Kamera OV2640.

2. Flowchart



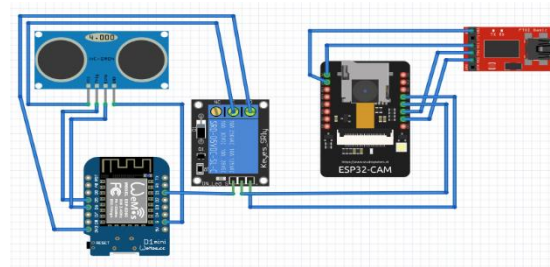
Gambar 9. Flowchart Sistem

2. Implementasi

Setelah melakukan Analisa dan desain, maka dilakukan implementasi. Dalam hal ini difokuskan pada pembuatan dan penggunaannya. Untuk level air ≤ 5 cm maka indikator aman, untuk level air ≥ 6 dan ≤ 10 cm maka indikator siaga, untuk level air ≥ 11 dan ≤ 15 cm maka indikator awas, untuk level air ≥ 16 cm maka indikator waspada, masing-masing terdapat notifikasi kecuali saat indikator berada di level aman atau pengguna dengan sengaja menonaktifkan notifikasi.

1. Perangkat Keras (*Hardware*)

Perangkat keras atau hardware yang digunakan pada alat peringatan dini ketinggian air bendungan adalah sensor HC-SR04 yang merupakan sensor Ultrasonik, ESP32-Cam yang merupakan mikrokontroler yang akan memproses data Kamera sehingga menjadi data output atau video, Relay sebagai switch ON/OFF sensor, Wemos D1 Mini untuk mengontrol Relay dan menerima Output Ultrasonik, FTDI sebagai komponen untuk meng-upload kode program ke mikrokontroler.



Gambar 10. Rangkaian Alat Monitoring Suhu dan Kelembaban

2) Penggunaan Alat

Ketika alat yang telah dirancang di hidupkan, Wifi module pada mikrokontroler akan menghubungkan ke jaringan wifi kemudian akan terhubung ke aplikasi Blynk, sensor Ultrasonik akan mulai bekerja dengan menembakkan gelombang Ultrasonik lalu mikrokontroler akan mengukur waktu tembak gelombang ditambah jeda waktu pantulan suara yang diterima kembali oleh sensor Ultrasonik, sehingga dihasilkan jarak ukur sensor dengan permukaan. Data jarak ukur akan diolah dan kemudian akan dikirimkan ke Blynk sebagai tampilan antar muka pengguna yang telah diatur. Demikian pula dengan kamera yang dapat yang berfungsi mengirimkan video live streaming yang dapat diakses dengan alamat IP. Alat ini juga terdapat Relay yang digunakan untuk menonaktifkan sensor jika tidak digunakan, yang mana dapat menghemat daya jika alat ingin dikembangkan menggunakan daya baterai.



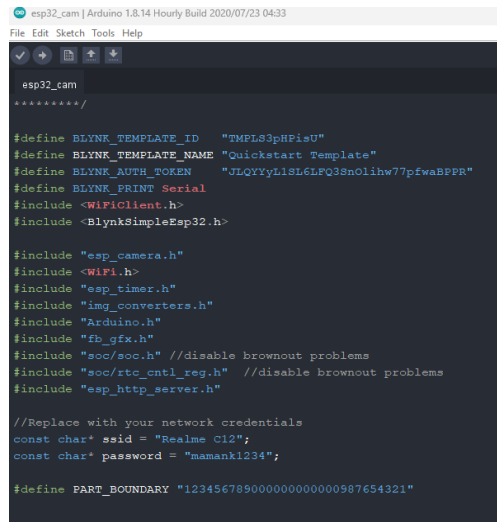
Gambar 11. Alat Pengukur Ketinggian Air bendungan

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan pada pembuatan program menggunakan aplikasi Arduino Integrated Development Environment untuk penulisan program pada mikrokontroler. Untuk menampilkan hasil pengukuran tinggi air menggunakan Smartphone serta sebagai control menggunakan aplikasi Blynk.

1) Pemrograman pada Mikrokontroler ESP32-CAM

Pada pembahasan ini akan dibahas tentang pembuatan program menggunakan aplikasi Arduino Integrated Development Environment. Dalam aplikasi Arduino Integrated Development Environment terdapat sketch untuk mengetikkan program dan tool-tool seperti verify atau compile, upload, new, open dan save project.

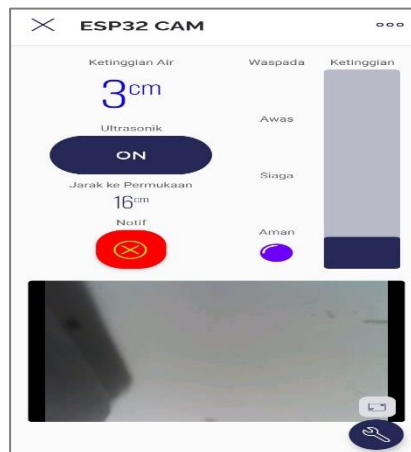


Gambar 12. Program Pada Arduino IDE

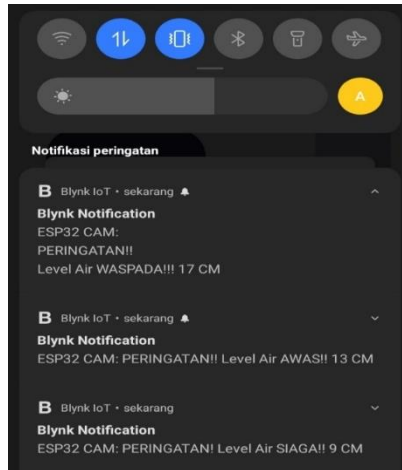
2) Tampilan Pada Aplikasi Blynk

Pada aplikasi blynk ini terdapat tampilan widget pada gambar 19. Tampilan pada Blynk akan dijelaskan fungsinya secara detail sebagai berikut:

- 4 LED dengan warna berbeda untuk menentukan tingkat level air (Aman, Siaga, Awas, Waspada).
- Kemudian terdapat Widget Level V yang menampilkan ketinggian air dalam bentuk Bar.
- Selain itu terdapat 2 buah widget Labeled Value, yaitu Ketinggian Air dan Jarak ke Permukaan. Fungsi dari Labeled Value “Ketinggian Air” adalah menunjukkan ketinggian air dihitung dari dasar air ke permukaan air, sedangkan Labeled Value “Jarak ke Permukaan” berfungsi menunjukkan jumlah jarak permukaan air ke sensor
- Terdapat widget berupa Button “Ultrasonik” yang mana akan berfungsi untuk tombol ON/OFF Relay untuk daya sensor HC-SR 04.
- Kemudian terdapat widget Button “Notif” yang akan berfungsi sebagai Tombol ON/OFF notifikasi.
- Terdapat widget Live Streaming untuk menampilkan video yang berasal dari kamera.



Gambar 13. Tampilan Pada Aplikasi Blynk



Gambar 14. Tampilan Notifikasi

4. Pengujian (*Testing*)

Pengujian dilakukan untuk mengetahui apakah sistem telah bekerja dengan baik sesuai dengan yang diinginkan. Pengujian dilakukan dengan pengujian pada perangkat keras (*hardware*) yang digunakan dengan metode pengujian *Black Box*.

1. Pengujian *Hardware*

1) Pengujian Sensor HC-SR04

Pengujian sensor Ultrasonik akan dilakukan 4 pengujian berbeda, Yaitu “Pengujian akurasi pengukuran jarak permukaan terhadap sensor”, “Pengujian akurasi pengukuran ketinggian air”, “Pengujian Notifikasi dan LED Blynk terhadap ketinggian air” dan “Pengujian akurasi waktu jeda notifikasi”.

No	HC-SR 04 (Jarak Sensor)	Blynk	Error (CM)
1.	18	18	0
2.	14	14	0
3	10	10	0
4.	6	6	0
5.	3	3	0

Tabel 1. Hasil pengujian jarak permukaan terhadap Sensor

Pengujian Tabel 2 untuk menguji akurasi pengukuran jarak permukaan air terhadap sensor yang diperbandingkan dengan ukuran jarak yang ditampilkan pada Blynk, kemudian hasil yang di dapat dalam 5 kali percobaan tidak didapatkan selisih error.

No	HC-SR 04 (Level Air)	Blynk	Error (CM)
1.	4	4	0
2.	9	9	0
3	14	14	0
4.	17	17	0
5.	19	19	0

Tabel 2. Hasil pengujian ketinggian air

Hasil pengujian pada Tabel 3 adalah untuk melihat perbandingan error yang terdapat pada output yang dihasilkan sensor pada serial monitor Arduino IDE dengan yang terdapat pada Blynk, pengujian dilakukan dalam 5 jarak yang berbeda dan tidak ada selisih error.

No	Ketinggian Air	LED (Blynk)				Tombol Notifikasi		Hasil
		Aman	Siaga	Awas	Waspada	On	Off	
1	0 – 5 Cm	1	0	0	0	0	0	Berhasil
2	6 – 10 Cm	0	1	0	0	1	0	Berhasil
3	11 – 15 Cm	0	0	1	0	1	0	Berhasil
4	16 – 20 Cm	0	0	0	1	1	0	Berhasil

Tabel 3. Hasil Pengujian Aktivasi Notifikasi dan LED Blynk

Hasil pengujian pada Tabel 4 adalah untuk menguji keberhasilan fungsi LED dan Notifikasi pada Blynk, dimana 0 dan 1 adalah nilai yang dikirimkan mikrokontroler ke Blynk berupa boolean. Ketika nilai boolean yang dikirim bernilai 1 maka Blynk akan mengeksekusi fungsi tersebut menjadi output LED (Aman, Siaga, Awas, Waspada) atau notifikasi akan dikirimkan ke antar muka pengguna, sebaliknya jika nilai yang dikirimkan oleh mikrokontroler bernilai 0, maka Blynk tidak akan mengeksekusi tugas apapun. Berikut detail penjelasan dari tabel 4:

- Jika ketinggian air 0 – 5 CM, maka aktifkan LED Aman dan matikan LED lainnya, kemudian jangan kirim Notif meskipun kondisi tombol Notifikasi dalam keadaan ON.
- Jika ketinggian air 6 – 10 CM, maka aktifkan LED Siaga dan matikan LED lainnya, kemudian kirim notifikasi Siaga jika Tombol Notifikasi = ON; Jika tombol Notifikasi = OFF, jangan kirim notifikasi.
- Jika ketinggian air 11 – 15 CM, maka aktifkan LED Awas dan matikan LED lainnya, kemudian kirim notifikasi Awas jika Tombol Notifikasi = ON; Jika tombol Notifikasi = OFF, jangan kirim notifikasi.
- Jika ketinggian air 16 – 20 CM, maka aktifkan LED Waspada dan matikan LED lainnya, kemudian kirim notifikasi Waspada jika Tombol Notifikasi = ON; Jika tombol Notifikasi = OFF, jangan kirim notifikasi.

Kesimpulan hasil yang didapat setelah melakukan pengujian diatas adalah semua fungsi telah berjalan dengan baik sesuai desain yang diinginkan.

No	LED	Jeda Notifikasi (Detik)	Stopwatch (Detik)	Error (Detik)
1	Aman	-	-	-
2	Siaga	30	30	0
3	Awas	20	20	0
4	Waspada	10	10	0

Tabel 4. Hasil Pengujian Jeda Pengiriman Notifikasi

Kemudian hasil pengujian pada tabel 5 adalah untuk menguji akurasi jeda waktu pengiriman notifikasi ke antar muka pengguna, apakah sesuai dengan jeda waktu telah ditetapkan pada sistem mikrokontroler dengan waktu Real Time. Alat bantu yang digunakan untuk memastikan keakuratan jeda waktu sistem ini menggunakan Stopwatch. Penjelasan nya adalah setiap notifikasi akan dikirim sesuai jeda waktu yang ditentukan (Aman = Tidak Ada Notifikasi, Siaga = 15 detik sekali, Awas = 10 detik sekali, Waspada = 5 detik sekali). Kesimpulan hasil pengujian ini adalah jeda notifikasi berjalan dengan baik dan tidak terdapat selisih error, sesuai dengan ketentuan yang telah ditetapkan.

2) Pengujian Relay

Pengujian Relay untuk mengetahui fungsi tombol Relay yang ada pada Blynk dapat bekerja dengan baik untuk mengontrol switch Sensor Ultrasonik.

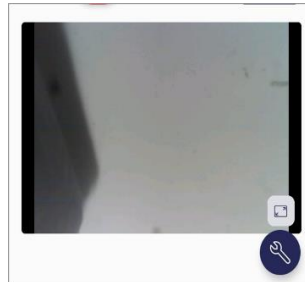
No	Tombol Relay (Blynk)	Relay	HC-SR04	Hasil
1.	1	ON	ON	Berhasil
2.	0	OFF	OFF	Berhasil

Tabel 5 Hasil pengujian Switch Relay dengan Blynk

Dari data yang di dapatkan dari tabel di atas, hasil yang didapatkan dari kendali Relay menggunakan Blynk dapat bekerja sesuai dengan algoritma yang ditentukan.

3) Pengujian Kamera

Fungsi dari kamera adalah untuk melakukan Live Streaming kondisi yang ada di bendungan pada antar muka User Blynk.



Gambar 15. Pengujian Kamera

Hasil pengujian Live Streaming menggunakan kamera OV2640 berhasil berfungsi dan dapat diakses pada aplikasi Blynk, namun masih terdapat delay pada video, harus dilakukan penelitian lebih lanjut menggunakan antena wifi dan kamera dengan kualitas yang lebih baik.

2. Pengujian *Black Box*

Pengujian black-box yaitu pengujian fungsi-fungsi pada alat tersebut, sehingga bisa dilihat sejauh mana alat tersebut bisa berjalan dengan baik. Pengujian sistem pada keseluruhan alat ini dilakukan oleh pengguna/pelanggan.

No	Uraian	Hasil yang diharapkan	Hasil Pengujian	Uji Coba	Berhasil	Gagal
1	Antar muka pengguna	Tampilan program mudah dipahami pengguna	Berhasil menampilkan antar muka yang mudah dipahami	5	5	0
2	Notifikasi	Menampilkan notifikasi sesuai ketentuan	Berhasil memunculkan notifikasi sesuai durasi	5	5	0

Tabel 6 Hasil Pengujian Switch Relay dengan Blynk

No	Uraian	Hasil yang diharapkan	Hasil Pengujian	Uji Coba	Berhasil	Gagal
3	Relay	Mem atikan dan mengaktifkan sensor Ultrasonik	Berhasil mengendalikan ON/OFF Ultrasonik	5	3	2
4	Sensor Ultrasonik	Mengukur ketinggian air	Sensor berhasil mengukur ketinggian air	5	3	2
5	Kam era	Menampilkan video streaming	Berhasil menampilkan video streaming	5	3	2
6	ESP32 - Cam	Dapat terkoneksi ke jaringan <i>W i f i</i> dengan mudah	Alat dapat terkoneksi dengan jaringan yang telah diiri s i a l i s a s i	5	5	0
7	Wem os D1 Mini	Dapat terkoneksi ke jaringan <i>W i f i</i> dengan mudah	Alat dapat terkoneksi dengan jaringan yang telah diiri s i a l i s a s i	5	5	0

Tabel 7 Hasil Pengujian *Switch* Relay dengan Blynk (Tabel Lanjutan)

4. Pengujian *White Box*

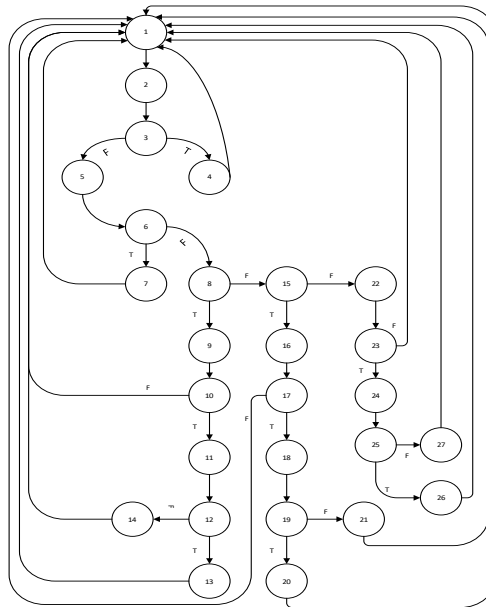
Tujuan utama pengujian white box adalah untuk memastikan bahwa seluruh jalur eksekusi kode telah diuji dan diakses, sehingga berbagai kondisi dan cabang logika diuji untuk menemukan bug dan kelemahan potensial dalam kode. Dengan mengeksplorasi struktur internal kode, pengujian white box membantu meningkatkan cakupan pengujian dan meningkatkan kualitas dan kehandalan perangkat lunak.

No	Code	Ket
1	Long distance, level; Const long intervalSiaga = 15000; Const long intervalAwat = 10000; Const long intervalWaspada = 5000;	Mendefinisikan variabel distance, level, dan interval notifikasi
2	Int thUltrasonik, notifiSensor, notifiOn	Mendefinisikan variabel thUltrasonik, notifiSensor, notifiOn
3	if((distance == 0) && (thUltrasonik == 0)){	Memeriksa nilai distance dan thUltrasonik
4	Blynk.virtualWrite(V5, 0); ledAman.off(); ledSiaga.off(); ledAwat.off(); ledWaspada.off();	Mengirimkan data ke blynk, data led dan level ketinggian = 0
5	else { //menampilkan ketinggian air pada widget level Blynk.virtualWrite(V5, level);	Menampilkan data level pada blynk
6	//LED ketinggian Air & Notifikasi if ((level >= 0)&&(level <= 5)) {	Memeriksa nilai level air 0 – 5 Cm
7	{ ledAman.on(); ledSiaga.off(); ledAwat.off(); ledWaspada.off(); }	Mengirimkan data led ke blynk
8	else if (level >= 6)&&(level <= 10)) {	Memeriksa nilai level air 6 – 10 Cm
9	{ ledAman.off(); ledSiaga.on(); ledAwat.off(); ledWaspada.off();	Mengirim data led pada Blynk
10	if (currentMillis - previousMillis >= intervalSiaga){	Memeriksa data milisecond yang telah berjalan pada mikrokontroler
11	previousMillis = currentMillis;	Memasukkan nilai Milisecond sekarang ke variabel Milisecond sebelumnya
12	if (notifiSensor != 1 && notifiOn == 1){	Memeriksa variabel notifiSensor tidak sama dengan 1 dan notifiOn sama dengan 1
13	notifiSensor = 1; Blynk.logEvent("peringatan_siaga", String("Level Air SIAGA!! ") + level + (" CM"));	Mengaktifkan notifikasi siaga dan mengisi notifiSensor dengan 1

Tabel 8. Pengujian *White Box*

No	Code	Ket
14	} else { notifiSensor = 0; } }	Memberikan nilai notifiSensor = 0
15	else if ((level >= 11)&&(level <= 15))	Memeriksa nilai level 11 - 15
16	{ ledAman.off(); ledSiaga.off(); ledAwat.on(); ledWaspada.off();	Mengaktifkan led Awat
17	if (currentMillis - previousMillis >= intervalAwat){	Memeriksa data milisecond yang telah berjalan pada mikrokontroler
18	previousMillis = currentMillis;	Memasukkan nilai Milisecond sekarang ke variabel Milisecond sebelumnya
19	if (notifiSensor != 2 && notifiOn == 1){	Memeriksa variabel notifiSensor tidak sama dengan 2 dan notifiOn sama dengan 1
20	notifiSensor = 2; Blynk.logEvent("peringatan_awat", String("Level Air AWAS!! ") + level + (" CM"));	Mengaktifkan notifikasi awat dan memasukan nilai notifiSensor = 2
21	} else { notifiSensor = 0; } }	Memberikan nilai notifiSensor = 0
22	else { ledAman.off(); ledSiaga.off(); ledAwat.off(); ledWaspada.on();	Mengaktifkan led waspada
23	if (currentMillis - previousMillis >= intervalWaspada){	Memeriksa data milisecond yang telah berjalan pada mikrokontroler
24	previousMillis = currentMillis;	Memasukkan nilai Milisecond sekarang ke variabel Milisecond sebelumnya
25	if (notifiSensor != 3 && notifiOn == 1){	Memeriksa variabel notifiSensor tidak sama dengan 3 dan notifiOn sama dengan 1
26	notifiSensor = 3; Blynk.logEvent("peringatan_awat", String("Level Air WASPADA!!! ") + level + (" CM"));	Mengaktifkan notifikasi siaga dan mengisi notifiSensor dengan 3
27	} else { notifiSensor = 0; } }	Memberikan nilai notifiSensor = 0

Tabel 9. Pengujian *White Box* (Tabel Lanjutan)



Gambar 16. Flow Graph

5. Maintenance

Setelah tahap testing atau pengujian telah selesai dan prototype yang dibuat berjalan dengan baik, maka prototype selanjutnya akan digunakan pada pengguna. Akan dilakukan instalasi program dan pembelajaran kepada user mengenai cara menggunakan prototype yang telah dibuat. Maintenance akan dilakukan apabila ditemukan adanya kesalahan program, upgrade alat atau mikrokontroler dan maintenance dilakukan untuk menyesuaikan kebutuhan dari user.

5. KESIMPULAN

Berdasarkan uraian-uraian dan hasil pengujian sistem yang telah dijelaskan pada pembahasan maka dapat diambil kesimpulan sebagai berikut :

- 1) Untuk membuat alat peringatan dini ketinggian air bendungan dibutuhkan ESP32-Cam, Wemos D1 Mini, FTDI, Relay, Kamera OV2640 dan aplikasi Blynk.
- 2) Alat peringatan dini ketinggian air bendungan ini memiliki 4 parameter level air, yaitu Aman, Siaga, Awas dan Waspada. Memiliki sistem switch ON/OFF sensor Ultrasonik, sistem Notifikasi berulang dan sistem pemantauan keadaan bendungan dengan Video Live Streaming melalui aplikasi Blynk.
- 3) Berdasarkan hasil pengujian yang telah dilakukan dapat disimpulkan bahwa alat peringatan dini ketinggian air bendungan dapat bekerja dengan baik, walaupun masih terdapat kekurangan pada bagian Live Video Streaming yang masih terkendala masalah koneksi dan kualitas video yang kurang.
- 4) Desain alat yang masih kurang karena rentan korsleting akibat air hujan ataupun jatuh.
- 5) Akurasi sensor Ultrasonik sangat dipengaruhi oleh suara disekitar, sehingga jika ada suara eksternal cukup nyaring akan berpengaruh pada Output sensor Ultrasonik.
- 6) Sensor Ultrasonik tidak bisa membaca data jarak jika objek yang diukur berjarak 1 CM, maka dari itu untuk mendapatkan hasil pengukuran ketinggian air bendungan diperlukan formulasi dan algoritma yang tepat.

6. SARAN

Setelah melakukan penulisan ini dapat diperoleh beberapa hal yang dapat dijadikan saran untuk dapat dilakukan perancangan lebih lanjut yaitu :

- 1) Diharapkan dapat menggunakan jenis kamera yang memiliki kualitas gambar yang baik dibanding kamera OV2640.
- 2) Diharapkan untuk menambahkan komponen antenna yang lebih baik agar tangkapan sinyal Wifi lebih kuat dan dapat menghasilkan Live Video Streaming yang lancar.
- 3) Diharapkan untuk menambahkan fitur Live Video Streaming yang dapat di akses dimanapun tanpa harus terhubung ke jaringan lokal, misalnya dengan menambahkan server atau akses lainnya.
- 4) Diharapkan untuk melakukan penelitian terhadap daya yang digunakan agar dapat menghasilkan alat yang lebih stabil dan mengurangi pengaruh noise dari tegangan, yang mana ini dapat memperpanjang umur komponen alat dan mengurangi kemungkinan error pada alat.

- 5) Diharapkan untuk menambahkan fitur catu daya yang menggunakan baterai dan pengisian daya yang menggunakan panel surya.
- 6) Diharapkan memperbaharui desain alat agar dapat bertahan dari cuaca buruk dan kemungkinan korsleting karena air hujan saat diimplementasikan.
- 7) Diharapkan untuk meneliti pin-pin yang ada pada ESP32-Cam yang dapat digunakan saat kamera diaktifkan sehingga tidak diperlukan lagi mikrokontroler lain untuk mengontrol switch Relay dan Ultrasonik. Hal ini sangat berguna untuk menghemat daya dengan desain alat yang lebih sederhana dan mengurangi biaya pembuatan alat, namun tetap dengan fitur yang sama dan memudahkan teknisi melakukan maintenance.
- 8) Memperbaharui algoritma dan formula pengukuran ketinggian air bendungan agar alat lebih mudah di-maintenance sehingga dapat dipakai di bendungan manapun tanpa harus memodifikasi kode program secara langsung.
- 9) Diharapkan untuk penelitian selanjutnya alat peringatan dini ketinggian air bendungan ini dapat diterapkan pada Bendungan.

7. DAFTAR PUSTAKA.

- Biswas, Shatadru Bipasha, dan M. Tariq Iqbal, 2018. Solar Water Pumping System Control Using a Low Cost ESP32 Microcontroller. 2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE).
- Budiman, Kholiq. 2022. Rekayasa Perangkat Lunak untuk Mahasiswa Ilmu Komputer. Mungkid: Pustaka Rumah C1nta.
- Chamim, 2015. Mikrokontroler Belajar Code Vision AVR Mulai Dari Nol. Yogyakarta: GrahaIlmu.
- Culkin, Jody dan Hagan, Eric. 2017. Make: Learn Electronics With Arduino; An Illustrated Beginner's Guide TO Physical Computing. San Fransisco: Maker Media, Inc.
- Ekojono. Parastiwi, Andriani. Rahmad, Cahya dan Rahmanto, Anugran Nur. 2018. Pemrograman Spreadsheet untuk Pemodelan Kontrol Rangkaian Elektronika. Malang: POLINEMA PRESS.
- Handy dan Susilo, J. (2014). Aplikasi Pengujian White Box IBII Online Judge. Jurnal Informatika dan Bisnis, 3(2), 57.
- Immersa Lab, 2014. Sistem Minimum Mikrokontroler, (<http://immersa-lab.com/sistem-minimum-mikrokontroler.htm>), diakses 7 Desember 2020
- Liang, Qilian. Mu, Jiasong. Wang, Wei dan Zhang Baoju. 2016. Proceedings of the 2015 International Conference on Communications, Signal Processing, and Systems. Berlin Heidelberg: Springer-Verlag.
- Mulyani, Sri. 2016. Metode Analisis dan Perancangan Sistem. Bandung: Abdi Sistematika.
- OmniVision Technologies, Inc. 2015. CMOS OV2640 Camera Module 1/4-Inch 2-Megapixel Module Datasheet.
- Puspasari, Fitri. Fahrurrozi, Imam. Satya, Trias Prima. Setyawan, Galih. Fauzan, Muhammad Rifqi Al dan Admoko, Estu Muhammad Dwi. 2019. Sensor Ultrasonik HCSR04 Berbasis Arduino Due untuk Sistem Monitoring Ketinggian, Jurnal Fisika Dan Aplikasinya. Vol: 15 (3). 37.
- Putra, M. Gilvy Langgawan Putra. Natasia, Sri Rahayu. Wiranti, Yuyun Tri. Octantia, Hemy dan Sadriansyah. 2020. Media Pembelajaran dengan Metode Gamification untuk Meningkatkan Motivasi Pembelajaran pada Perguruan Tinggi di Masa Covid-19. Malang: Media Nusa Creative.
- Rasi, D dan Indirani, M. 2017. Software Engineering. Madurai: Shanlax Publication.
- Sumardi, 2015. Mikrokontroler Belajar AVR Mulai Dari Nol. Yogyakarta: GrahaIlmu
- Sudarso, Hermawan. 2014. "Implementasi Aplikasi Blynk Teknologi IOT".
- Shenzhen Ai-Thinker Technology Co., Ltd. 2017. ESP32-CAM Module. China: Shenzhen Ai-Thinker Technology Co., Ltd
- Sitorus, Lamhot. 2015. Algoritma dan Pemrograman. Yogyakarta: ANDI.
- Sari, Indah Purnama, S.T., M.Kom. 2021. Buku Ajar Rekayasa Perangkat Lunak. Medan: UMSU Press.
- Seneviratne, Pradeeka. 2018. Hands-On Internet of Things with Blynk. Birmingham: Pack Publishing Ltd.
- Serrhini, Mohammed. Silva, Carla dan Aljahdali, Sultan. 2019. Innovation in Information System and Technologies to Support Learning Research. Switzerland: 2020.
- Sugianto, Mikael ST, 2015. Microsoft Visio : Membuat Beragam Desain Diagram Dan Flowchart, Salemba Infotek.
- Wahyudi, Adhie Tri. 2021. Pemrograman Komputer Menggunakan Flowgorithm dan APP Inventor. Surakarta: USB PRESS.

UCAPAN TERIMA KASIH

Terima kasih kepada seluruh dosen dan *staff* STMIK Widya Cipta Dharma dan seluruh kawan-kawan yang telah memberikan memori panjang penuh cerita dimasa perkuliahan.