

Sistem Kontrol Telemetry Banjir di Kelurahan Belian Batam

(Flood Telemetric Control System in Belian Batam)

Helpi Torong¹, Djoko Anwar Mardiono^{2*}, Bambang Apriyanto³, John Friadi⁴

Universitas Batam, Batam^{1,2,3,4}

djoko.anwar@univbatam.ac.id



Riwayat Artikel

Diterima pada 23 Juni 2023

Revisi 1 pada 30 Juni 2023

Revisi 2 pada 3 Juli 2023

Revisi 3 pada 7 Juli 2023

Disetujui pada 12 Juli 2023

Abstract

Purpose: The design of this flood telemetry control system aims to determine the water level in the drainage and provide warnings of potential flooding to the public with direct notifications to applications running on Android smartphones and to reduce the impact of flooding, a flood detection system tool was built using sensors that connected to a microcontroller that will detect the water level.

Methodology: The design of this tool uses 3 (three) sources of information that are installed separately far apart, namely locations A, B, and C connected via the internet network and provide information in the form of monitoring, direct warnings to applications installed on Android Smart Phones.

Result: The results of testing this tool work well. And can find out the water level when the rainy season arrives in the environment around the residence from a distance. This tool provides notifications with a delay of about $\pm 1000\text{ms}$ or 1 second at the smartphone internet access speed to the teleduino webserver or vice versa $\pm 142\text{ms}$.

Conclusion: The system accurately monitors water levels and sends timely flood alerts via Android apps. It performs with a delay of around 1 second, supporting early warning needs.

Limitation: Testing was limited in scope and may not reflect real-world variability such as unstable internet or sensor faults.

Contribution: This tool offers a low-cost, real-time flood monitoring solution to help communities prepare for potential flooding.

Keywords: Ethernet Shield, Limit Switch, MIT App Inventor.

How to cite: Torong, H., Mardiono, D. A., Apriyanto, B., Friadi, J. (2023). Sistem Kontrol Telemetry Banjir di Kelurahan Belian Batam. *Jurnal Teknologi Riset Terapan*, 1(2), 55-67.

1. Pendahuluan

Kota Batam merupakan salah satu kota di Provinsi Kepulauan Riau yang perkembangannya cukup pesat yang secara geografis memiliki letak yang sangat strategis karena berada pada jalur pelayaran internasional dan hanya berjarak 12,5 mil laut dengan negara tetangga Singapura (Nurhidayati, Arifiya, Setiawan, Larasakti, & Heriansyah, 2022). Posisi yang strategis ini menempatkan Kota Batam sebagai pintu gerbang pembangunan ekonomi, baik skala propinsi maupun nasional (Hernawan & Srimulyani, 2022). Pertumbuhan kota dan perkembangan industri menimbulkan dampak yang cukup besar pada siklus hidrologi sehingga berpengaruh besar terhadap sistem drainase perkotaan. Sebagai contoh adalah perkembangan kawasan hunian yang disinyalir sebagai penyebab banjir dan genangan di lingkungan sekitarnya (Riski & Widiana, 2020). Hal ini disebabkan karena perkembangan urbanisasi yang menyebabkan perubahan tata guna lahan (Oktaviani & Sembiring, 2022). Oleh karena itu perkembangan kota harus diikuti dengan peningkatan dan perbaikan sistem drainase, selain itu saluran drainase yang telah ada pun efektifitasnya telah berkurang karena adanya pembuangan sampah di saluran drainase (Masbullah & Bahri, 2021), saluran drainase juga kurang

mendapatkan pemeliharaan Seperti tumbuhnya rumput liar dan banyaknya sampah di saluran drainase sehingga menghambat aliran air saat hujan turun. Akibatnya setiap musim hujan air dari saluran drainase meluap membanjiri daerah pemukiman, jalan raya, ataupun fasilitas umum disekitar saluran drainase (Mardiono, Nanra, & Rican, 2023). Berdasarkan latar belakang masalah yang telah dipaparkan di atas, maka dibuat sebuah alat monitoring ketinggian air yang akan dipasang pada saluran drainase untuk memberikan informasi banjir kepada masyarakat, khususnya perumahan Taman Raya Tahap II Kelurahan Belian Kota Batam (Melvi, Ulvan, Sidiq, & Batubara, 2023). Tujuannya adalah untuk mengetahui ketinggian air pada drainase, dan memberikan peringatan potensi terjadinya banjir dengan notifikasi langsung ke aplikasi yang dijalankan pada smartphone android. Dengan adanya alat sistem telemetri banjir ini diharapkan alat ini dapat digunakan pada masyarakat umum yang tempat tinggalnya sering terjadi banjir, guna langkah persiapan atas bencana banjir yang akan terjadi (Chania, Sara, & Sadalia, 2022; Novianti, Akhyari, Hakim, & Saputra, 2021).

2. Tinjauan Pustaka

Networking merupakan “jaringan antar komputer” yang menghubungkan “suatu komputer” dengan “komputer lainnya” di dalam jaringan oleh “alat-alat jaringan”. Dalam dunia komputer, *Networking* adalah praktek menghubungkan dua atau lebih perangkat bersama-sama untuk tujuan berbagi (*sharing*) data serta berbagai manfaat lainnya (*resources*). Jaringan (*Networking*) dibangun dengan perangkat keras komputer (*hardware*) atau fisik dan perangkat lunak komputer (*software*) logik (Adjie, 2015). Android adalah sistem operasi untuk perangkat *mobile* berbasis linux yang awalnya dikembangkan oleh Android Inc. Android terdiri dari sistem operasi, *middleware* dan aplikasi. Android adalah istilah bahasa inggris yang berarti robot yang menyerupai manusia (Melvi, Nurhayati, Batubara, Septama, & Ulvan, 2023). Pada tahun 20015, Google secara resmi telah membeli Android, sehingga perkembangan Android sepenuhnya berada ditangan Google. Dalam proses pengembangan sistem aplikasi Android, dibentuklah organisasi Open Handset Allisnce. Google merilis *software open source* untuk Android, sehingga dapat berkontribusi untuk mengembangkan Android (Rachmawati, Nugraha, & Awaluddin, 2017).

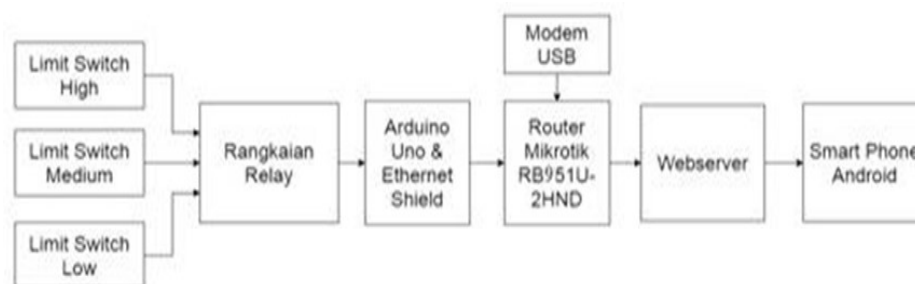
MIT App Inventor merupakan IDE generasi kedua dari App Inventor yang dikelola oleh Massachusetts Institute of Technology (MIT). MIT App Inventor berbasis cloud yang diakses menggunakan internet browser. Masuk kategori dalam visual programming, MIT App Inventor menggunakan *block puzzle* yang disusun untuk menjadi rangkaian kode (Utomo, Azizah, & Pangestu, 2022). MIT App Inventor memiliki 3 bagian utama, *Component Designer*, *Block Editor* dan *Android Device* yang digunakan untuk pengujian. Pengujian bisa menggunakan emulator maupun perangkat sebenarnya. Untuk perangkat sebenarnya bisa dihubungkan melalui jaringan *wireless* dan menggunakan USB (Wihidayat, 2017). Teleduino adalah Web Server yang digunakan untuk menghubungkan arduino dengan aplikasi lain menggunakan koneksi internet. Web Server ini dapat dapat dijadikan sebagai media koneksi pengontrolan atau monitoring antara mikrokontroller dengan perangkat lain yang terhubung ke jaringan internet berdasarkan API Key yang disediakan oleh Web Server Teleduino (Melvi, Nurhayati, et al., 2023). Setiap perangkat yang akan dihubungkan ke Web Server teleduino harus merequest api key, kemudian akan di masukkan kedalam mikrokontroller, sebagai jalur akses dari Teleduino ke perangkat tersebut. Untuk mengakses teleduino bias dengan cara membuka situs teleduino resmi yaitu <https://www.teleduino.org/> dari web browser (Hesti & Adewasti, 2018).

Router adalah sebuah alat yang mengirimkan paket data melalui sebuah jaringan atau Internet menuju tujuannya, melalui sebuah proses yang dikenal sebagai routing (Riesna, Pujiyanto, Efendi, Nugroho, & Saputra, 2023). Proses routing terjadi pada lapisan 3 (Lapisan jaringan seperti Internet *Protocol*) dari *stack* protokol tujuh lapis OSI. Router memiliki fasilitas *DHCP* (*Dynamic Host Configuration Procotol*), dengan mensetting *DHCP*, maka kita dapat membagi *IP Address*, fasilitas lain dari Router adalah adanya *NAT* (*Network Address Translator*) yang dapat memungkinkan suatu *IP Address* atau koneksi internet disharing ke *IP Address* lain (Adjie, 2015). *API* adalah singkatan dari *Application Programming Interface*, dan memungkinkan *developer* untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan. API terdiri dari berbagai elemen seperti *function*, *protocols*, dan *tools* lainnya yang

memungkinkan *developers* untuk membuat aplikasi. Tujuan penggunaan API adalah untuk mempercepat proses *development* dengan menyediakan *function* secara terpisah sehingga *developer* tidak perlu membuat fitur yang serupa (Dalena, Ali, & Ediwarman, 2022). Penerapan API akan sangat terasa jika fitur yang diinginkan sudah sangat kompleks, tentu membutuhkan waktu untuk membuat yang serupa dengannya. Misalnya: integrasi dengan *payment gateway*. Terdapat berbagai jenis sistem API yang dapat digunakan, termasuk sistem operasi, *library*, dan web (Sandi, 2017). Winbox adalah utility yang digunakan untuk konektivitas dan konfigurasi MikroTik menggunakan MAC Address atau protokol IP. Dengan winbox kita dapat melakukan konfigurasi MikroTik RouterOS menggunakan modus GUI dengan cepat dan sederhana. Winbox dibuat menggunakan win32 binary tapi dapat dijalankan pada Linux, Mac OSX dengan menggunakan Wine. Semua fungsi winbox didesain dan dibuat semirip dan sedekat mungkin dengan fungsi *console*, sehingga Anda akan menemukan istilah-istilah yang sama pada fungsi console. (Yuniarahmah & <https://dwiyunia12.wordpress.com/>, 2018).

3. Metode penelitian

Perancangan ini terdiri dari dua bagian, yaitu perangkat keras (hardware) dan perangkat lunak (software) (Dharma, Agustina, & Windah, 2021). perangkat keras meliputi pembuatan desain peralatan mikrokontroller, dan peralatan jaringan yang akan dipasang pada 3 lokasi yang berbeda. dimana peralatan mikrokontroller yang dipergunakan adalah limit switch, dan rangkaian relay sebagai pendeteksi ketinggian air, dan arduino uno, serta Ethernet Shield sebagai media penerima dan pengolahan signal yang didapat dari sensor level air dan mengirimkannya ke Web Server melalui media perangkat jaringan, dimana perangkat jaringan yang dipergunakan adalah Router Board Mikrotik RB 951 dan modem usb sebagai fungsi pencampuran jaringan lokal dan internet yang akan diberikan kepada perangkat Ethernet Shield agar bisa terkoneksi ke Web Server. Sedangkan perangkat lunak meliputi pembuatan aplikasi yang akan dijalankan pada smartphone dengan os android. Pada gambar 1 ditunjukkan konfigurasi dari sistem telemetri banjir pada satu lokasi dimana limit switch sebagai input untuk mendeteksi level ketinggian air mencapai level Low, Medium, dan High kemudian sinyal yang didapat diarahkan ke rangkaian relay yang outputnya telah diberikan supply 5V dc untuk memanipulasi logika 0 atau 1, kemudian logika yang didapat dari output rangkaian relai akan dihubungkan ke GPIO Arduino uno dan Ethernet Shield, data yang didapat kemudian diberikan ke Web Server melalui akses jaringan Router Board (Yando, Panusunan, & Fauzan, 2023). Kemudian interface aplikasi akan dibuat dengan mengarahkan ke data Web Server untuk memperoleh informasi serta notifikasi level ataupun kondisi ketinggian air secara realtime.



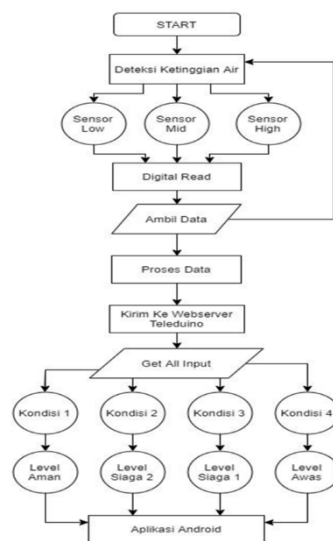
Gambar 1. Blok Diagram Sistem Telemetri Pada Satu Lokasi

Gambar di atas menunjukkan blok diagram sistem telemetri yang terdiri dari:

1. Limit Switch yang dipergunakan adalah limit switch *waterproof*, terdiri dari 3 limit switch yang outputnya dihubungkan ke rangkaian relay untuk mendeteksi 3 kemungkinan level air yaitu *Low*, *Medium*, dan *High*.
2. Rangkaian Relay yang dipergunakan adalah rangkaian relay 4 jalur dengan supply 5V DC, dimana rangkaian ini akan dihubungkan ke Arduino Uno dan Ethernet Shield untuk memanipulasi logika 1 atau 0, dikarenakan kurang efesienya tegangan yang didapat dari limit switch jika langsung dihubungkan pada Perangkat Mikrokontroller.
3. Arduino Uno dan Ethernet Shield berfungsi sebagai rangkaian yang menerima dan mengolah data yang diterima dari 3 relay dengan logika masing-masing GPIO.

4. Modem USB berfungsi sebagai akses internet yang didapat dari jaringan provider sim card yang dipasang pada modem, dan memberikan akses internet tersebut ke Router.
5. Router Mikrotik RB 951 2HnD, Router ini memiliki fungsi sebagai interaksi jaringan lokal client Ethernet Shield dan mengkonfigurasi jaringan tersebut dengan internet untuk mengakses ke Web Server Teleduino.
6. Web Server berfungsi sebagai media yang dapat menerima command dan mengarahkannya melalui proxy berdasarkan API Key yang sudah ditanamkan ke Ethernet Shield .
7. Smartphone Android berfungsi sebagai interface monitoring serta media notifikasi aplikasi yang telah dibuat dari MIT App Inventor dengan membaca keadaan dari semua limit switch yang telah terpasang pada Ethernet Shield Arduino melalui akses Web Server Teleduino.

Berdasarkan Diagram Alir di bawah diketahui bahwa program diawali dengan inisialisasi dengan sistem digital read yang didapat dari logika 0 atau 1 tergantung pada kondisi ketiga sensor ketinggian air yang terpasang pada bagian mikrokontroller, kemudian logika tersebut akan dikirim ke Web Server Teleduino dengan proxy berdasarkan API Key yang telah terdaftar dan diflash pada rangkaian Ethernet Shield dan Arduino.



Gambar 2. Diagram Alir sistem kontrol telemetry pada satu Lokasi

Pada kondisi 1 sensor Low terdeteksi logika 0 dikarenakan air belum mencapai sensor tersebut, maka pada kondisi ini air juga belum mencapai sensor Mid dan High maka sensor Mid dan High akan memberikan logika 0. Dari kondisi 1 didapatkan Logika 0,0,0 maka kondisi 1 tersebut Aplikasi akan memberikan monitoring serta notifikasi dengan level Aman (Salim, Jatnika, & Yudiana, 2023). Pada Kondisi 2 sensor Low terdeteksi logika 1 dikarenakan air sudah mencapai sensor tersebut, namun pada kondisi ini air belum mencapai sensor Mid dan High maka sensor Mid dan High akan memberikan logika 0. Dari kondisi 2 didapatkan Logika 1,0,0 maka dari logika kondisi 2 tersebut Aplikasi akan memberikan monitoring serta notifikasi dengan level Siaga 2. Pada Kondisi 3 sensor Low, dan sensor Mid terdeteksi logika 1 dikarenakan air sudah mencapai sensor Mid, namun pada kondisi ini air belum mencapai sensor High maka sensor High akan memberikan logika 0. Dari kondisi 3 didapatkan Logika 1,1,0 maka dari logika kondisi 3 tersebut Aplikasi akan memberikan monitoring serta notifikasi dengan level Siaga 1. Pada Kondisi 4 sensor Low, Sensor Mid dan sensor High terdeteksi logika 1 dikarenakan air sudah mencapai sensor High. Dari kondisi 4 didapatkan Logika 1,1,1 maka dari logika kondisi 4 tersebut Aplikasi akan memberikan dengan monitoring serta notifikasi dengan level Awas (Chika, Oshiohwemoh, & Promise, 2022). Pada perangkat jaringan terdiri dari Modem USB (Support Mikrotik), Router Mikrotik RB 951, dan kabel belden dengan soket RJ45. Dimana modem USB berfungsi sebagai media pembaca akses jaringan internet dari sim card GSM provider, kemudian jaringan internet tersebut akan diarahkan dan dikonfigurasi dari Router RB 951 melalui jaringan lokal agar semua client dari Router yang terhubung ke slot Ethernet Router bisa terkoneksi ke jaringan internet. Modem yang digunakan pada alat ini adalah Modem USB model

Arduino Internet / Web Control x Teledune Unique API Key - Home x

teledune.org/tools/request-key

Apps Settings Amazon MIT App Inventor ESI - ADD3M4TCO Teledune ECU-Linker Pinwiz

Introduction
Request Key
 Tools
 Documentation
 Downloads
 Terms and Conditions
 Sponsors
 Contact

Request Key

A key is required to uniquely identify your Teledune device.

Keys are generated and emailed to you within a few minutes.

Your privacy is protected, and your email address is not passed on to third parties.

Please fill in the below form to request a key. If you require multiple keys for multiple devices, you can complete the form multiple times using the same email address.

First Name:

Last Name:

Email Address:

Are you human? (type "yes")

☒ I accept the Teledune Terms and Conditions

Submit Request

```

File Edit Sketch Tools Help

Teleduino328EthernetClientProxy | Arduino 1.9.0-beta-1

// Teleduino328EthernetClientProxy

#include <Servo.h>
#include <Ethernet.h>
#include <EthernetUdp.h>
#include <SPI.h>
#include <Wire.h>

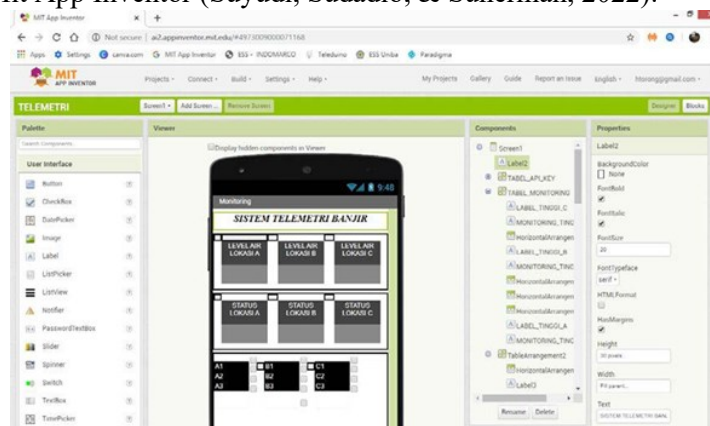
// User configurable variables
byte useudp = 1; //
byte usecdns = true;
byte use[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
Teleduino deviceloc(102, 100, 1, 100); // Only if useudp is false
Teleduino gateway(102, 100, 1, 1); // Only if useudp is false
Teleduino sub(102, 100, 1, 1); // Only if useudp is false
Teleduino sub2(255, 255, 255, 0); // Only if useudp is false
Teleduino serverloc(170, 230, 152, 170); // Only if usecdns is false
char servername[] = "us01.proxy.teleduino.org"; // Only if usecdns is true
unsigned int serverport = 5303; // Can be set to either 53 or 5355
byte statusledPin = 8;

// User configurable key, this is used to authenticate with the proxy server
// This is checked against the EEPROM on boot and written if necessary
// The proxy server retrieves the key from the EEPROM
byte key[] = { 0x27, 0x00, 0x00, 0x00,
              0x1C, 0x19, 0x20, 0x00,
              0x00, 0x07, 0x00, 0x00,
              0x07, 0x00, 0x00, 0x00 };

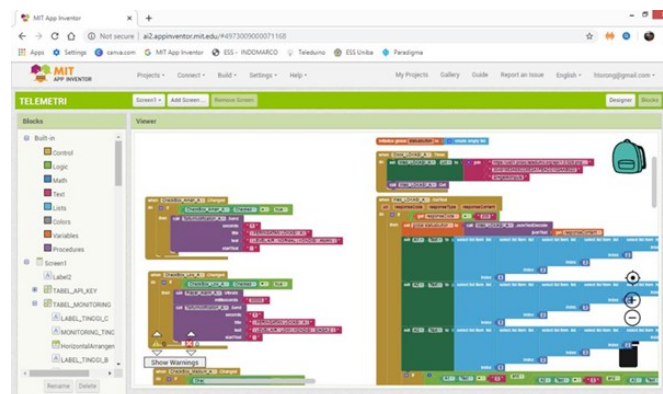
// Global variables
int use = 0;
int use2 = 0;
int use3 = 0;
int use4 = 0;
int use5 = 0;
int use6 = 0;
int use7 = 0;
int use8 = 0;
int use9 = 0;
int use10 = 0;
int use11 = 0;
int use12 = 0;
int use13 = 0;
int use14 = 0;
int use15 = 0;
int use16 = 0;
int use17 = 0;
int use18 = 0;
int use19 = 0;
int use20 = 0;
int use21 = 0;
int use22 = 0;
int use23 = 0;
int use24 = 0;
int use25 = 0;
int use26 = 0;
int use27 = 0;
int use28 = 0;
int use29 = 0;
int use30 = 0;
int use31 = 0;
int use32 = 0;
int use33 = 0;
int use34 = 0;
int use35 = 0;
int use36 = 0;
int use37 = 0;
int use38 = 0;
int use39 = 0;
int use40 = 0;
int use41 = 0;
int use42 = 0;
int use43 = 0;
int use44 = 0;
int use45 = 0;
int use46 = 0;
int use47 = 0;
int use48 = 0;
int use49 = 0;
int use50 = 0;
int use51 = 0;
int use52 = 0;
int use53 = 0;
int use54 = 0;
int use55 = 0;
int use56 = 0;
int use57 = 0;
int use58 = 0;
int use59 = 0;
int use60 = 0;
int use61 = 0;
int use62 = 0;
int use63 = 0;
int use64 = 0;
int use65 = 0;
int use66 = 0;
int use67 = 0;
int use68 = 0;
int use69 = 0;
int use70 = 0;
int use71 = 0;
int use72 = 0;
int use73 = 0;
int use74 = 0;
int use75 = 0;
int use76 = 0;
int use77 = 0;
int use78 = 0;
int use79 = 0;
int use80 = 0;
int use81 = 0;
int use82 = 0;
int use83 = 0;
int use84 = 0;
int use85 = 0;
int use86 = 0;
int use87 = 0;
int use88 = 0;
int use89 = 0;
int use90 = 0;
int use91 = 0;
int use92 = 0;
int use93 = 0;
int use94 = 0;
int use95 = 0;
int use96 = 0;
int use97 = 0;
int use98 = 0;
int use99 = 0;
int use100 = 0;
int use101 = 0;
int use102 = 0;
int use103 = 0;
int use104 = 0;
int use105 = 0;
int use106 = 0;
int use107 = 0;
int use108 = 0;
int use109 = 0;
int use110 = 0;
int use111 = 0;
int use112 = 0;
int use113 = 0;
int use114 = 0;
int use115 = 0;
int use116 = 0;
int use117 = 0;
int use118 = 0;
int use119 = 0;
int use120 = 0;
int use121 = 0;
int use122 = 0;
int use123 = 0;
int use124 = 0;
int use125 = 0;
int use126 = 0;
int use127 = 0;
int use128 = 0;
int use129 = 0;
int use130 = 0;
int use131 = 0;
int use132 = 0;
int use133 = 0;
int use134 = 0;
int use135 = 0;
int use136 = 0;
int use137 = 0;
int use138 = 0;
int use139 = 0;
int use140 = 0;
int use141 = 0;
int use142 = 0;
int use143 = 0;
int use144 = 0;
int use145 = 0;
int use146 = 0;
int use147 = 0;
int use148 = 0;
int use149 = 0;
int use150 = 0;
int use151 = 0;
int use152 = 0;
int use153 = 0;
int use154 = 0;
int use155 = 0;
int use156 = 0;
int use157 = 0;
int use158 = 0;
int use159 = 0;
int use160 = 0;
int use161 = 0;
int use162 = 0;
int use163 = 0;
int use164 = 0;
int use165 = 0;
int use166 = 0;
int use167 = 0;
int use168 = 0;
int use169 = 0;
int use170 = 0;
int use171 = 0;
int use172 = 0;
int use173 = 0;
int use174 = 0;
int use175 = 0;
int use176 = 0;
int use177 = 0;
int use178 = 0;
int use179 = 0;
int use180 = 0;
int use181 = 0;
int use182 = 0;
int use183 = 0;
int use184 = 0;
int use185 = 0;
int use186 = 0;
int use187 = 0;
int use188 = 0;
int use189 = 0;
int use190 = 0;
int use191 = 0;
int use192 = 0;
int use193 = 0;
int use194 = 0;
int use195 = 0;
int use196 = 0;
int use197 = 0;
int use198 = 0;
int use199 = 0;
int use200 = 0;
int use201 = 0;
int use202 = 0;
int use203 = 0;
int use204 = 0;
int use205 = 0;
int use206 = 0;
int use207 = 0;
int use208 = 0;
int use209 = 0;
int use210 = 0;
int use211 = 0;
int use212 = 0;
int use213 = 0;
int use214 = 0;
int use215 = 0;
int use216 = 0;
int use217 = 0;
int use218 = 0;
int use219 = 0;
int use220 = 0;
int use221 = 0;
int use222 = 0;
int use223 = 0;
int use224 = 0;
int use225 = 0;
int use226 = 0;
int use227 = 0;
int use228 = 0;
int use229 = 0;
int use230 = 0;
int use231 = 0;
int use232 = 0;
int use233 = 0;
int use234 = 0;
int use235 = 0;
int use236 = 0;
int use237 = 0;
int use238 = 0;
int use239 = 0;
int use240 = 0;
int use241 = 0;
int use242 = 0;
int use243 = 0;
int use244 = 0;
int use245 = 0;
int use246 = 0;
int use247 = 0;
int use248 = 0;
int use249 = 0;
int use250 = 0;
int use251 = 0;
int use252 = 0;
int use253 = 0;
int use254 = 0;
int use255 = 0;
int use256 = 0;
int use257 = 0;
int use258 = 0;
int use259 = 0;
int use260 = 0;
int use261 = 0;
int use262 = 0;
int use263 = 0;
int use264 = 0;
int use265 = 0;
int use266 = 0;
int use267 = 0;
int use268 = 0;
int use269 = 0;
int use270 = 0;
int use271 = 0;
int use272 = 0;
int use273 = 0;
int use274 = 0;
int use275 = 0;
int use276 = 0;
int use277 = 0;
int use278 = 0;
int use279 = 0;
int use280 = 0;
int use281 = 0;
int use282 = 0;
int use283 = 0;
int use284 = 0;
int use285 = 0;
int use286 = 0;
int use287 = 0;
int use288 = 0;
int use289 = 0;
int use290 = 0;
int use291 = 0;
int use292 = 0;
int use293 = 0;
int use294 = 0;
int use295 = 0;
int use296 = 0;
int use297 = 0;
int use298 = 0;
int use299 = 0;
int use300 = 0;
int use301 = 0;
int use302 = 0;
int use303 = 0;
int use304 = 0;
int use305 = 0;
int use306 = 0;
int use307 = 0;
int use308 = 0;
int use309 = 0;
int use310 = 0;
int use311 = 0;
int use312 = 0;
int use313 = 0;
int use314 = 0;
int use315 = 0;
int use316 = 0;
int use317 = 0;
int use318 = 0;
int use319 = 0;
int use320 = 0;
int use321 = 0;
int use322 = 0;
int use323 = 0;
int use324 = 0;
int use325 = 0;
int use326 = 0;
int use327 = 0;
int use328 = 0;
int use329 = 0;
int use330 = 0;
int use331 = 0;
int use332 = 0;
int use333 = 0;
int use334 = 0;
int use335 = 0;
int use336 = 0;
int use337 = 0;
int use338 = 0;
int use339 = 0;
int use340 = 0;
int use341 = 0;
int use342 = 0;
int use343 = 0;
int use344 = 0;
int use345 = 0;
int use346 = 0;
int use347 = 0;
int use348 = 0;
int use349 = 0;
int use350 = 0;
int use351 = 0;
int use352 = 0;
int use353 = 0;
int use354 = 0;
int use355 = 0;
int use356 = 0;
int use357 = 0;
int use358 = 0;
int use359 = 0;
int use360 = 0;
int use361 = 0;
int use362 = 0;
int use363 = 0;
int use364 = 0;
int use365 = 0;
int use366 = 0;
int use367 = 0;
int use
```

Pengoperasian Web Server Teleduino dan Flash Program Arduino IDE dan untuk mengakses Web Server Teleduino melalui link <http://Teleduino.org>. Untuk meminta API key bisa menekan menu

Request Key, dan mengisi form dengan data yang diberikan oleh Teleduino, kemudian klik Submit Request. Maka API Key akan dikirim melalui email, sesuai dengan email yang isi pada table email address. Untuk proses generate API Key, klik menu Tools, Arduino Sketch Key, masukan API Key pada kolom key, lalu klik generate. Bite key ini akan diinput pada Sketch Teleduino 328 dari aplikasi Arduino IDE, sebelum diupload ke Rangkaian mikrokontroler (Darmawan & Roba'in, 2022). Setelah library Teleduino 328 diimport, buka Sketch Teleduino328 Ethernet Client Proxy, klik file, Teleduino328, Ethernet Client Proxy, pada kolom byte key rubah dengan hasil generate API Key. Gambar di atas adalah tampilan dari proses Upload Sketch ke Rangkaian Ethernet Shield dan Arduino UNO. Berdasar Flowcart di atas diketahui bahwa aplikasi akan berjalan jika Smartphone android telah mendapat akses jaringan internet, setelah itu aplikasi akan mengakses Proxy melalui API Key yang telah disesuaikan pada aplikasi selama 1 detik sekali untuk mengambil data dari sistem getAllInput, dengan membaca logika pada masing Sensor yang telah terhubung ke Ethernet Shield tiap-tiap lokasi. Kemudian Aplikasi akan memberikan informasi berupa monitoring serta notifikasi setiap kali terdapat perubahan pada data yang didapat untuk setiap lokasinya. Pembuatan aplikasi berbasis android pada alat ini memakai Mit App Inventor (Suyudi, Sudadio, & Suherman, 2022).



Gambar 6. Design Project Aplikasi



Gambar 7. Blok Program aplikasi

4. Hasil dan Pembahasan

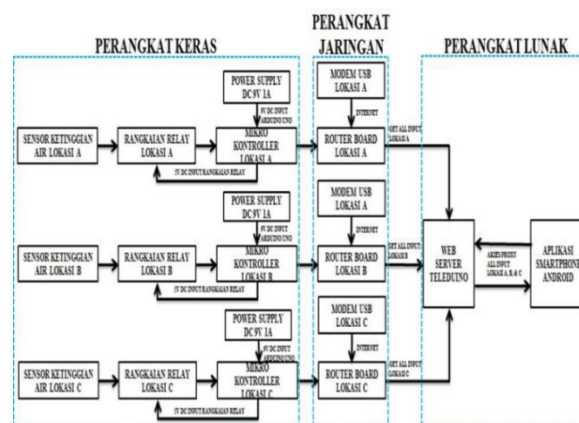
Pengujian dilakukan terhadap perangkat yang telah dibuat secara keseluruhan dilakukan dengan mengintegrasikan seluruh sistem kerja pada alat dari perangkat keras, perangkat jaringan, dan perangkat lunak/software yang digunakan. Pengujian sistem ini meliputi pengujian alat yang telah dikalibrasi dan diketahui bahwa tiap bagian sistem dapat bekerja dengan baik. Pengujian dilakukan untuk mengetahui kondisi normal saat alat yang direncanakan dan dirangkai bekerja sesuai dengan harapan.

4.1 Pengujian Perangkat Keras

Pengujian ini meliputi dari pengecekan seluruh perangkat keras, mulai dari seluruh limit switch yang difungsikan sebagai sensor pendeteksi ketinggian air dari setiap lokasi, rangkaian relay yang digunakan sebagai perantara sinyal dari limit switch ke perangkat mikrokontroller, serta perangkat mikrokontroller Ethernet shield dan Arduino Uno yang difungsikan sebagai perangkat yang menampung data serta mengirimkan data-data tersebut ke Web Server.

4.2 Pengujian Sensor Ketinggian Air

Pengujian ini bertujuan untuk mengetahui apakah sensor sudah bekerja sesuai fungsinya. Pengujian sensor dilakukan dengan mengukur koneksi dari limit switch Low, Medium, dan High pada lokasi A, B, dan C dengan cara memasukan rangkaian limit switch ke dalam media air. Pengujian ini dilakukan untuk mengetahui apakah limit switch yang terpasang sudah sesuai urutannya dengan perancangan yang seharusnya terdapat 3 kondisi pada masing-masing lokasi, yaitu : Kondisi Low, dimana media air pada lokasi tersebut telah menyentuh limit switch Low, sehingga output pin limit switch Low menjadi terhubung. Kondisi Medium, dimana media air menyentuh limit switch Low, dan limit switch Medium sehingga output pin limit switch Low, dan Medium menjadi terhubung : Kondisi High, dimana media air telah menyentuh limit switch Low, Medium, dan High sehingga output pin ketiga limit switch menjadi terhubung. Persentase level air diperoleh dari ukuran penyangga sensor yang panjangnya 41cm, sensor Low 7cm, sensor Mid 21cm, dan sensor High 36cm. Dari tabel pengujian sensor ketinggian air di atas Output limit switch dari ketiga lokasi telah sesuai dengan perancangan. Dimana output dari tiap-tiap limit switch akan dihubungkan kerangkaian relay disetiap lokasi.



Gambar 8. Blok diagram keseluruhan



Gambar 9. Pengujian sensor ketinggian air

Tabel 1. Hasil Pengujian sensor air Lokasi A

Level Air	OutPut Pin Limit Switch Lokasi A		
	Limit Switch Low	Limit Switch Medium	Limit Switch High
17%	Terhubung	Terputus	Terputus
51%	Terhubung	Terhubung	Terputus
88%	Terhubung	Terhubung	Terhubung

Tabel 2. Hasil Pengujian sensor air Lokasi B

Level Air	OutPut Pin Limit Switch Lokasi B		
	Limit Switch Low	Limit Switch Medium	Limit Switch High
17%	Terhubung	Terputus	Terputus
57%	Terhubung	Terhubung	Terputus
88%	Terhubung	Terhubung	Terhubung

Tabel 3. Hasil Pengujian sensor air Lokasi C

Level Air	OutPut Pin Limit Switch Lokasi C		
	Limit Switch Low	Limit Switch Medium	Limit Switch High
17%	Terhubung	Terputus	Terputus
57%	Terhubung	Terhubung	Terputus
88%	Terhubung	Terhubung	Terhubung

4.3 Pengujian Rangkaian Relay

Pengujian ini dilakukan untuk memastikan bahwasanya rangkaian relay yang telah dimodifikasi dapat beroperasi dengan normal sesuai dengan perancangan, yaitu: keadaan relay NO atau NC. Keadaan relay NO artinya *Normally Open* seharusnya output dari relay memberikan tegangan 0V DC atau logika 0, sedangkan keadaan NC artinya *Normally Close* seharusnya output relay memberikan tegangan 5V DC atau berlogika 1. Tahapan pengujian ini dilakukan dengan cara mengaktifkan limit switch *Low*, *Medium*, & *High*.



Gambar 10. Pengujian Rangkaian Relay

Tabel 4. Hasil Pengujian Rangkaian Relay Lokasi A

LOKASI A			
Tegangan Limit Switch	Kondisi Low A	Kondisi Medium A	Kondisi High A
Tegangan Com – NO (V) Relay Low A	4.89	4.84	4.77
Tegangan Com – NO (V) Relay Medium A	0	4.84	4.77
Tegangan Com – NO (V) Relay High A	0	0	4.77

Tabel 5. Hasil Pengujian Rangkaian Relay Lokasi B

LOKASI B			
Tegangan Limit Switch	Kondisi Low B	Kondisi Medium B	Kondisi High B
Tegangan Com – NO (V) Relay Low B	4.86	4.81	4.74
Tegangan Com – NO (V) Relay Medium B	0	4.81	4.74
Tegangan Com – NO (V) Relay High B	0	0	4.74

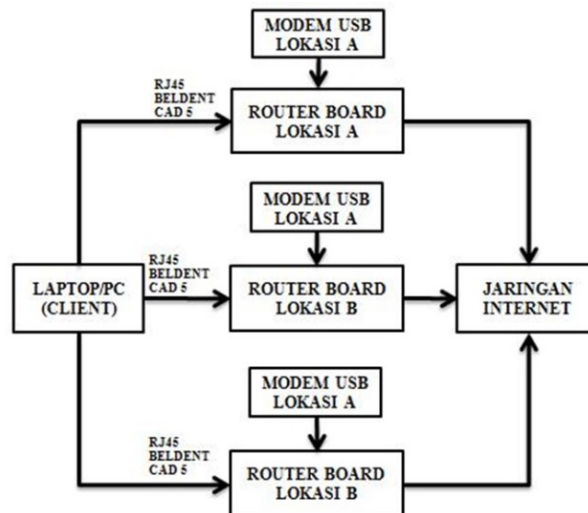
Tabel 6. Hasil Pengujian Rangkaian Relay Lokasi C

LOKASI C			
Tegangan Limit Switch	Kondisi Low C	Kondisi Medium C	Kondisi High C
Tegangan Com – NO (V) Relay Low C	4.92	4.87	4.80
Tegangan Com – NO (V) Relay Medium C	0	4.87	4.80
Tegangan Com – NO (V) Relay High C	0	0	4.80

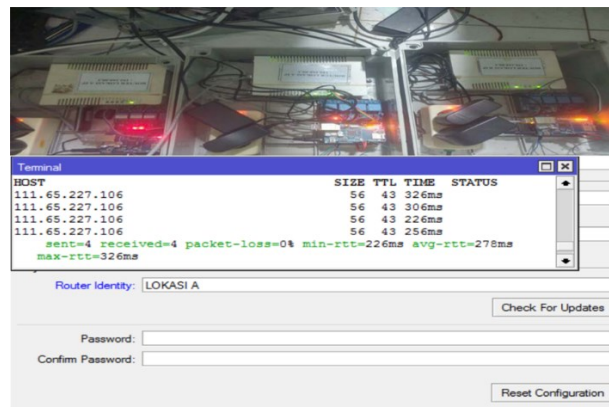
Pengujian ini menggunakan multimeter dengan mengukur tegangan pin Com dan NO pada relay pada ketiga kondisi *Low*, *Medium*, dan *High* pada setiap lokasinya. Dari hasil pengukuran tabel di atas terdapat 3 kemungkinan pada setiap lokasi terhadap masing - masing relay, pada lokasi A kondisi Low, Medium, dan High. Pada kondisi Low tegangan Com – NO relay Low 4.89V, relay Medium 0.0V, dan relay High 0.0V, maka didapatkan logika 1,1,0. Pada kondisi Medium relay Low 4.84V, relay Medium 4.84V, dan relay High 0.0V, maka didapatkan logika 1,1,0. Pada kondisi High relay Low 4.77V, relay Medium 4.77V, dan relay High 4.77V, maka didapatkan logika 1,1,1. Hasil pengukuran tabel di atas mempunyai nilai berbeda pada masing-masing kondisi, perbedaan ini dikarenakan tegangan pada power supply yang terpakai untuk mensupply coil pada relay.

4.4 Pengujian Perangkat Jaringan

Pengujian ini dilakukan untuk memastikan apakah semua perangkat jaringan telah terkoneksi sesuai dengan topologi pada perancangan, dimana semua client yang terhubung pada port Ethernet Router Board dari setiap lokasi dapat terkoneksi ke jaringan internet dengan menggunakan Modem USB (Riesna et al., 2023). Dimana simcard yang dipergunakan memakai provider Telkomsel. Tahap ini dilakukan dengan cara menghubungkan laptop pada masing-masing routerboard setiap lokasi, dan mengakses terminal router dengan memberikan perintah ping ke hosting Web Server Teleduino.org, pengujian ini akan dilakukan dengan periode selama 24 jam, dengan pengambilan data per 3 jam sekali.



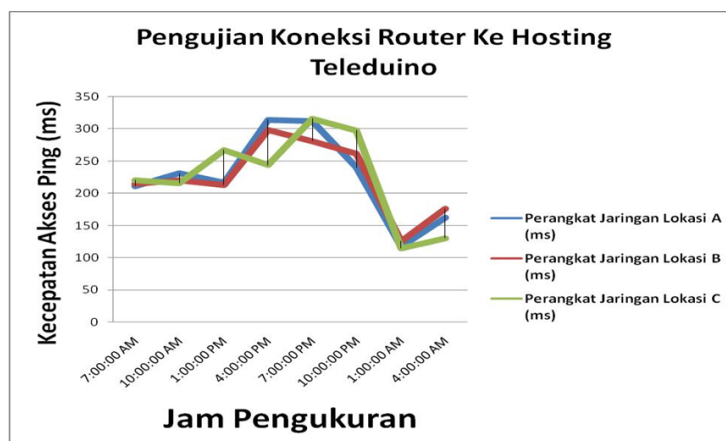
Gambar 11. Blok diagram Pengujian Perangkat Jaringan



Gambar 12. Pengujian Perangkat Jaringan

Tabel 7. Hasil Pengujian Koneksi Perangkat Jaringan

Pengujian Koneksi Router Ke Hosting Teleduino				
Jam	Perangkat Jaringan Lokasi A (ms)	Perangkat Jaringan Lokasi B (ms)	Perangkat Jaringan Lokasi C (ms)	Rata-Rata Koneksi Jaringan A, B, & C (ms)
7:00:00 AM	211	215	221	223.3333333
10:00:00 AM	231	220	216	
13:00:00 PM	216	213	267	
16:00:00 PM	314	298	244	284.8888889
19:00:00 PM	312	281	316	
22:00:00 PM	240	261	298	
1:00:00 AM	117	126	115	137.8333333
4:00:00 AM	162	176	131	



Gambar 13. Grafik Pengujian Koneksi Perangkat Jaringan



Gambar 14. Pengujian Keseluruhan Sistem Terhadap Aplikasi Android

Berdasarkan tabel hasil pengujian di atas seluruh fungsi aplikasi yang telah dibuat telah berfungsi sesuai dengan perancangan. Dimana fitur monitoring dapat merubah respon code dari get all input Web Server menjadi sebuah label monitoring, notifikasi, serta alarm berdasarkan kondisi dari limit switch yang terpasang. Terdapat perbedaan waktu respon maksimal 3 detik dari perubahan data pada Web Server, hal ini dikarenakan set timer refresh dari aplikasi untuk pengambilan data dari Web Server Teleduino adalah 2 detik. Dari 9 kondisi pengujian pada seluruh lokasi pada tabel di atas didapatkan rata-rata waktu respon adalah 2506.777778 ms atau 2.5 detik, dan kecepatan akses dari aplikasi ke Web Server adalah 355.96 ms. Berdasarkan rata-rata waktu respon dan kecepatan akses dari aplikasi ke Web Server maka didapatkan data bahwa alat ini mampu bekerja dengan delay 1000ms atau 1 detik, pada kecepatan akses aplikasi ke Web Server 142 ms.

Tabel 8. Hasil Pengujian Keseluruhan Sistem Terhadap Aplikasi Android

Lokasi	Kondisi	Fungsi Aplikasi				Waktu Respon (ms)	Kecepatan Akses Ke Web Server (ms)
		Monitoring	Notifikasi	Alarm	Reset		
Lokasi A	Low	Berfungsi	Berfungsi	-	-	2680	380.56
	Med	Berfungsi	Berfungsi	-	-	2241	318.22
	High	Berfungsi	Berfungsi	Berfungsi	Berfungsi	2183	309.99
Lokasi B	Low	Berfungsi	Berfungsi	-	-	2809	398.88
	Med	Berfungsi	Berfungsi	-	-	2291	325.32
	High	Berfungsi	Berfungsi	Berfungsi	Berfungsi	2504	355.57
Lokasi C	Low	Berfungsi	Berfungsi	-	-	2731	387.80
	Med	Berfungsi	Berfungsi	-	-	2935	416.77
	High	Berfungsi	Berfungsi	Berfungsi	Berfungsi	2187	310.55

5. Kesimpulan

Aplikasi Sistem Telemetry Banjir yang dirancang mampu memonitoring ketinggian pergerakan air pada drainase, dimana pergerakan tersebut dibagi atas 3 kondisi, yaitu : *Low*, *Medium*, dan *High*, yang akan memvisualisasikan kondisi tersebut melalui notifikasi langsung ke aplikasi android. Dan dapat mengetahui ketinggian air ketika musim penghujan tiba pada lingkungan sekitar tempat tinggalnya dari jarak jauh. Alat ini memberikan notifikasi dengan jeda waktu sekitar $\pm 1000\text{ms}$ atau 1 detik pada kecepatan akses internet Smartphone ke webserver teleduino atau sebaliknya $\pm 142\text{ms}$.

Referensi

- Adjie, M. (2015). Pengertian Networking Lengkap.
- Chania, M. F., Sara, O., & Sadalia, I. (2022). Analisis Risk dan Return Investasi pada Ethereum dan Saham LQ45. *Studi Ilmu Manajemen dan Organisasi*, 2(2), 139-150. doi:[10.35912/simo.v2i2.669](https://doi.org/10.35912/simo.v2i2.669)
- Chika, O. V., Oshiohwemoh, D., & Promise, E. (2022). Impact of Tax Reforms on Economic Growth of Nigeria (2000-2021). *Goodwood Akuntansi dan Auditing Reviu*, 1(1), 79-95. doi:[10.35912/gaar.v1i1.1506](https://doi.org/10.35912/gaar.v1i1.1506)
- Dalena, M. N. R., Ali, S., & Ediwarman, E. (2022). Pengaruh Lingkungan Kerja dan Beban Kerja terhadap Kepuasan Kerja Karyawan Wisma Sehati BSD Tangerang Selatan. *Studi Ilmu Manajemen dan Organisasi*, 1(2), 115-136. doi:[10.35912/simo.v1i2.712](https://doi.org/10.35912/simo.v1i2.712)
- Darmawan, J., & Roba'in, H. (2022). Pengaruh Keberagaman Gender dan Pengungkapan Risiko Terhadap Efisiensi Investasi. *Goodwood Akuntansi dan Auditing Reviu*, 1(1), 29-47. doi:[10.35912/gaar.v1i1.1484](https://doi.org/10.35912/gaar.v1i1.1484)
- Dharma, F., Agustina, Y., & Windah, A. (2021). Penerapan Manajemen Keuangan dan Digital Marketing pada Pelaku Usaha Rengginang di Desa Kunjir. doi:<https://doi.org/10.35912/jnm.v1i1.620>
- Hernawan, B. D., & Srimulyani, V. A. (2022). Dampak Kompensasi terhadap Retensi Karyawan dengan Kepuasan Kerja sebagai Pemediiasi. *Reviu Akuntansi, Manajemen, dan Bisnis*, 1(2), 115-125. doi:[10.35912/rambis.v1i2.888](https://doi.org/10.35912/rambis.v1i2.888)
- Hesti, E., & Adewasti, A. (2018). APLIKASI ANDROID SEBAGAI PENGONTROL JARAK JAUH SMARTHOME DENGAN KONEKSI JARINGAN INTERNET. *Jurnal Surya Energy*, 2(2), 157-165.
- Mardiono, D. A., Nanra, S., & Rican, D. (2023). Rancang Bangun Pengaman Pintu Menggunakan RFID Dengan Mikrokontroler Atmega 328. doi:<https://doi.org/10.35912/jatra.v1i1.1872>
- Masbullah, M., & Bahri, S. Y. (2021). Manajemen pendataan aset sekretariat daerah kabupaten Lombok Timur. *Studi Ilmu Manajemen dan Organisasi*, 2(1), 1-9. doi:[10.35912/simo.v2i1.289](https://doi.org/10.35912/simo.v2i1.289)
- Melvi, M., Nurhayati, N., Batubara, M. A. M., Septama, H. D., & Ulvan, A. (2023). Unjuk Kerja Teknologi Akses Jamak TD-CDMA dan TD-SCDMA pada Infrastruktur Jaringan High Altitude Platform Stations. *Jurnal Teknologi Riset Terapan*, 1(1), 51-59. doi:[10.35912/jatra.v1i1.1790](https://doi.org/10.35912/jatra.v1i1.1790)
- Melvi, M., Ulvan, A., Sidiq, M. R., & Batubara, M. A. M. (2023). Rancang Bangun Sistem Monitoring Ketinggian Muka Air Laut Menggunakan Arduino Pro Mini dan NodeMCU ESP8266. doi:<https://doi.org/10.35912/jatra.v1i1.1794>
- Novianti, A., Akhyari, M. R., Hakim, S. N., & Saputra, N. (2021). Meneropong Dampak Stress, Social Skills dan Internet Addiction pada Office Workers semasa Covid-19. *Reviu Akuntansi, Manajemen, dan Bisnis*, 1(1), 39-60. doi:[10.35912/rambis.v1i1.501](https://doi.org/10.35912/rambis.v1i1.501)
- Nurhidayati, R., Arifiya, N., Setiawan, A., Larasakti, R., & Heriansyah, H. (2022). Strategi Pemasaran dengan Teori Permainan pada Usaha Minuman Street Boba dan Kamsia Boba. *Studi Ilmu Manajemen dan Organisasi*, 3(2), 303-314. doi:[10.35912/simo.v3i2.1388](https://doi.org/10.35912/simo.v3i2.1388)
- Oktaviani, E., & Sembiring, F. M. (2022). Pengaruh Likuiditas, Leverage, Profitabilitas dan Ukuran Perusahaan terhadap Financial Distress: Studi Pada Perusahaan Sub Sektor Wholesale di BEI Periode. *Reviu Akuntansi, Manajemen, dan Bisnis*, 1(2), 127-144. doi:[10.35912/rambis.v1i2.944](https://doi.org/10.35912/rambis.v1i2.944)

- Rachmawati, A., Nugraha, A. L., & Awaluddin, M. (2017). Desain Aplikasi Mobile Informasi Pemetaan Jalur Batik Solo Trans Berbasis Android Menggunakan Location Based Service. *Jurnal Geodesi Undip*, 6(2), 46-55.
- Riesna, D. M. R., Pujiyanto, D. E., Efendi, A. J. I., Nugroho, B. A., & Saputra, D. I. S. (2023). Identifikasi Platform dan Faktor Sukses dalam Manajemen Proyek Teknologi Informasi. *Jurnal Teknologi Riset Terapan*, 1(1), 1-9. doi:[10.35912/jatra.v1i1.1458](https://doi.org/10.35912/jatra.v1i1.1458)
- Riski, S., & Widian, M. E. (2020). Pengaruh iklim komunikasi dan promosi jabatan terhadap kinerja dan motivasi dengan disiplin dan kepuasan kerja sebagai variabel intervening. *Studi Ilmu Manajemen dan Organisasi*, 1(1), 33-55. doi:[10.35912/simo.v1i1.61](https://doi.org/10.35912/simo.v1i1.61)
- Salim, A., Jatnika, R., & Yudiana, W. (2023). Hubungan Fear of Missing out dengan Social Media Addiction pada Pengguna Instagram. doi:<https://doi.org/10.35912/jisted.v2i1.2372>
- Sandi, A. (2017). Mengenal Apa itu Web API.
- Suyudi, I., Sudadio, S., & Suherman, S. (2022). Pengenalan Bahasa Isyarat Indonesia menggunakan Mediapipe dengan Model Random Forest dan Multinomial Logistic Regression. doi:<https://doi.org/10.35912/jisted.v1i1.1899>
- Utomo, K. B., Azizah, A., & Pangestu, M. A. (2022). Peran Computer Assited Test dalam Implementasi Penilaian di SD Negeri 005 Palaran. doi:<https://doi.org/10.35912/jisted.v1i1.1529>
- Wihidayat, E. S. (2017). Pengembangan Aplikasi Android Menggunakan Integrated Development Environment (Ide) App Inventor-2. *Jurnal Ilmiah Edutic: Pendidikan dan Informatika*, 4(1), 1-12.
- Yando, J. R., Panusunan, P., & Fauzan, F. (2023). Penggunaan Filler Tanah (Silt) sebagai Perencanaan Campuran Aspal Beton AC-WC. doi:<https://doi.org/10.35912/jatra.v1i1.1873>
- Yuniarahmah, & <https://dwiyunia12.wordpress.com/>. (2018). Winbox Pengertian, fungsi Dan Kegunaannya.