

DAFTAR PUSTAKA

- Ahmed, M.A.M. & Wei, W.G. Study on Automated Car Parking System Based on Microcontroller. *Int. J. Eng. Res. Technol. (IJERT)* 2014, 3, 256–258.
- Amri, M. R., dkk. (2016). *Risiko Bencana Indonesia*. Jakarta: Direktort Pengurangan Resiko Bencana Deputi Bidang Pencegahan dan Kesiapsiagaan.
- Andrianto, H., & Darmawan, A. *Arduino Belajar Cepat dan Pemrograman*. Bandung: Informatika.
- Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks*, 54(15), 2787-2805.
- Awtrey, D. (1999). A 1-Wire Rain Gauge. *The Journal of Applied Sensing Technology*, 16(12), 34-40.
- BNPB. (2012). *Pedoman Sistem Peringatan Dini Berbasis Masyarakat*. Jakarta: Badan Nasional Penanggulangan Bencana.
- BNPB. (2021). *Data Jumlah Kejadian dan Dampak Bencana Provinsi Sulawesi Selatan Tahun 2020*. (Online), (<https://bpbdsulselprov.go.id/>, Diakses pada tanggal 29 November 2022).
- BNPB. (2022). *Data & Informasi Bencana Indonesia*. (Online), (<http://dibi.bnpb.go.id/>, Diakses pada tanggal 29 November 2022).
- Boxall, J. (2013). *Arduino Worskhop - A hands-on introduction with 65 Projects*. In No Starch Press.
- Bruch, J. & Grudniski, G. (2014). *Analisa dan Sistem Informasi*. Yogyakarta: Deepublish.
- Cakara, A. C., Hanny, H., Desi, P. K., & Setia, A. (2015). Logika Fuzzy Menggunakan Metode Tsukamoto Untuk Prediksi Perilaku Konsumen di Toko Bangunan. *Jurnal Techno.Com*, 14(4), 255-2265.
- Cañiza, D. dan Mitjans, F. (2018). Remote Monitoring System of Automatic Rain Gauges using Machine-to-Machine application. *Journal of Electronics and Communication Engineering*, 13(1), 4-13.
- Castillo, O. and Melin, P. (2008). *Type-2 Fuzzy Logic: Theory and Application*. Springer-Verlag Berlin Heidelberg.
- Chodhury, S. Singh, G. K., & Mehra, R. M. (2014). Design and Verification Serial Peripheral Interface (SPI) Protocol for Low Power Applications. *International Journal of Innovative Research in Science, Engineering and Technology*, 3(10), 16750-16758.
- Djuandi, F. (2011). *Pengenalan Arduino*. Jakarta: Penerbit Elexmedia.

- Enterprise, J. (2017). *Otodidak pemrograman JavaScript*. Jakarta: Elex Media Komputindo.
- Fankhauser, R. (1997). Measurement Properties of Tipping Bucket Rain Gauges and Their Influence on Urban Runoff Simulation. *Water Science & Technology*, 36(8-9), 7-12.
- Fankhauser, R. (1998). Influence of Systematic Errors from Tipping Bucket Rain Gauges on Recorded Rainfall Data. *Water Science & Technology*, 37(11), 121-129.
- Fortino, G., Savaglio, C., Palau, C.E., de Puga, J.S., Ganzha, M., Paprzycki, M., Montesinos, M., Liotta, A. and Llop, M. (2018). Towards multi-layer interoperability of heterogeneous IoT platforms: The INTER-IoT approach. *In Integration, interconnection, and interoperability of IoT systems* (pp. 199-232).
- Fraden, J. (2016). *Handbook of Modern Sensors Fifth Edition*. San Diego, CA, USA: Springer International Publishing.
- Geetanjali, V., Subramanian, I., Kannan, G., Prathiba, S. B., & Raja, G. (2019). IoTexpert: Interconnection, interoperability and integration of IoT platforms. *Proceedings of the 11th International Conference on Advanced Computing (ICoAC)*, 212-219. <https://doi.org/10.1109/ICoAC48765.2019.246842>
- Handoko, P. (2017). Sistem kendali monolitik perangkat elektronika. *Program Studi Teknik Informatika, Fakultas Teknologi dan Desain, Universitas Pembangunan Jaya*.
- Hardjono, D. (2006). *Menguasai Pemrograman Web dengan PHP 5*. Semarang: Wahana Komputer.
- Hidayatullah, P., & Kawistara, J. K. (2015). *Pemrograman web*. Bandung: Informatika.
- Hutahaean, J. (2015). *Konsep Sistem Informasi*. Yogyakarta: Deepublish CV. Budi Utama.
- Indrajani. (2011). *Bedah Kilat 1 Jam Pengantar dan Sistem Basis Data*. Jakarta: PT. Elex Media Komputindo.
- Iqbal, M. A., Olaleye, O. G., & Bayoumi, M. A. (2016). A Review on Internet of Things (Iot): Security and Privacy Requirements and the Solution Approaches. *Global Journal of Computer Science and Technology*, 16(7), 9-10.
- Ismail, A. N., Sumarsono, F., Nuryana, & Kurniawan, T. (2019). Perancangan Website Data Karyawan Dengan Menggunakan PHP dan MySQL. *Jurnal Sistem Basis Data*.
- ITU Internet Reports. (2005). *The Internet of Things*.

- Kadli, P., Reddy, R., Byali, R., Reddy, H. B. S., Divya, P. B., Maskikar, S. V, Ne, C., & Bhonsle, S. H. (2022). IoT Based Early Flood Detection Using Machine Learning. *International Journal of Innovative Research In Technology*, 9(1), 350-355.
- Kastina, M., & Silalahi, M. (2016). Logika Fuzzy Metode Mamdani Dalam Sistem Keputusan Fuzzy Produksi Menggunakan Matlab. *Jurnal Ilmu Komputer*, 1(2), 171–181.
- Kaur, A., Naveen, K. N., Kishore, P. V., Chandrakanth, M., Reddy, M. M., & Muralidhar, M. V. (2021). Secure Home Using Magnetic Reed Switch. *International Journal of Computer and Organization Trends*, 11(2), 21–24. <https://doi.org/10.14445/22492593/ijcot-v11i2p307>
- Kelemen, M., Virgala, I., Kelemenová, T., Miková, Ľ., Frankovský, P., Lipták, T., & Lörinc, M. (2015). Distance Measurement via Using of Ultrasonic Sensor. *Journal of Automation and Control*, 3(3), 71–74. <https://doi.org/10.12691/automation-3-3-6>
- Kf, L., Jm, J., Sc, W., & Ch, C. (2019). Tipping Bucket Rain Gauge Performance Analysis under Heavy Rainfall. *Advancements in Civil Engineering & Technology*, 3(3), 319–322. <http://dx.doi.org/10.31031/ACET.2019.03.000564>
- Kustiyahningsih, Y., dan Anamisa, D. R. (2011). *Pemrograman Basis Data Berbasis Web Menggunakan PHP Dan Mysql*. Yogyakarta: Graha Ilmu
- Kusumadewi, S., & Purnomo, H. 2010. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu.
- Lasse, K. (2020). State of the IoT 2020: 12 billion IoT Connections, Surpassing Non-IoT for The First Time (Online). (<https://iot-analytics.com/state-of-theiot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/>, diakses 26 Januari 2022).
- Lattice Semiconductor Corporation, “Micro SD Card Breakout Board Tutorial Datasheet”, 2019.<https://learn.adafruit.com/adafruit-micro-sd-breakout-board-card-tutorial>.
- Lay, M. E. (2017). E-Commerce Gitar Akustik dan Sparepart Kota Malang Menggunakan Metode Customer to Customer. *Jurnal Mahasiswa Teknik Informatika*, 1(2), 1-7.
- Louis, L. (2016). Working Principle of Arduino and Using it as a Tool for Study and Research. *International Journal of Control, Automation, Communication and Systems*, 1(2), 21-29. <https://doi.org/10.5121/ijcacs.2016.1203>
- Madakam, S., Lake, V., Lake, V. and Lake, V. (2015). Internet of Things (IoT): A literature review. *Journal of Computer and Communications*, 3(3), 164-173.
- Maniah & Hamidin, D. (2017). *Analisis Dan Perancangan Sistem Informasi*

Pembahasan Secara Praktis Dengan Contoh Kasus, 1st ed. Yogyakarta: Deepublish.

- Manurung, M. J., Poningsih, Andani, S. R., Safii, M., & Irawan. (2021). Door Security Design Using Fingerprint and Buzzer Alarm Based on Arduino. *Journal of Computer Networks , Architecture and High Performance Computing*, 3(1), 42–51.
- Mariko, S. (2019). Aplikasi Website Berbasis HTML dan JavaScript untuk Menyelesaikan Fungsi Integral pada Mata Kuliah Kalkulus. *Jurnal Inovasi Teknologi Pendidikan*, 6(1), 80-91.
- Maryanto, B. (2009). Memanfaatkan Cascading Style Sheet untuk Memperindah Tampilan Web. *Media Informatika*, 8(2), 82-89.
- Mnati, M. J. et al. (2020). An Open-source Non-Contact Thermometer Using Low-Cost Electronic Components. *HardwareX*, 9(629), 1-13.
- Mousavi, F. S., Yousefi, S., Abghari, H., & Ghasemzadeh, A. (2021). "Design of an IoT-based Flood Early Detection System using Machine Learning," *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*, 1-7, doi: 10.1109/CSICC52343.2021.9420594.
- Mulyadi. (2016). *Sistem Akuntansi*. Jakarta: Salemba Empat.
- Parihar, Y. S. (2019). Internet of Things and Nodemcu A review of use of Nodemcu ESP8266 in IoT products. *Journal of Emerging Technologies and Innovative Research*, 6(6), 1085-1088.
- Perera, E. D. P., & Lahat, L. (2015). Fuzzy logic based flood forecasting model for the Kelantan River basin, Malaysia. *Journal of Hydro-Environment Research*, 9(4), 542-553. <https://doi.org/10.1016/j.jher.2014.12.001>
- Perry, W. E. (1995). *Effective Methods for Software Testing*. Indianapolis, Indiana: Wiley Publishing.
- Rabbani, I. M., Kusuma, P. D., dan Saputra, R. E. (2018). Pengembangan Alat Deteksi Banjir Berbasis Metode Fuzzy. *e-Proceeding of Engineering*, 5(3), 6283-6290.
- Ray, P. P. (2018). A survey on Internet of Things architectures. *Journal of King Saud University – Computer and Information Sciences*, 30(3), 291–319. <https://doi.org/10.1016/j.jksuci.2016.10.003>
- Ross, T. J. (2004). *Fuzzy Logic with Engineering Applications, Second Edition*. University of New Mexico, USA: John Wiley & Sons, Ltd.
- Ross, T. J. (2010). *Fuzzy Logic with Engineering Applications, Third Edition*. University of New Mexico, USA: John Wiley & Sons, Ltd.
- Samikwa, E., Voigt, T., & Eriksson, J. (2020). "Flood Prediction Using IoT and Artificial Neural Networks with Edge Computing," *2020 International*

- Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCoM) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics)*, 234-240. doi: 10.1109/iThings-GreenCom-CPSCoM-SmartData-Cybermatics50389.2020.00053.
- Saptaji, H. W. (2015). *Mudah belajar Mikrokontroller dengan Arduino*. Bandung: Widya Media.
- Saputra, E. W. (2020). Optimasi Fungsi Keanggotaan Fuzzy Mamdani Menggunakan Algoritma Genetika Untuk Penentuan Penerima Beasiswa. *Jurnal Sarjana Teknik Informatika*, 8(2), 34–50.
- Sari, A. O., & Abdilah, A. Sunarti. (2015). *Web Programming*. Yogyakarta: Graha Ilmu.
- Scott, G. M. (2004). *Prinsip-prinsip Sistem Informasi Manajemen*. Budiman, A. N. Jakarta: Raja Grafindo Persada.
- Sinha, A., Kumar, P., Rana, N. P., Islam, R., & Dwivedi, Y. K. (2019). Impact of internet of things (IoT) in disaster management: a task-technology fit perspective. *Annals of Operations Research*, 283(1–2), 759–794. <https://doi.org/10.1007/s10479-017-2658-1>
- Siswanto, H, T. H., & Junaedi, M. (2020). Prototype Smart Home dengan Konsep Iot (*Internet of Thing*) Berbasis Nodemcu Dan Telegram. *Jurnal SIMIKA*, 3(1), 85-93.
- Soleman, R., Mirza, M., & Sofwan, A. (2019). Rancang Bangun Prototype Sensor Cerdas Parkir Mobil Sebagai Sensor Mundur. *TEKINFO: Jurnal Penelitian Teknik dan Informatika*, 1(2), 119–127.
- Solichin, A. (2016). *Pemrograman Web dengan PHP dan MySQL*. Google Buku. Universitas Budi Luhur.
- Soni, A., & Aman, A. (2018). Distance Measurement of an Object by using Ultrasonic Sensors with Arduino and GSM Module. *IJSTE-International Journal of Science Technology & Engineering* |, 4(11), 23–28. www.ijste.org
- Sosrodarsono, S. (1976). *Hidrologi untuk Pengairan*. Jakarta: PT. Pradya Paramita
- Srivastava, L. (2006). Pervasive, ambient, ubiquitous: the magic of radio. *Proceedings of European Commission Conference “From RFID to the Internet of Things”*, Bruxelles, Belgium.
- Sterling, B. (2005). Shaping Things – Mediawork Pamphlets. *The MIT Press*
- Subagyo, L. A., & Suprianto, B. (2017). Sistem Monitoring Arus Tidak Seimbang 3 Fasa Berbasis Arduino Uno. *Jurnal Teknik Elektro*, 6(3), 213-221.
- Sunarfrihantono, B. (2002). *PHP dan MySQL untuk Web*. Yogyakarta: Andi

- Suryantoro, H. & Budiyo, A. (2019). Prototype Sistem Monitoring Level Air Berbasis LabView & Arduino Sebagai Sarana Pendukung Praktikum Instrumentasi Sistem Kendali. *Indonesian Journal of Laboratory*, 1(3), 20-32.
- Suryanto, M. & Rijanto, T. (2019). Rancang Bangun Alat Pencatat Biaya Pemakaian Energi Listrik pada Kamar Kos menggunakan Modul Global System for Mobile Communications (GSM) 800L Berbasis Arduino Uno. *Jurnal Teknik Elektro*, 1(8), 47-55.
- Tun, A. T., Htwe, T. Z., & Aung, C. S. (2020). Automatic Water Refilling System. *Iconic Research and Engineering Journals*, 4(2), 56–60.
- Werner, M G F., Schellekens, J and Kwadijk, JCJ. (2005). Flood Early Warning Systems for Hydrological (sub) Catchments. *Encyclopedia of Hydrological Sciences*.
- Wicaksono, W. A., & Silalahi, L. M. (2020). Rancang Bangun Alat Pendeteksi Banjir Menggunakan Arduino Dengan Metode Fuzzy Logic. *Jurnal Teknologi Elektro*, 11(2), 93-99. <https://doi.org/10.22441/jte.2020.v11i2.005>
- Widaningsih, S. (2017). Analisis Perbandingan Metode Fuzzy Tsukamoto, Mamdani dan Sugeno dalam Pengambilan Keputusan Penentuan Jumlah Distribusi Raskin di Bulog Sub. Divisi Regional (Divre) Cianjur. *Jurnal Informatika dan Manajemen STMIK*, 11(1), 51-65.
- Wilson, J. S. (2005). *Sensor Technology Handbook*. Newnes, Elsevier, USA.
- Yudihartanti, Y. (2011). Analisis Komparasi Metode Fuzzy Mamdani dan Sugeno dalam Penjadwalan Mata Kuliah. *Progresif*, 7(2), 731-780.
- Yuliantika, S., & Kartika, D. L. (2022). Implementasi Metode Fuzzy Mamdani sebagai Deteksi Awal Banjir Lokal di Bendung Gerak Serayu. *Journal of Mathematics and Mathematics Education*, 4(1), 17-25. <https://dx.doi.org/10.21580/square.2020.2.1.10783>
- Yuwono, T., Ruzardi, Nordin, R., Mohamed, R., & Ismail, M. (2020). The Model of Fuzzy Logic in IoT System as Decision Support System for Determining Flood Disaster Status. *Journal of Physics: Conference Series*, 1529 (5).

Lampiran 1. Hasil pengukuran pulsa echo terhadap jarak

Jarak (cm)	Pulsa (μS)			Rata-rata
	Uji 1	Uji 2	Uji 3	
1	161	161	161	161
2	180	161	161	167,33
3	161	161	161	161
4	161	161	161	161
5	237	237	237	237
6	293	293	293	293
7	369	363	369	367
8	426	426	426	426
9	483	483	483	483
10	536	536	536	536
11	618	618	618	618
12	680	680	680	680
13	680	680	680	680
14	761	761	762	761,33
15	843	843	844	843,33
16	905	905	905	905
17	963	963	963	963
18	1019	1019	1019	1019
19	1075	1075	1075	1075
20	1131	1131	1131	1131
21	1187	1187	1187	1187
22	1243	1243	1244	1243,33
23	1300	1300	1300	1300
24	1352	1335	1352	1346,33
25	1386	1386	1386	1386
26	1420	1420	1420	1420
27	1505	1505	1505	1505
28	1550	1550	1550	1550
29	1590	1584	1590	1588
30	1668	1668	1668	1668
31	1719	1719	1719	1719
32	1780	1780	1780	1780
33	1811	1811	1811	1811
34	1917	1917	1917	1917
35	1947	1947	1948	1947,33

36	2008	2008	2008	2008
37	2039	2039	2039	2039
38	2142	2142	2142	2142
39	2155	2155	2155	2155
40	2209	2209	2209	2209
41	2290	2290	2290	2290
42	2371	2371	2371	2371
43	2384	2384	2384	2384
44	2438	2438	2438	2438
45	2486	2486	2492	2488
46	2523	2523	2523	2523
47	2605	2605	2605	2605
48	2687	2687	2664	2679,33
49	2746	2746	2746	2746
50	2804	2804	2804	2804

Lampiran 2. Hasil pengukuran jarak *sensing* terhadap jarak referensi

Jarak (cm)	Jarak sensing (cm)			Rata-rata	Error Relatif (%)
	Uji 1	Uji 2	Uji 3		
1	2,9	2,79	2,88	2,86	185,67
2	2,9	2,9	2,9	2,90	45,00
3	2,9	2,9	2,9	2,90	3,33
4	2,9	2,9	2,9	2,90	27,50
5	4,27	4,16	4,14	4,19	16,20
6	5,27	5,17	5,18	5,21	13,22
7	6,64	6,53	6,66	6,61	5,57
8	7,66	7,68	7,56	7,63	4,58
9	8,69	8,58	8,69	8,65	3,85
10	10,02	10,02	10,13	10,06	0,57
11	11,12	11,13	11,22	11,16	1,42
12	12,23	12,21	12,23	12,22	1,86
13	12,34	13,32	13,34	13,00	0,00
14	13,81	13,79	13,7	13,77	1,67
15	15,16	14,8	15,18	15,05	0,31
16	16,27	16,27	16,27	16,27	1,69
17	17,32	17,3	17,3	17,31	1,80
18	18,32	18,32	18,32	18,32	1,78
19	19,33	19,33	19,33	19,33	1,74
20	20,34	20,34	20,23	20,30	1,52
21	21,34	21,32	21,36	21,34	1,62
22	22,35	22,24	22,37	22,32	1,45
23	23,37	23,25	23,27	23,30	1,29
24	24,31	24,2	24,31	24,27	1,14
25	24,92	24,92	24,81	24,88	0,47
26	25,84	25,73	26,34	25,97	0,12
27	27,26	27,24	27,36	27,29	1,06
28	27,97	27,96	27,87	27,93	0,24
29	29,39	28,59	29,5	29,16	0,55
30	30,01	29,99	30,01	30,00	0,01
31	30,9	31,01	31,01	30,97	0,09
32	32	32	31,98	31,99	0,02
33	33,1	32,67	33,37	33,05	0,14
34	33,64	33,92	34,46	34,01	0,02
35	35	35	35,02	35,01	0,02

36	36,12	36,1	36,12	36,11	0,31
37	37,46	37,48	37,48	37,47	1,28
38	38	38,02	37,91	37,98	0,06
39	38,99	38,97	38,74	38,90	0,26
40	40,2	39,71	39,71	39,87	0,32
41	41,17	41,06	41,17	41,13	0,33
42	42,14	42,14	42,03	42,10	0,25
43	43,11	43	43,09	43,07	0,16
44	44,08	43,83	43,95	43,95	0,11
45	44,8	45,18	44,71	44,90	0,23
46	45,79	45,98	46,09	45,95	0,10
47	46,94	46,83	46,92	46,90	0,22
48	48	48,3	47,87	48,06	0,12
49	49,36	48,93	49,04	49,11	0,22
50	50,4	50,4	50,51	50,44	0,87

Lampiran 3. Pengukuran tinggi air oleh alat dengan tinggi air referensi

Tinggi Referensi (cm)	Tinggi air hasil pengukuran alat (cm)			Rata-rata
	Uji 1	Uji 2	Uji 3	
2	2,11	2,74	2,72	2,52
3	3,26	3,78	3,15	3,40
4	4,52	4,72	4,52	4,59
5	5,01	5,13	5,22	5,12
6	5,81	6,68	6,19	6,23
7	7,14	7,16	7,16	7,15
8	8,11	8,22	7,88	8,07
9	9,34	9,34	9,34	9,34
10	10,31	10,31	10,31	10,31
11	11,76	11,76	11,28	11,60
12	12,23	12,23	12,25	12,24
13	13,33	13,34	13,33	13,33
14	14,42	14,42	13,88	14,24
15	14,98	15,25	14,98	15,07
16	16,35	16,35	16,33	16,34
17	17,33	17,44	17,44	17,40
18	17,98	18	17,98	17,99
19	19,08	19,24	19,08	19,13
20	19,85	19,98	19,96	19,93
21	20,88	20,88	21,38	21,05
22	22,1	21,99	21,99	22,03
23	23,01	23,01	23,01	23,01
24	23,82	24,29	24,4	24,17
25	25,01	24,9	24,9	24,94
26	25,32	25,93	25,82	25,69
27	26,54	26,77	26,77	26,69
28	28,1	28,12	28,1	28,11
29	29,13	29,11	29,11	29,12
30	30,13	30,24	30,13	30,17
31	31,14	31,14	31,14	31,14
32	32,14	32,14	32,14	32,14
33	33,15	33,49	33,49	33,38
34	34,59	33,87	34,59	34,35
35	34,86	34,97	34,61	34,81
36	36	35,96	35,92	35,96

Lampiran 4. Hasil pengujian Data *Logger*

No	Tanggal	Jam	Tinggi Air	Curah Hujan	Prediksi	Kesimpulan
1	20/05/2023	14:53:00	0.00	0.00	Potensi Rendah	Sesuai
2	20/05/2023	14:54:00	0.00	5.00	Potensi Rendah	Sesuai
3	20/05/2023	14:55:00	31.68	13.60	Potensi Tinggi	Sesuai
4	20/05/2023	14:56:00	24.49	0.00	Potensi Rendah	Sesuai
5	20/05/2023	14:57:00	24.60	4.80	Potensi Sedang	Sesuai
6	20/05/2023	14:58:00	26.65	0.00	Potensi Sedang	Sesuai
7	20/05/2023	14:59:00	31.35	1.60	Potensi Sedang	Sesuai
8	20/05/2023	15:00:00	26.65	0.00	Potensi Sedang	Sesuai
9	20/05/2023	15:01:00	33.74	0.00	Potensi Sedang	Sesuai
10	20/05/2023	15:02:00	26.41	0.80	Potensi Sedang	Sesuai
11	20/05/2023	15:03:00	26.30	0.00	Potensi Sedang	Sesuai
12	20/05/2023	15:04:00	26.65	0.40	Potensi Sedang	Sesuai
13	20/05/2023	15:05:00	1.09	0.20	Potensi Rendah	Sesuai
14	20/05/2023	15:06:00	17.46	7.40	Potensi Sedang	Sesuai
15	20/05/2023	15:07:00	17.46	0.00	Potensi Rendah	Sesuai
16	20/05/2023	15:08:00	0.67	0.00	Potensi Rendah	Sesuai
17	20/05/2023	15:09:00	27.76	0.60	Potensi Sedang	Sesuai
18	20/05/2023	15:10:00	27.76	0.00	Potensi Sedang	Sesuai
19	20/05/2023	15:11:00	27.76	0.20	Potensi Sedang	Sesuai
20	20/05/2023	15:12:00	1.09	1.60	Potensi Rendah	Sesuai
21	20/05/2023	15:13:00	29.45	5.00	Potensi Sedang	Sesuai
22	20/05/2023	15:14:00	29.34	0.00	Potensi Sedang	Sesuai
23	20/05/2023	15:15:00	29.34	9.60	Potensi Tinggi	Sesuai
24	20/05/2023	15:16:00	22.66	8.40	Potensi Sedang	Sesuai
25	20/05/2023	15:17:00	22.66	0.20	Potensi Rendah	Sesuai
26	20/05/2023	15:18:00	27.31	0.60	Potensi Sedang	Sesuai
27	20/05/2023	15:19:00	27.44	1.20	Potensi Sedang	Sesuai
28	20/05/2023	15:20:00	27.42	0.20	Potensi Sedang	Sesuai
29	20/05/2023	15:21:00	35.94	0.40	Potensi Sedang	Sesuai
30	20/05/2023	15:22:00	35.96	0.00	Potensi Sedang	Sesuai

Lampiran 5. Hasil pengujian RTC DS1302

No	Waktu sebenarnya	Waktu RTC	Selang waktu	Jumlah detik RTC/menit
1	02:21:00.172	02:21:00	-0,074	59,926
2	02:22:00.098	02:22:00	-0,105	59,895
3	02:22:59.993	02:23:00	-0,047	59,953
4	02:23:59.946	02:24:00	-0,079	59,921
5	02:24:59.867	02:25:00	-0,068	59,932
6	02:25:59.799	02:26:00	-0,078	59,922
7	02:26:59.721	02:27:00	-0,083	59,917
8	02:27:59.638	02:28:00	-0,074	59,926
9	02:28:59.564	02:29:00	-0,068	59,932
10	02:29:59.496	02:30:00	-0,078	59,922
11	02:30:59.418	02:31:00	-0,068	59,932
12	02:31:59.350	02:32:00	-0,105	59,895
13	02:32:59.245	02:33:00	-0,060	59,940
14	02:33:59.185	02:34:00	-0,102	59,898
15	02:34:59.083	02:35:00	-0,045	59,955
16	02:35:59.038	02:36:00	-0,102	59,898
17	02:36:58.936	02:37:00	-0,045	59,955
18	02:37:58.891	02:38:00	-0,076	59,924
19	02:38:58.815	02:39:00	-0,086	59,914
20	02:39:58.729	02:40:00	-0,072	59,928
21	02:40:58.657	02:41:00	-0,067	59,933
22	02:41:58.590	02:42:00	-0,083	59,917
23	02:42:58.507	02:43:00	-0,075	59,925
24	02:43:58.432	02:44:00	-0,094	59,906
25	02:44:58.338	02:45:00	-0,056	59,944
26	02:45:58.282	02:46:00	-0,108	59,892
27	02:46:58.174	02:47:00	-0,047	59,953
28	02:47:58.127	02:48:00	-0,079	59,921
29	02:48:58.048	02:49:00	-0,060	59,940
30	02:49:57.988	02:50:00	-0,072	59,928

Lampiran 6. Hasil pengujian *database*

No	Waktu pengiriman	Waktu penerimaan	<i>delay</i> (detik)
1	22:33:59.715	22:34:00.434	0,719
2	22:34:59.638	22:35:00.624	0,986
3	22:35:59.537	22:36:00.224	0,687
4	22:36:59.480	22:37:00.433	0,953
5	22:37:59.391	22:38:00.155	0,764
6	22:38:59.335	22:39:00.243	0,908
7	22:39:59.248	22:40:00.042	0,794
8	22:40:59.204	22:41:00.173	0,969
9	22:41:59.093	22:41:59.860	0,767
10	22:42:59.052	22:42:59.974	0,922
11	22:43:58.988	22:43:59.658	0,670
12	22:44:58.890	22:44:59.901	1,011
13	22:45:58.817	22:45:59.580	0,763
14	22:46:58.748	22:46:59.075	0,327
15	22:47:58.666	22:47:59.075	0,409
16	22:48:58.589	22:48:59.483	0,894
17	22:49:58.540	22:49:59.321	0,781
18	22:50:58.474	22:50:59.412	0,938
19	22:51:58.405	22:51:59.093	0,688
20	22:52:58.328	22:52:58.999	0,671
21	22:53:58.251	22:53:59.014	0,763
22	22:54:58.150	22:54:59.134	0,984
23	22:55:58.065	22:55:58.823	0,758
24	22:56:58.02	22:56:58.722	0,702
25	22:57:57.942	22:57:58.768	0,826
26	22:58:57.872	22:58:58.638	0,766
27	22:59:57.759	22:59:58.548	0,789
28	23:00:57.679	23:00:58.364	0,685
29	23:01:57.643	23:01:58.360	0,717
30	23:02:57.536	23:02:58.270	0,734

Lampiran 7. Hasil pengujian *website*

No	Deskripsi Pengujian	Hasil yang diharapkan	Kesimpulan
1	Mengunjungi domain "gitgif.com" pada browser	Menampilkan halaman beranda <i>website</i>	Berhasil
2	Menekan elemen menu	Menampilkan 2 submenu yaitu submenu beranda dan submenu data	Berhasil
3	Menekan submenu beranda	Menampilkan halaman beranda	Berhasil
4	Menekan submenu data	Menampilkan halaman tabel data	Berhasil
5	Menambahkan data terbaru pada tabel database	Informasi ketinggian air, curah hujan, hasil prediksi, tanggal <i>update</i> akan berubah sesuai data terbaru pada database.	Berhasil
		Nilai data terakhir pada grafik <i>realtime</i> akan berubah sesuai data terbaru pada database.	Berhasil
		Data pada tabel data akan bertambah sesuai penambahan data pada database.	Berhasil
6	Prediksi potensi banjir rendah	<i>Background</i> pada potensi banjir akan berwarna hijau	Berhasil
7	Prediksi potensi banjir sedang	<i>Background</i> pada potensi banjir akan berwarna kuning	Berhasil
8	Prediksi potensi banjir tinggi	<i>Background</i> pada potensi banjir akan berwarna merah	Berhasil
9	Menambahkan 10 data pengukuran pada tabel database	Menampilkan 10 data terbaru pada grafik <i>real-time</i>	Berhasil
10	Mengarahkan kursor pada grafik <i>realtime</i>	Menampilkan informasi nilai sensor pada titik data yang terdekat dengan kursor	Berhasil
11	Menekan ikon <i>zoom in</i> dan <i>zoom out</i> pada elemen grafik	Memperbesar atau memperkecil grafik <i>realtime</i>	Berhasil

12	Menekan ikon <i>selection zoom</i> pada elemen grafik	Memperbesar grafik <i>realtime</i> sebesar ukuran yang dipilih	Berhasil
13	Menekan ikon <i>reset zoom</i> pada elemen grafik	Mengatur ulang <i>zoom</i>	Berhasil
14	Menekan ikon menu pada elemen grafik	Menampilkan submenu <i>download</i> grafik format SVG, PNG, dan CSV.	Berhasil
15	Menekan <i>download</i> SVG, <i>download</i> PNG, dan <i>download</i> CSV pada menu grafik	Grafik <i>realtime</i> akan terunduh otomatis dengan format SVG, PNG, atau CSV.	Berhasil
16	Menggeser dan <i>zoom in/zoom out</i> peta lokasi sensor	Peta dapat bergeser dan <i>zoom in/zoom out</i>	Berhasil
17	Menekan kotak <i>show entries</i> pada tabel data	Menampilkan pilihan 10, 25, 50, dan 100 untuk panjang tabel data yang ingin ditampilkan	Berhasil
18	Menekan angka 10, 25, 50, atau 100 pada kotak <i>show entries</i>	Menampilkan 10, 25, 50, atau 100 data terbaru	Berhasil
19	Menekan tombol <i>next</i> pada elemen <i>pagination</i>	Menampilkan halaman tabel data selanjutnya	Berhasil
20	Menekan tombol <i>previous</i> pada elemen <i>pagination</i>	Menampilkan halaman tabel data sebelumnya	Berhasil
21	Menekan angka yang ada pada elemen <i>pagination</i>	Menampilkan halaman tabel data sesuai angka yang dipilih	Berhasil
22	Menekan tombol <i>Copy</i> pada tabel data	Menyalin seluruh data ke clipboard	Berhasil
23	Menekan tombol <i>Print</i>	Menampilkan halaman <i>print</i> untuk tabel data	Berhasil
24	Menekan tombol CSV, Excel, atau PDF.	Mengunduh seluruh data dalam format CSV, Excel, atau PDF.	Berhasil
25	Melakukan pencarian pada kotak <i>search</i> tabel data	Menampilkan hasil pencarian pada tabel data	Berhasil

Lampiran 8. Script Arduino Uno

//Muhammad Yahya Departemen Geofisika Universitas Hasanuddin

```
//Import library
#include <virtuabotixRTC.h>
#include <LiquidCrystal_I2C.h>
#include <SD.h>
#include <SPI.h>

//Pin Modul RTC pada Arduino Uno
#define PIN_CLK 5
#define PIN_DAT 4
#define PIN_RST 2
int pinCS = 10;

//Inisialisasi
LiquidCrystal_I2C lcd(0x27, 16, 2);
virtuabotixRTC RTC(PIN_CLK, PIN_DAT, PIN_RST);
const float resolusi = 0.2; //Resolusi minimal curah hujan
float mmTotal = 0;
float chTotal = 0;
int sensor = 0;
int keadaanawal = 0;
int keadaanawaltravel = 0;
int tmp = 0;
float r=0;
float tinggiair;
String kirimdata = "";
float a;
File myFile;
int i = 0;

int pulsa = 9; //Pin sensor reed switch
int buzzerPin = 6; //Pin Buzzer pada Arduino Uno
//Pin sensor ultrasonik
int trigPin = 8;
int echoPin = 7;

void setup() {
  Serial.begin(9600);
  pinMode(pinCS, OUTPUT);
  lcd.init();
  lcd.backlight();
  lcd.clear();
```

```

pinMode(pulsa, INPUT);
keadaanawal = sensor;
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);

//cek ketersediaan file excel pada SD Card
if (!SD.begin(pinCS)){
  Serial.println("Gagal inisialisasi modul SD card.");
  return;
}
Serial.println("Modul SD card siap");
if (!SD.exists("data.csv")){
  Serial.println ("File data.csv tidak ditemukan, membuat file baru");
  myFile = SD.open("data.csv", FILE_WRITE); //Membuat file excel
  myFile.println("Tanggal, Jam, Tinggi Air, Curah Hujan, Prediksi");
  myFile.close();
} else{
  Serial.println("File data.csv ditemukan");
}
}

void loop() {
  RTC.updateTime();
  String jam, menit, detik, tanggal, bulan, tahun;
  jam = RTC.hours;
  menit = RTC.minutes;
  detik = RTC.seconds;
  tanggal = RTC.dayofmonth;
  bulan = RTC.month;
  tahun = RTC.year;
  int getLCD = RTC.seconds;
  // Menampilkan waktu pada serial monitor Arduino Uno
  // Serial.print("realtime :");
  // Serial.println(jam + ":" + menit + ":" + detik);
  if(RTC.seconds == 0){ //Artinya pengambilan data dan perhitungan prediksi
    setiap 1 menit (untuk mempermudah uji coba)
    float inputTMA = tinggiair;
    float inputCH = chTotal;
    float rendah = rendah_TMA (inputTMA);
    float sedang = sedang_TMA (inputTMA);
    float tinggi = tinggi_TMA (inputTMA);
    float gerimis = gerimis_CH (inputCH);
    float normal = normal_CH (inputCH);
    float lebat = lebat_CH (inputCH);
    //Rule
    // 1: Tinggi air RENDAH & Curah hujan GERIMIS -> Potensi KECIL
  }
}

```

```

float SR1 = min(rendah, gerimis);
// 2: Tinggi air RENDAH & Curah hujan NORMAL -> Potensi KECIL
float SR2 = min(rendah, normal);
// 3: Tinggi air RENDAH & Curah hujan LEBAT -> Potensi SEDANG
float SR3 = min(rendah, lebat);
// 4: Tinggi air SEDANG & Curah hujan GERIMIS -> Potensi KECIL
float SR4 = min(sedang, gerimis);
// 5: Tinggi air SEDANG & Curah hujan Normal -> Potensi SEDANG
float SR5 = min(sedang, normal);
// 6: Tinggi air SEDANG & Curah hujan LEBAT -> Potensi TINGGI
float SR6 = min(sedang, lebat);
// 7: Tinggi air TINGGI & Curah hujan GERIMIS -> Potensi SEDANG
float SR7 = min(tinggi, gerimis);
// 8: Tinggi air TINGGI & Curah hujan NORMAL -> Potensi TINGGI
float SR8 = min(tinggi, normal);
// 9: Tinggi air TINGGI & Curah hujan LEBAT -> Potensi TINGGI
float SR9 = min(tinggi, lebat);
// Serial.print("Fungsi keanggotaan hasil = ");
float potensi;
float sumA = 0;
float sumB = 0;
for (potensi = 0; potensi <= 100; potensi++) {
    float keanggotaan_hasil = hasil_potensi(SR1, SR2, SR3, SR4, SR5, SR6, SR7,
SR8, SR9, potensi);
    sumA += potensi * keanggotaan_hasil;
    sumB += keanggotaan_hasil;
    // Serial.print(potensi);
    // Serial.print("\t");
    // Serial.println(keanggotaan_hasil);
}
// Serial.print("Hasil Defuzzyfikasi = ");
float hasil_defuzzyfikasi = sumA/sumB;
// Serial.println(hasil_defuzzyfikasi);
String predict;
if (hasil_defuzzyfikasi < 33.33){
    predict = "Potensi Rendah";
} else if (hasil_defuzzyfikasi >= 33.33 && hasil_defuzzyfikasi <= 66.66){
    predict = "Potensi Sedang";
} else {
    predict = "Potensi Tinggi";
}

String predictshort;
if (hasil_defuzzyfikasi < 33.33){
    predictshort = "Rendah";
} else if (hasil_defuzzyfikasi >= 33.33 && hasil_defuzzyfikasi <= 66.66){

```

```

    predictshort = "Sedang";
} else {
    predictshort = "Tinggi";
}
// Serial.print("Time :");
// Serial.println(jam + ":" + menit + ":" + detik);

//Tampilkan data pada Serial Monitor Arduino
    kirimdata =
    ("#")+(String(tinggiair))+(",")+(String(mmTotal))+(",")+(String(hasil_defuzzyfik
asi));
    Serial.println(kirimdata);

//Baca data Serial Monitor Arduino kemudian disimpan pada SD Card
    myFile = SD.open("data.csv", FILE_WRITE);
    if (myFile) {
        myFile.print(tanggal + "/" + bulan + "/" + tahun);
        myFile.print(",");
        myFile.print(jam + ":" + menit + ":" + detik);
        myFile.print(",");

myFile.println(String(tinggiair)+(",")+String(mmTotal)+(",")+String(predict));
        myFile.close(); // close the file
        Serial.println("Data berhasil ditulis ke file data.csv");
    }
    else {
        Serial.println("Gagal membuka file untuk edit data.csv");
        // Serial.print(kirimdata);
    }

//Menampilkan informasi pada LCD
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("H:");
    lcd.setCursor(2,1);
    lcd.print(tinggiair);
    lcd.setCursor(7,1);
    lcd.print("cm");
    lcd.setCursor(9,0);
    lcd.print("Potensi");
    lcd.setCursor(10,1);
    lcd.print(predictshort);
    lcd.setCursor(0,0);
    lcd.print("R:");
    lcd.setCursor(2,0);
    lcd.print(mmTotal);

```

```

    lcd.setCursor(6,0);
    lcd.print("mm");
    mmTotal = 0;
    chTotal = 0;
    delay(1000);
    a = hasil_defuzzyfikasi;
}

//Membunyikan Buzzer
if ((a >= 66.67) && (RTC.seconds == 3)) {
    tone(buzzerPin, 1000);
}
if ((a >= 66.67) && (RTC.seconds == 8)){
    noTone(buzzerPin);
}
CH(); //Memanggil fungsi perhitungan curah hujan dan tinggi air
//Serial.println(mmTotal);
TMA();

//Curah hujan dan tinggi air menjadi 0 saat pertama kali menjalankan sistem
if(tmp == 0){r = 0;} else {r = resolusi;}
tmp = tmp + 1;
if(tmp == 553){ tmp = 0;}
delay(50);
}

void CH() { //Fungsi perhitungan curah hujan
    sensor = digitalRead(9);
    if (sensor != keadaanawal) {
        mmTotal = mmTotal + r;
        const float batas_bawah = 0; //memberi batasan nilai untuk curah hujan untuk
        perhitungan potensi banjir
        const float batas_atas = 20;
        chTotal = constrain(mmTotal, batas_bawah, batas_atas);
    }
    keadaanawal = sensor;
}

void TMA() { //Fungsi perhitungan tinggi air
    digitalWrite(trigPin, LOW);
    delayMicroseconds(10);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    int TravelTime = pulseIn(echoPin, HIGH);
    float jarakukur = ((TravelTime + 0.4484)/55.639); //Fungsi transfer
    float tinggi = 50 - jarakukur;

```

```

const int batas_bawah = 0;
const int batas_atas = 36;
tinggiair = constrain(tinggi, batas_bawah, batas_atas); //Memberi batasan nilai
pada ketinggian air untuk perhitungan potensi banjir
// Serial.println(tinggiair);
}

```

//Perhitungan Potensi Banjir dengan Logika Fuzzy Mamdani

// Fungsi keanggotaan HASIL

```

float hasil_potensi (float _SR1, float _SR2, float _SR3, float _SR4, float _SR5,
float _SR6, float _SR7, float _SR8, float _SR9, float PRED) {
    float c1 = clipping_potensirendah_PRED (_SR1, PRED);
    float c2 = clipping_potensirendah_PRED (_SR2, PRED);
    float c3 = clipping_potensisedang_PRED (_SR3, PRED);
    float c4 = clipping_potensirendah_PRED (_SR4, PRED);
    float c5 = clipping_potensisedang_PRED (_SR5, PRED);
    float c6 = clipping_potensitinggi_PRED (_SR6, PRED);
    float c7 = clipping_potensisedang_PRED (_SR7, PRED);
    float c8 = clipping_potensitinggi_PRED (_SR8, PRED);
    float c9 = clipping_potensitinggi_PRED (_SR9, PRED);

```

```

    float max123456789;
    float max12 = max(c1, c2);
    float max34 = max(c3, c4);
    float max56 = max(c5, c6);
    float max78 = max(c7, c8);
    float max1234 = max(max12, max34);
    float max5678 = max(max56, max78);
    float max12345678 = max(max1234, max5678);
    return max123456789 = max(max12345678, c9);
}

```

// Hasil min(SR, Potensi)

```

float clipping_potensirendah_PRED (float SR, float PRED) {
    float value = min(SR, potensirendah_PRED(PRED));
    return value;
}

```

```

float clipping_potensisedang_PRED (float SR, float PRED) {
    float value = min(SR, potensisedang_PRED(PRED));
    return value;
}

```

```

float clipping_potensitinggi_PRED (float SR, float PRED) {
    float value = min(SR, potensitinggi_PRED(PRED));
    return value;
}

```

```

}

// Fungsi keanggotaan Tinggi Air
float rendah_TMA (float TMA) {
    float value = 0;
    if (TMA <= 6) value = 1;
    if (TMA > 6 && TMA <18) value = 1.5 - TMA/12;
    return value;
}

float sedang_TMA (float TMA) {
    float value = 0;
    if (TMA > 6 && TMA <= 18) value = TMA/12 - 0.5;
    if (TMA > 18 && TMA <30) value = 2.5 - TMA/12;
    return value;
}

float tinggi_TMA (float TMA) {
    float value = 0;
    if (TMA >= 30) value = 1;
    if (TMA > 18 && TMA <30) value = TMA/12 - 1.5;
    return value;
}

// Fungsi keanggotaan Curah Hujan
float gerimis_CH (float CH) {
    float value = 0;
    if (CH <= 3.33) value = 1;
    if (CH > 3.33 && CH <10) value = 1.49925037 - CH/6.67;
    return value;
}

float normal_CH (float CH) {
    float value = 0;
    if (CH > 3.33 && CH <= 10) value = CH/6.67 - 0.49925037;
    if (CH > 10 && CH <16.67) value = 2.49925037 - CH/6.67;
    return value;
}

float lebat_CH (float CH) {
    float value = 0;
    if (CH >= 16.67 ) value = 1;
    if (CH > 10 && CH <16.67) value = CH/6.67 - 1.49925037;
    return value;
}

// Fungsi keanggotaan Potensi Banjir

```

```
float potensirendah_PRED (float PRED) {  
    float value = 0;  
    if (PRED < 33.33) value = 1 - PRED/33.33;  
    return value;  
}
```

```
float potensisedang_PRED (float PRED) {  
    float value = 0;  
    if (PRED >= 33.33 && PRED <= 50) value = PRED/16.67 - 1.99940012;  
    if (PRED > 50 && PRED <= 66.66) value = 4.00120048 - PRED/16.66;  
    return value;  
}
```

```
float potensitinggi_PRED (float PRED) {  
    float value = 0;  
    if (PRED > 66.66 ) value = PRED/33.34 - 1.99940012;  
    return value;  
}
```

Lampiran 9. Script NodeMCU

//Muhammad Yahya Departemen Geofisika Universitas Hasanuddin

//NodeMCU untuk menerima data dari Arduino kemudian dikirim ke database dan mengirim notifikasi Telegram

```
//Import Library
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>

#define BOTtoken "6128552023:AAF5KI3XzTJQ-qUSAfjloyq0D5_8tbteH-A"
// Bot token dari botfather Telegram
#define CHAT_ID "991631953" // Enter your chatID you got from chatid bot
X509List cert(TELEGRAM_CERTIFICATE_ROOT);
WiFiClientSecure client;
UniversalTelegramBot bot(BOTtoken, client);

//Inisialisasi
String kirimdata;
String dt[3];
String predict;
int i;
boolean parsing = false;

//Wi-Fi
const char* ssid = "HUAWEI-8C67";
const char* password = "B655H3FM4DY";

void setup() {
  Serial.begin(9600);
  configTime(0, 0, "pool.ntp.org");
  client.setTrustAnchors(&cert);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) { //Menghubungkan ke Wi-Fi
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");
  bot.sendMessage(CHAT_ID, "Wifi Connected!", "");
  delay(1000);
  bot.sendMessage(CHAT_ID, "System has Started!", "");
}
```

```

//Fungsi untuk parsing data yang diterima dari Arduino
void parsingData() {
    int j=0;
    dt[j]="";
    for (i=1; i< kirimdata.length();i++){
        if ((kirimdata[i] == ',')){
            j++;
            dt[j]="";
        }
        else{
            dt[j] += kirimdata[i];
        }
    }
}

boolean tmp_fix = false;
void loop() {
    if(Serial.available()>0){
        char inChar = (char)Serial.read();
        if(inChar == '#'){
            tmp_fix = true;
        }
        if(tmp_fix){
            kirimdata += inChar;
            if (inChar == '\n') {parsing = true;tmp_fix=false;}
        }
    }
    if (parsing){
        parsingData();
        parsing=false;
        kirimdata="";

        // Mengirim data ke script PHP
        WiFiClient client;
        HTTPClient http;
        http.begin(client, "http://gitgif.com/phpscript.php");
        http.addHeader("Content-Type", "application/x-www-form-urlencoded");
        String predict;
        if (dt[2].toInt() <= 33.33){
            predict = "Potensi Rendah";
        } else if (dt[2].toInt() <= 66.66){
            predict = "Potensi Sedang";
        } else {
            predict = "Potensi Tinggi";
        }
    }
}

```

```

        String data = "data1=" + String(dt[0]) + "&data2=" + String(dt[1]) + "&data3="
+ String(predict);
        int httpCode = http.POST(data);
        String response = http.getString();
        Serial.println(response);
        http.end();

//Mengirim notifikasi Telegram
        if ((dt[2].toInt() >= 66.67)) {
            bot.sendMessage (CHAT_ID,"Waspada!. Tinggi air "+dt[0]+" cm dan
"+dt[1]+" mm. Potensi tinggi terjadinya banjir.", "");
        }
    }
}

```