

Rancang Bangun *Wearable Coronavirus Realtime Updater*

**Agus Ramelan¹, Nanda Hafidz Rivanda¹, Hari Maghfiroh¹, Irwan Iftadi¹,
Muhammad Adli Rizqulloh²**

¹ *Program Studi Teknik Elektro, Fakultas Teknik, Universitas Sebelas Maret, Solo*
¹ *Departemen Pendidikan Teknik Elektro, Fakultas Pendidikan Teknologi dan Kejuruan,
Universitas Pendidikan Indonesia, Bandung*

E-mail: agusramelan@staff.uns.ac.id

Received: 22-12-2020

Accepted: 31-12-2020

Published: 31-12-2020

Abstrak

Pandemi covid-19 sejak akhir 2019 lalu telah menjadi perhatian khusus dunia. Satuan tugas khusus telah dibentuk pemerintah untuk memonitoring dan menangani permasalahan ini. Dalam artikel ini dipaparkan sebuah rancang bangun *wearable devices* yang *portable* dan *low-cost* untuk menampilkan jumlah kasus Covid-19 secara *real time*. Metode yang digunakan yang digunakan adalah perancangan hardware, web scrapping, dan integrasi sistem dengan komunikasi berbasis internet (IoT). Dari hasil percobaan, data time series dan jumlah penyebaran virus terupdate berhasil diambil dari pihak ketiga dengan cara *web scrapping*. Data tersebut kemudian diolah oleh mikrokontroler Wemos D1 dan ditampilkan pada OLED 0.96 inch. Data yang ditampilkan pada alat ini sama dengan data yang tertampil pada website pihak ketiga yang diambil.

Kata kunci: *Covid-19, Wearable, Real-Time Updater, IoT*

Abstract

The covid-19 pandemic since late 2019 has been a special concern in the world. A special task force has been set up by the government to monitor and mitigates these issues. This article features a design to portable and low-cost wearable devices to display the number of Covid-19 cases in real time. The methods used are hardware design, web scrapping, and system integration with Internet-based communication (IoT). From the results of the experiment, the time series data and the number of updated virus spreads were successfully retrieved from third parties by web scrapping. The data is then processed by the Wemos D1 microcontroller and displayed on a 0.96 inch OLED. The data displayed on this tool is the same as the data displayed on the third party web-sites taken.

Keywords: *Covid-19, Wearable, Real-Time Updater, IoT*

Pendahuluan

Pada akhir tahun 2019, WHO China Country Office melaporkan kasus pneumonia yang tidak diketahui etiologinya di Kota Wuhan, Provinsi Hubei, Cina. Pada tanggal 7 Januari 2020, Cina mengidentifikasi pneumonia yang tidak diketahui etiologinya tersebut sebagai jenis baru coronavirus (novel coronavirus, 2019-nCoV). Penambahan jumlah kasus 2019-nCoV berlangsung cukup cepat dan sudah terjadi penyebaran ke luar wilayah Wuhan dan negara lain. Sampai dengan 26 Januari 2020, secara global 1.320 kasus terkonfirmasi di 10 negara dg 41 kematian (CFR 3,1%). Rincian China 1297 kasus konfirmasi (termasuk Hongkong, Taiwan, dan Macau) dengan 41 kematian (39 kematian di Provinsi Hubei, 1 kematian di Provinsi Hebei, 1 kematian di Provinsi Heilongjiang), Jepang (3 kasus), Thailand (4 kasus), Korea Selatan (2 kasus), Vietnam (2 kasus), Singapura (3 kasus), USA (2 kasus), Nepal (1 kasus), Perancis (3 kasus), Australia (3 kasus). Diantara kasus tersebut, sudah ada beberapa tenaga kesehatan yang dilaporkan terinfeksi. Sampai dengan 24 Januari 2020, WHO melaporkan bahwa penularan dari manusia ke manusia terbatas (pada kontak keluarga) telah dikonfirmasi di sebagian besar Kota Wuhan, China dan negara lain [1].

Berdasarkan pandemi tersebut telah diciptakan semacam parameter penghitung jumlah atas kasus positif, kematian, yang berhasil sembuh, dan sebagainya terhadap masyarakat di dunia melalui sebuah website. Parameter tersebut digunakan untuk membantu analisis atas perkembangan dari coronavirus tersebut agar setiap orang dapat melihat perkembangan penyebaran Covid-19. Setiap orang berhak melihat perkembangan tersebut, namun bagi beberapa orang seperti peneliti, dokter, satuan tugas khusus maupun masyarakat sendiri untuk melihat perkembangan dengan membuka website tertentu membutuhkan waktu dan peran interaksi dari pengguna.

Oleh karena itu dalam artikel ini diusulkan sebuah alat yang bersifat *portable* dan *wearable* yang berfungsi sebagai penampil jumlah kasus Covid-19 secara *real time*. Alat ini bernama "Wearable Coronavirus Realtime Updater". Alat ini cukup kecil dan mudah dibawa. Selain itu alat ini dapat menunjukkan perkembangan dari coronavirus sesuai di situs pihak ketiga, misalnya (<https://www.worldometers.info/>). Karena alat ini mengambil dan memisahkan data dari situs tersebut saat alat ini terhubung oleh jaringan internet yang biasa disebut *Internet of Things*. Sehingga bagi beberapa orang seperti peneliti, dokter, satgas khusus penanganan covid, maupun masyarakat sendiri dapat mengetahui

perkembangan jumlah penyebaran virus Covid-19 dengan mudah dan secara *real-time*. Alat ini memiliki fitur kebaruan yaitu sebagai berikut:

- Bersifat *wearable* dan *portable*
- Memudahkan untuk melihat perkembangan coronavirus baik di dunia maupun di Indonesia secara *real time*.
- Dapat menampilkan baik waktu maupun kalender.
- Memiliki sebuah baterai 500 mAh dan dapat di *recharge*.

Wearable Coronavirus Realtime Updater ini menggunakan teknologi *Internet of Things*. *Internet of Things (IoT)* secara umum diartikan sebagai sebuah kemampuan untuk menghubungkan objek-objek cerdas dan memungkinkannya untuk berinteraksi dengan objek lain, lingkungan maupun dengan peralatan komputasi cerdas lainnya melalui jaringan internet [2]. Peran dan kontribusi IoT dalam penanganan Covid-19 sudah mulai banyak dijelaskan dalam studi-studi komprehensif [3][4][5][6]. Di samping itu, beberapa penelitian aplikasi teknologi IoT dalam penanganan Covid-19 juga sudah mulai banyak dilakukan, dalam *healthcare*, *tracking*, *preventing*, dan beberapa aplikasi lainnya [7][8][9][10].

Metode Penelitian

Pada *prototype* ini dilakukan dengan beberapa langkah dimulai dari *Web Scrapping*, *Generate API* pada Thingspeak, dan *Request HTTP API* pada *microcontroller*.

A. Web Scrapping

Web scrapping merupakan proses mengekstraksi informasi yang berguna dari sebuah halaman HTML yang dapat diimplementasikan menggunakan bahasa skrip yang dikenal sebagai *Prolog Server Pages (PSP)* berdasarkan Prolog [11]. *Web Scrapping* ini menggunakan sebuah browser sebagai contoh menggunakan google chrome. *Web* yang digunakan untuk *scrapping* antara lain <https://www.worldometers.info/coronavirus/#countries> untuk analisa perkembangan coronavirus dan <https://time.is/Chilachap> untuk mengambil waktu secara tepat. Berikut adalah urutan metode *web scrapping* yang dilakukan.

- Membuka browser
- Masuk ke sebuah situs <https://time.is/> dan cari lokasi yang tepat.

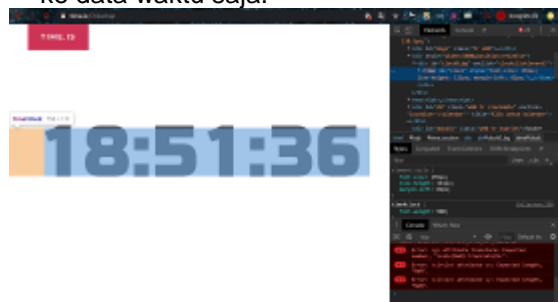


Gambar 1. Tampilan waktu di situs time.is

#	Country/Other	Total Cases	New Cases	Total Deaths	New Deaths	Total Recovered	Active Cases	Serious, Critical	Total Cases/1M pop	Deaths/1M pop	Total Tests	Tests/1M pop
1	World	79,853,291	+130,893	1,751,733	+3,162	56,247,593	21,853,965	105,202	10,244	224.7		
2	USA	19,111,326		337,066		11,219,123	7,866,137	28,776	67.675	1.016	242,068,492	729.1
3	India	10,147,468		147,128		9,717,834	282,506	8,844	7.319	106	168,305,762	119.1
4	Brazil	7,425,583		190,032		6,449,822	785,739	8,318	34.815	891	28,600,000	134.1
5	Russia	2,992,798	+29,018	63,659	+463	2,398,254	540,793	2,300	20.503	368	88,094,380	603.1
6	France	2,627,809		62,268		188,639	2,276,602	2,652	38.680	963	32,361,074	495.1
7	UK	2,188,587		69,625		N/A	N/A	1,629	32.158	1,023	52,257,888	787.1
8	Turkey	2,100,712		19,115		1,935,292	146,305	4,806	24.780	226	23,225,837	273.1
9	Italy	2,009,317		70,900		1,344,785	593,632	2,689	33.257	1,173	25,752,360	426.1
10	Spain	1,869,610		49,824		N/A	N/A	1,907	39.980	1,065	25,991,493	555.1
11	Germany	1,614,326		29,681		1,184,400	400,245	6,354	19.238	354	33,708,381	481.1
12	Argentina	1,671,880		42,392		1,384,277	145,011	3,390	34.622	934	4,997,960	101.1
13	Colombia	1,559,796		41,454		1,417,316	100,996	2,678	30.498	811	7,755,989	161.1
14	Mexico	1,362,664	+12,485	121,172	+661	1,016,266	226,136	3,913	10.616	936	3,483,572	26.1

Gambar 4. Situs www.worldometers.info/coronavirus/

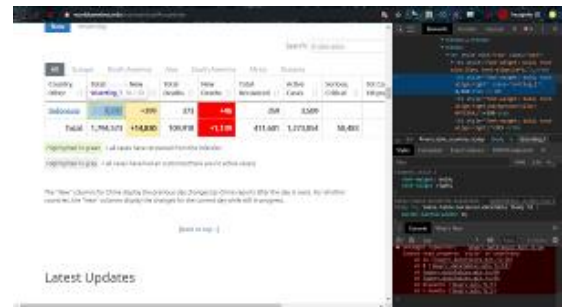
- c. Blok waktu dan klik kanan lalu pilih *Inspect element*.
- d. Carilah *block kode* html yang hanya merujuk ke data waktu saja.



Gambar 2. Block Code yang menunjukkan data waktu

- i. Karena akan menampilkan kasus di Indonesia dan total seluruh dunia, maka pada kolom search ketik "Indonesia"
- j. Selanjutnya *web scraping* pada data *cases*, *death*, *recovered* di Indonesia dan total seluruhnya.
- k. Langkah *web scraping* sama seperti sebelumnya, *block* data yang diinginkan pada web, lalu *inspect element*, dan cari *block code html* yang sesuai dan *copy xpath*, dan simpan *xpath* tersebut satu-satu. *Xpath* tersebut digunakan pada generate API di thingspeak.

- e. Klik kanan *block code* dan pilih *copy Xpath*
- f. Simpan *xpath* tersebut sebagai waktu jika dipaste akan menunjukkan body parsing web tersebut (`//*[@id="clock"]`)
- g. Sama seperti sebelumnya, data yang diambil selanjutnya kalender. Dan simpan *xpath* tersebut sebagai kalender



Gambar 5. Proses web scraping data



Gambar 3. Web scraping data kalender

A. Generate API pada Thingspeak

Thingspeak merupakan sebuah layanan web sebagai server untuk *Internet of Things*. Untuk generate API pada thingspeak menggunakan layanan ThingsHTTP.

- h. Selanjutnya buka situs <https://www.worldometers.info/coronavirus> dan menuju ke tabel *country*.

- a. Buka situs <https://thingspeak.com/> dan sign in.
- b. Pilih apps lalu masuk pada thingsHTTP
- c. Lalu buat thingHTTP pada "New ThingsHTTP"
- d. Isikan nama ThingHTTP, URL yaitu situs yang akan di-*web scraping*, dan Parse String yaitu *xpath* dari data situs tersebut.

- e. Jika diakhir dari xpath tidak ada “ /text() “ maka tambahkan. “ /text() “ ini berguna agar saat Request HTTP yang dipanggil hanya menampilkan datanya saja bukan dengan *block code*.
- f. Ulangi langkah tersebut untuk data kalender, *cases*, *death*, *recovered* pada Indonesia dan Total seluruhnya.
- g. Simpan semua URL API tersebut untuk digunakan pada *microcontroller*.

B. Request HTTP API pada Microcontroller

Selanjutnya Request HTTP pada microcontroller dilakukan pada Wemos D1 dan di program melalui aplikasi arduino IDE. Indentifikasi host yang digunakan yaitu "api.thingspeak.com" dan menggunakan port 80, url yaitu berisi request API yang terdapat dari thingsHTTP. Perintah tersebut untuk membuat klien dengan variabel nama http.

```

if( http.begin(host,httpPortRead,uz00))
{
  int httpCode = http.GET();
  if (httpCode > 0)
  {
    if (httpCode == HTTP_CODE_OK || httpCode == HTTP_CODE_MOVED_PERMANENTLY)
    {
      String payload = http.getString();
      Serial.println(payload);
    }
  }
}
    
```

Gambar 6. GET Request HTTP API

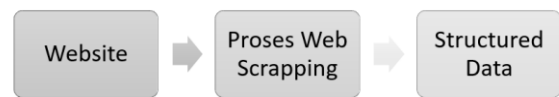
Perintah diatas merupakan untuk malakukan GET Request HTTP jika terhubung akan terbaca payload atau isi dari HTTP API tersebut.

Hasil dan Diskusi

A. Prinsip Kerja

Cara kerja *Coronavirus Realtime Updater Wearable* adalah dengan cara mengambil dan memisahkan data dari situs resmi (<https://www.worldometers.info/>) atau tekniknya disebut *web scrapping*. *Web Scraping* adalah proses pengambilan sebuah dokumen semi-terstruktur dari internet, umumnya berupa halaman-halaman web dalam bahasa markup seperti HTML atau XHTML, dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut untuk digunakan bagi kepentingan lain [12].

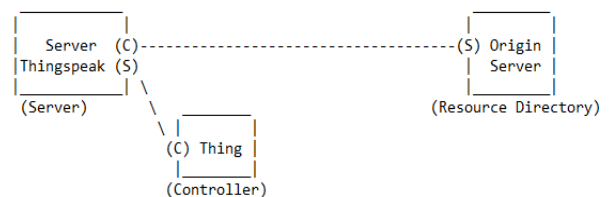
Web scraping menggunakan metode Xpath. XPath adalah bahasa yang kuat yang sering digunakan untuk web scrapping. Ini memungkinkan Anda untuk memilih node atau menghitung nilai dari dokumen XML atau HTML dan sebenarnya adalah salah satu bahasa yang dapat di gunakan dari sebuah situs.



Gambar 7. Diagram alur Web Scrapping

Setelah data di ambil dan dipisahkan dari markup HTML selanjutnya dibuat API (*Application programming interface*). Application programming interface (API) merupakan suatu dokumentasi yang terdiri dari *interface*, fungsi, kelas, struktur dan sebagainya untuk membangun sebuah perangkat lunak. Dengan adanya API ini, maka memudahkan programmer untuk “membongkar” suatu software, kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya yang memungkinkan programmer menggunakan sistem *function* [13].

API ini di-generate dengan bantuan server dari Thingspeak, sehingga setelah di-generate dapat diimplementasikan ke *microcontroller* Wemos D1. Dimana Wemos D1 ini akan menjadi otak yang menerima data dari situs resmi dan ditampilkan pada LCD kecil. Secara tidak langsung teknologi *Internet of Things* ini menggunakan protocol HTTP dengan arsitektur *The Representational State Transfer* (REST). REST adalah seperangkat pedoman dan praktik terbaik untuk membangun sistem hypermedia terdistribusi. Komponen dari sistem RESTfull yaitu klien atau *server*.



Gambar 8. Arsitektur REST dengan Thingspeak

Bagian terpenting dari REST adalah URIs. URIs digunakan untuk menunjukkan sumber daya untuk interaksi, untuk referensi sumber daya dari sumber daya lain, untuk mengiklankan atau menandai sumber daya, atau untuk mengindeks sumber daya oleh mesin pencari.

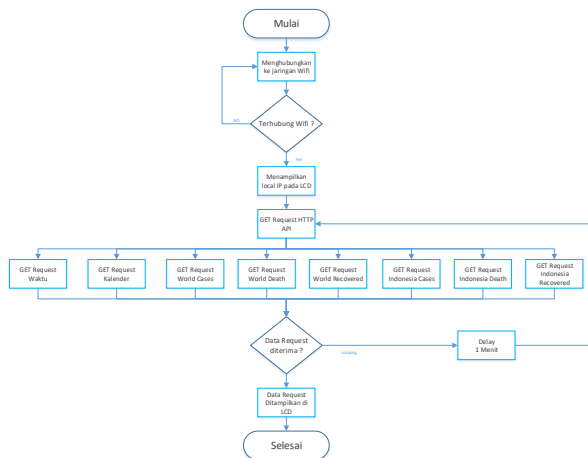
Metode HTTP yang digunakan pada alat ini adalah GET. Metode GET meminta representasi saat ini untuk sumber daya target, sementara server asal harus memastikan bahwa tidak ada

efek samping pada kondisi sumber daya. Hanya server asal yang perlu mengetahui bagaimana masing-masing pengidentifikasi sumber dayanya sesuai dengan suatu implementasi dan bagaimana setiap implementasi mengelola untuk memilih dan mengirim representasi terkini dari sumber daya target dalam respons terhadap GET [14].

Secara garis besar, berikut arsitektur sistem alat ini :

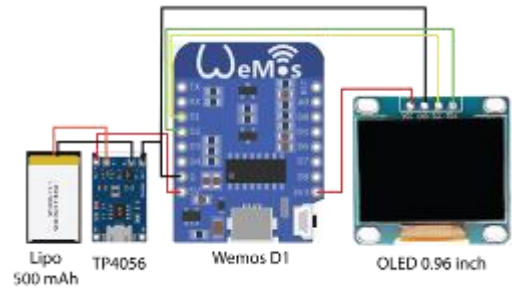
- a. *Perception Layer*
Wemos D1 (sebagai microcontroller yang terintegrasi esp8266)
- b. *Network Layer*
 - Layer Physical : Wifi Router (IEEE 802.11)
 - Layer Data Link : IEEE 802.11 wireless LAN
 - Layer Network : IPv4
 - Layer Transport : TCP
 - Layer Application : HTTP
- c. *Application Layer*

Thingspeak sebagai generate API atas web scrapping agar dapat digunakan dengan mudah oleh microcontroller. Berikut diagram alir sistem :



Gambar 9 Flowchart Coronavirus Realtime Updater Wearable

Berikut schematic sistem :



Gambar 10. Skematik dari Wearable Corona Virus Realtime Updater

B. Hasil Scrapping Data

Setelah data dari web diambil dan dipisahkan dengan menggunakan *web scrapping* pada situs <https://time.is/> dan <https://www.worldometers.info/coronavirus> dengan metode xpath.

Contoh hasil copy dari xpath :

```
//*[@@id="clock"]/text()
```

Perintah “ /text() “ digunakan agar hasil request yang tertampil hanya datanya bukan serta *block code html* nya.

Maka selanjutnya dibuatlah API yang di *generate* oleh ThingsHTTP di Thingspeak.

Contoh hasilnya adalah :

```
https://api.thingspeak.com/apps/thinghttp/send_request?api_key=K93JAT9XF094DF50
```

API tersebut digunakan untuk melakukan *GET Request HTTP* pada microcontroller setiap 1 menit dan akan ditampilkan pada LCD.

Berikut hasil yang ditampilkan :

```
192.168.43.226
Sunday, April 12, 2020
08:28:36
Indonesian Cases: 4,241
Indonesian Deaths: 373
Indonesian Recovered: 359
World Cases: 1,796,052
World Deaths: 110,012
World Recovered: 412,049
```

Gambar 11. Serial Monitor Wearable Coronavirus Realtime Updater

Simpulan

Wearable Coronavirus Realtime Updater ini merupakan alat yang dapat membantu beberapa orang agar dapat melihat hasil perkembangan dari virus corona secara cepat dan mudah. Alat ini dibekali baterai 500 mAh dan *rechargeable*. Alat ini menggunakan teknologi *Internet of Thing*, yang mengambil data dari situs resmi dengan metode web scraping yang selanjutnya diolah menjadi API pada ThingsHTTP, agar dapat diimplementasikan pada microcontroller dengan metode GET Request API. Arsitektur yang digunakan pada sistem ini adalah arsitektur *The Representational State Transfer (REST)* yang merupakan seperangkat pedoman dan praktik terbaik untuk membangun sistem hypermedia terdistribusi. Dan menggunakan protocol HTTP, karena menggunakan metode GET Request API untuk mengambil data setelah di web scraping.

Ucapan Terima Kasih

Ucapan terima kasih disampaikan kepada Laboratorium Internet of Things (IoT) FT UNS atas dukungan fasilitas yang telah diberikan.

Daftar Pustaka

- [1] Direktorat Jenderal Pencegahan dan Pengendalian Penyakit (P2P). Pedoman Kesiapsiagaan Menghadapi Infeksi Novel Coronavirus (2019-nCoV), Jakarta Selatan: Kementerian Kesehatan RI, 2020.
- [2] Ernita, "Internet Of Things - Keamanan Dan Privasi," *Jurnal Seminar Nasional dan Expo Teknik Elektro*, 2015.
- [3] V. Chamola, V. Hassija, V. Gupta and M. Guizani, "A Comprehensive Review of the COVID-19 Pandemic and the Role of IoT, Drones, AI, Blockchain, and 5G in Managing its Impact," in *IEEE Access*, vol. 8, pp. 90225-90265, 2020, doi: 10.1109/ACCESS.2020.2992341.
- [4] S. S. Kumar, "Emerging Technologies and Sensors That Can Be Used During the COVID-19 Pandemic," 2020 International Conference on UK-China Emerging Technologies (UCET), Glasgow, United Kingdom, 2020, pp. 1-4, doi: 10.1109/UCET51115.2020.9205424.
- [5] M. Ndiaye, S. S. Oyewobi, A. M. Abu-Mahfouz, G. P. Hancke, A. M. Kurien and K. Djouani, "IoT in the Wake of COVID-19: A Survey on Contributions, Challenges and Evolution," in *IEEE Access*, vol. 8, pp. 186821-186839, 2020, doi: 10.1109/ACCESS.2020.3030090.
- [6] S. Jaafari, A. Alhasani, E. alghosn, R. alfahhad and S. M. Almutairi, "Certain Investigations on IoT system for COVID-19," 2020 International Conference on Computing and Information Technology (ICCIT-1441), Tabuk, Saudi Arabia, 2020, pp. 1-4, doi: 10.1109/ICCIT-144147971.2020.9213760.
- [7] A. Roy, F. H. Kumbhar, H. S. Dhillon, N. Saxena, S. Y. Shin and S. Singh, "Efficient Monitoring and Contact Tracing for COVID-19: A Smart IoT-Based Framework," in *IEEE Internet of Things Magazine*, vol. 3, no. 3, pp. 17-23, September 2020, doi: 10.1109/IOTM.0001.2000145.
- [8] S. S. Vedaiei et al., "COVID-SAFE: An IoT-Based System for Automated Health Monitoring and Surveillance in Post-Pandemic Life," in *IEEE Access*, vol. 8, pp. 188538-188551, 2020, doi: 10.1109/ACCESS.2020.3030194.
- [9] S. Karmore, R. Bodhe, F. Al-Turjman, R. L. Kumar and S. Pillai, "IoT Based Humanoid Software for Identification and Diagnosis of Covid-19 Suspects," in *IEEE Sensors Journal*, doi: 10.1109/JSEN.2020.3030905.
- [10] B. Benreguia, H. Moumen and M. A. Merzoug, "Tracking COVID-19 by Tracking Infectious Trajectories," in *IEEE Access*, vol. 8, pp. 145242-145255, 2020, doi: 10.1109/ACCESS.2020.3015002.
- [11] S. K. Malik and S. Rizvi, "Information Extraction Using Web Usage Mining, Web Scrapping and Semantic Annotation," 2011 International Conference on Computational Intelligence and Communication Networks, Gwalior, 2011, pp. 465-469, doi: 10.1109/CICN.2011.97.
- [12] M. Turland, "php| architect's Guide to Web," Vols. str, 2, pp. Introduction-Web Scrapping, 2010.
- [13] M. F. Ramadhani, "Pembangunan Aplikasi Informasi, Pengaduan, Kritik, Dan Saran Seputar Kota Cimahi Pada Platform Android," 2015.
- [14] E. M. K. A. Keranen, "RESTful Design for Internet of Things Systems," 18 March 2018. [Online]. Available: <https://tools.ietf.org/id/>. [Accessed 12 April 2020].