

BAB II KAJIAN PUSTAKA

A. Studi Pustaka

Studi Pustaka yang digunakan pada penelitian ini menggunakan beberapa penelitian terkait yang dijadikan sebagai referensi dalam penulisan ini. Penelitian terkait yang digunakan sebagai referensi berjarak lima (5) tahun terhitung kebelakang dari tahun penelitian saat ini, berikut tabel penelitian terkait

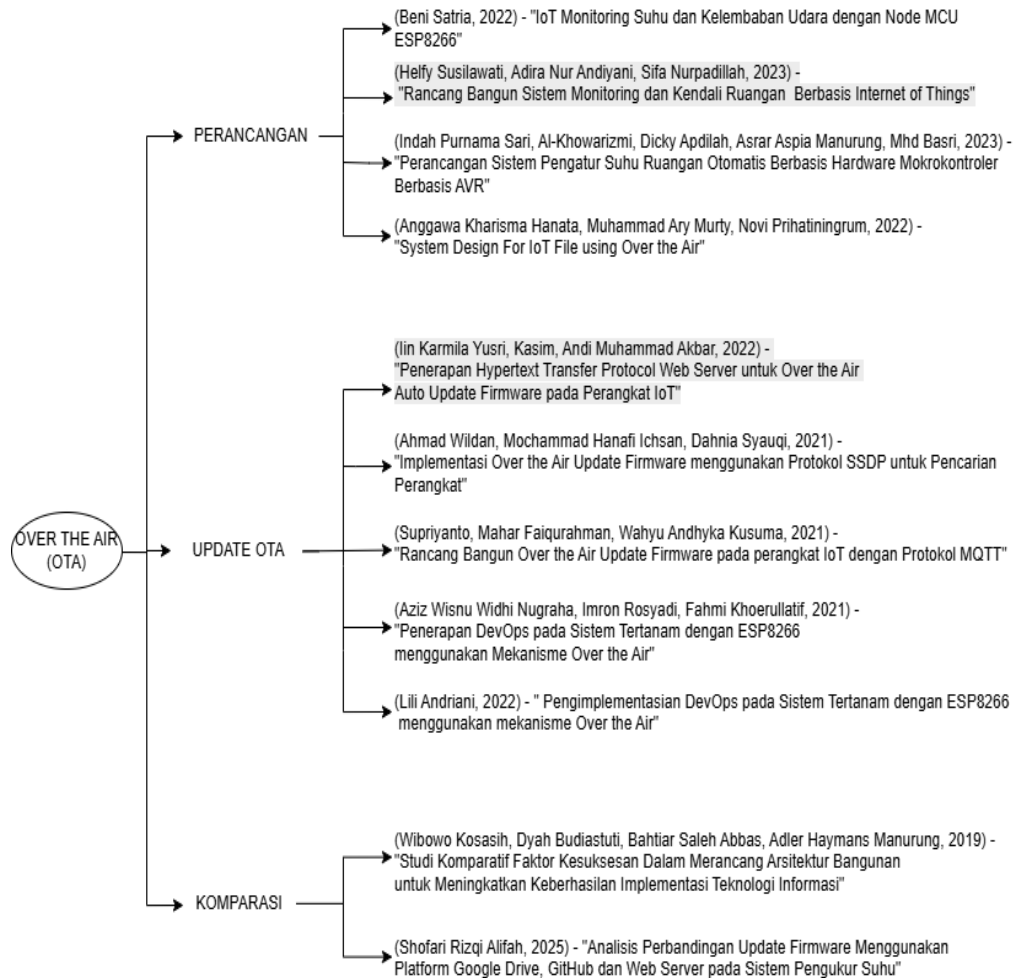
Tabel 1 Penelitian Terkait

No	Nama (Tahun)	Judul
(1)	(2)	(3)
1.	F. Rak dan J. Wiora (2021)	Judul: Perbandingan <i>platform</i> pemrograman ESP Kesimpulan: Pemilihan bahasa pemrograman ESP optimal bervariasi sesuai keahlian pengguna dan tujuan proyek: MicroPython/C++ untuk pemula/proyek sederhana; Lua/C++ untuk menengah/kompleks non-komersial; serta C (RTOS)/Lua untuk lanjutan/komersial.
2.	A. Wildan, M. H. H. Ichsan, and D. Syauqy (2021)	Judul: Implementasi <i>Over the Air Update</i> menggunakan protokol SSDP untuk pencarian perangkat. Kesimpulan: Pembaruan <i>firmware</i> perangkat IoT menggunakan metode Over-The-Air (OTA) dan protokol SSDP berhasil diimplementasikan. Hasil pengujian menunjukkan bahwa semakin dekat jarak perangkat ke router, semakin cepat proses pencarian dan pembaruan perangkat dilakukan.
3.	B. Satria(2022)	Judul: IoT Monitoring Suhu dan Kelembaban Udara dengan NodeMCU ESP8266 Kesimpulan: Sistem monitoring suhu dan kelembaban berbasis IoT menggunakan NodeMCU ESP8266 dan sensor DHT11 berhasil diimplementasikan dan dapat memantau data secara real-time melalui aplikasi Blynk di smartphone
4.	W. Kosasih, D. Budiastuti, B. S. Abbas, dan A. H.	Judul: Studi Komparatif Faktor Kesuksesan dalam Merancang Arsitektur Bangunan untuk Meningkatkan Keberhasilan Implementasi Teknologi Informasi Kesimpulan:

No	Nama (Tahun)	Judul
(1)	(2)	(3)
	Manurung (2019)	Industri konstruksi dapat menjadi rujukan bagi implementasi teknologi informasi, dengan menekankan pentingnya panduan desain (design guideline) berbasis tiga karakter: Governance, Operation, dan Growth untuk meningkatkan keberhasilan implementasi TI di berbagai sektor industri.
5.	A. W. W. Nugraha, I. Rosyadi, dan F. Khoerullatif, (2021)	Judul: Penerapan DevOps pada Sistem Tertanam dengan ESP8266 menggunakan Mekanisme <i>Over the Air</i> Kesimpulan: DevOps berhasil diterapkan pada sistem tertanam berbasis ESP8266, memungkinkan proses <i>build</i>, <i>test</i>, dan <i>deployment firmware</i> secara otomatis dengan waktu rata-rata 77,21 detik, serta waktu <i>deployment</i> hanya 1,41 detik.
6.	I. P. Sari, Al-Khowarizmi, D. Apdilah, A. A. Manurung, dan M. Basri (2023)	Judul: Perancangan Sistem Pengaturan Suhu Ruangan Otomatis Berbasis Mikrokontroler Berbasis AVR Kesimpulan: Penelitian ini merancang sistem otomatis pengatur suhu ruangan berbasis mikrokontroler AVR dengan dukungan sensor DHT11, modul infrared, dan konektivitas internet. Sistem mampu membaca suhu dan kelembaban ruangan secara real-time, menyimpan data ke server, dan mengontrol AC secara otomatis. Hasil pengujian menunjukkan tingkat akurasi suhu sebesar 95,84%, kelembaban 91,67%, dan akurasi kendali AC mencapai 99,40%, serta fitur notifikasi bekerja 100% melalui Telegram.
7.	R. Kusumah, H. I. Islam, dan Susilawati (2023)	Judul: Sistem Monitoring Suhu Kelembaban Berbasis Internet of Things (IoT) pada Ruang Data Center Kesimpulan: Sistem monitoring berbasis IoT berhasil memantau suhu dan kelembaban ruang data center secara akurat, <i>real-time</i>, dan dapat diakses jarak jauh melalui <i>dashboard</i>.
8.	A. K. Hananta, M. A. Murti, dan N. Prihatiningrum (2022)	Judul: <i>System Design for IoT File Firmware Update Using Over the Air Update</i> Kesimpulan:

No	Nama (Tahun)	Judul
(1)	(2)	(3)
		Penelitian ini merancang sistem pembaruan <i>firmware</i> IoT secara <i>Over the Air</i> (OTA) menggunakan NodeMCU ESP8266. Hasilnya, pembaruan berhasil 100%, cepat terhubung ke jaringan baru, dan meskipun sedikit lebih banyak memori terpakai, tidak memengaruhi performa perangkat.
9.	I. K. Yusri, Kasim, dan A. M. Akbar (2022)	Judul: Penerapan <i>Hypertext Transfer Protocol</i> Web Server untuk <i>Over the Air</i> Auto Update <i>firmware</i> pada perangkat IoT Kesimpulan: Penerapan web server berbasis HTTP berhasil digunakan untuk pembaruan <i>firmware</i> otomatis (OTA) pada perangkat IoT. Sistem ini memudahkan pembaruan tanpa kabel dan meningkatkan efisiensi distribusi <i>firmware</i> secara jarak jauh.
10.	Supriyanto, M. Faiqurahman, dan W. A. Kusuma (2021)	Judul: Rancang Bangun <i>Over the Air</i> Update <i>firmware</i> pada Perangkat IoT dengan Protokol MQTT Kesimpulan: Penelitian ini membuat sistem <i>update firmware</i> IoT secara OTA menggunakan MQTT. Hasilnya, QoS 2 paling direkomendasikan karena memiliki tingkat keberhasilan tertinggi (80%) meski menggunakan lebih banyak sumber daya.
11.	H. Susilawati, A. N. Andiyani, dan S. Nurpadillah (2023)	Judul: Rancang Bangun Sistem Monitoring dan Kendali Suhu Ruangan Berbasis Internet of Things Kesimpulan: Sistem ini merancang alat monitoring dan kendali suhu ruangan berbasis IoT di industri farmasi menggunakan NodeMCU ESP8266, sensor DHT11, dan servo SG90. Kendali dilakukan melalui website, memungkinkan pengaturan suhu AC sentral dari jarak jauh. Sistem terbukti akurat dan efisien dalam memantau serta mengatur suhu secara real-time.
12.	K. Madani, R. Hidayati, dan Ristian	Judul: Sistem <i>Update Firmware</i> Perangkat IoT Menggunakan Teknik OTA Berbasis HTTP Kesimpulan:

No	Nama (Tahun)	Judul
(1)	(2)	(3)
		Optimalisasi pemrograman ESP bergantung pada keahlian dan tujuan: MicroPython/C++ untuk pemula, Lua/C++ untuk menengah, dan C (RTOS)/Lua untuk lanjutan. <i>Update</i> OTA via HTTP terbukti efisien untuk <i>firmware</i> IoT.
13.	J. Bauwens, P. Ruckebusch, S. Giannoulis, I. Moerman, and E. De Poorter (2020)	Judul: <i>Over the Air Software Updates</i> in the Internet of Things: An Overview of Key Principles Kesimpulan: Pembaruan perangkat lunak jarak jauh (OTA) untuk IoT, menekankan pentingnya efisiensi energi dan keamanan. Pembaruan modular lebih hemat energi dibandingkan <i>firmware</i> penuh. Proses dibagi dua tahap: manajemen modul dan penyebaran aman.
14.	K. Kerliu, A. Ross, G. Tao, Z. Yun, Z. Shi, S. Han, and S. Zhou (2020)	Judul: <i>Secure Over the Air Firmware Updates for Sensor Networks</i> Kesimpulan: Merancang sistem OTA <i>firmware update</i> yang aman untuk jaringan sensor, memungkinkan satu node memancarkan <i>firmware</i> ke banyak node sekaligus. Sistem menggunakan bootloader khusus, enkripsi, dan protokol efisien. Hasilnya, pembaruan berhasil dalam beberapa putaran transmisi, bahkan di kondisi non-ideal
15.	L. Andriani (2022)	Judul: Pengimplementasian DevOps pada Sistem Tertanam dengan ESP8266 menggunakan Mekanisme <i>Over the Air</i> Kesimpulan: Mengusulkan infrastruktur DevOps untuk sistem tertanam ESP8266 dengan OTA, berhasil mengotomatisasi tahapan build, test, release, dan deploy.



Gambar 1 MindMap

B. Landasan Teori

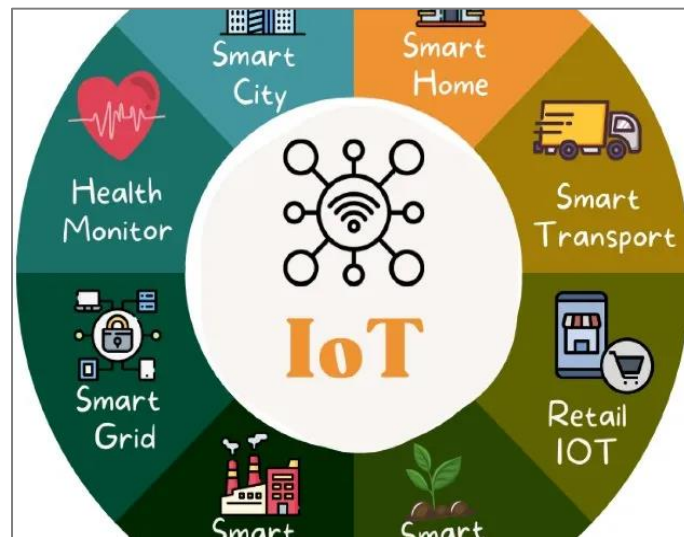
Landasan teori yang digunakan pada penelitian “Analisis Perbandingan *update firmware* menggunakan Google Drive, github dan Web Server pada sistem pengukur suhu” sebagai berikut :

1. Metode Komparatif

Metode Komparatif yaitu metode penelitian yang membandingkan dua obyek atau lebih dengan tujuan menemukan obyek yang dibandingkan[6]. Tujuan dari metode komparatif ini untuk menentukan mana yang lebih efisien sehingga dapat membantu dalam pengambilan keputusan dari pengujian yang dilakukan, selain itu metode komparatif juga dapat mengevaluasi kinerja pada setiap aspek yang diuji [7]. Pada penelitian ini metode komparatif digunakan untuk membandingkan kinerja *platform* google drive, GitHub dan web server pada *update firmware*, aspek yang dibandingkan yaitu kecepatan, penyimpanan, autentikasi akun dan autentikasi perangkat pada saat *update firmware*.

2. *Internet of Things (IoT)*

Internet of Things adalah sebuah gagasan dimana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung[4][1][8]. IoT juga merupakan sebuah konsep dimana suatu benda atau objek ditanamkan teknologi-teknologi seperti sensor dan *software* dengan tujuan untuk berkomunikasi, mengendalikan, menghubungkan dan bertukar data melalui perangkat lain selama masih terhubung internet[1][8].

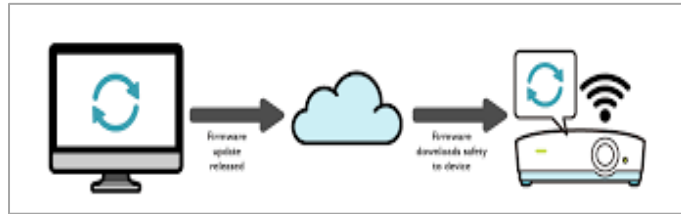


Gambar 2 Ilustrasi Penerapan IoT

Beberapa sektor yang mengimplementasikan IoT mulai dari *smart home*, *smart transportasi*, retail atau toko-toko, *smart agriculture* di bidang pertanian, *smart industry* di budang industri, *smart grid* digunakan untuk mengatur dalam pendistribusian dan pemantauan listrik, *healt monitor* digunakan pada sektor kesehatan untuk memantau kesehatan, dan *smart city* dimana semua serba canggih dalam pengelolaan kota.

3. *Over the Air (OTA)*

Over the Air adalah proses memuat *firmware* pada modul ESP perangkat IoT menggunakan koneksi jaringan Wi-Fi dari pada menggunakan kabel *port* serial, OTA secara umum merupakan mekanisme penggunaan *wireless* untuk mengirim data, memperbarui paket utuk pembaruan *firmware* atau perangkat lunak ke perangkat mobile, sehingga pengguna tidak perlu mengakses fisik perangkat untuk mengubah aplikasi, parameter, *firmware* atau memperbarui *software*[5].

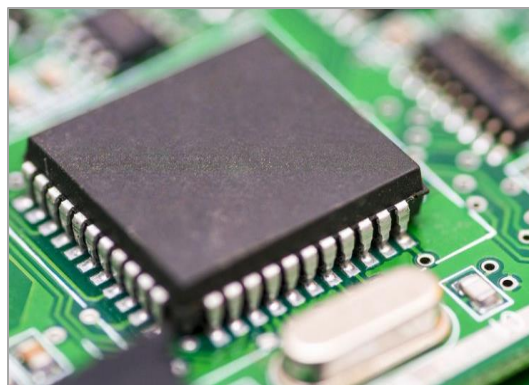


Gambar 3 Ilustrasi OTA

OTA pada proses pengujian ini digunakan sebagai proses untuk melakukan pembaruan *firmware*. Sehingga proses pembaruan dilakukan tanpa menggunakan kabel USB seperti sebelumnya.

4. *Firmware*

Firmware atau sistem tertanam merupakan sebuah sistem (rangkaiian elektronik) digital yang merupakan bagian dari sebuah sistem yang lebih besar, yang biasanya bukan berupasisistem elektronik. Contoh pengaplikasiannya adalah instrument medik, *automated Vehicles control* dan perangkat komunikasi. Sistem tertanam diimplementasikan melalui mikrokontroler[2]. *Firmware* memiliki fungsi untuk mengontrol fungsi dasar dari perangkat tersebut. Karakteristik dari *firmware* tertanam pada perangkat keras, tidak terlihat oleh pengguna dan dapat diperbarui. Dengan kata lain *firmware* merupakan instruksi bawaan pabrik yang tertanam secara permanen dalam suatu perangkat dan mengeksekusi instruksi inti perangkat.



Gambar 4 *Firmware*

Pada pengujian ini *firmware* digunakan sebagai landasan operassional yang akan diperbarui terutama pada tampilan versi pada *output*. Contoh yang dapat diperbarui pada *firmware* yaitu kesalahan pada versi sebelumnya, penambahan fitur pada perangkat. pada intinya *firmware* digunkana untuk merepresentasikan pengembangan lebih baru dari yang *firmware* lama.

5. ESP8266

ESP8266 merupakan pengatur utama pada perangkat yang memiliki orientasi yaitu mengendalikan sesuatu berdasarkan masukan (*input*) yang

diterima kemudian diproses oleh mikrokontroler setelah itu menghasilkan luaran (*output*) sesuai dengan program yang sudah dimasukan sebelumnya.

ESP8266 juga merupakan lingkungan pengembangan perangkat lunak dan perangkat keras sumber terbuka (*open source*) yang dibangun disekitar *System on Chip* (SoC). ESP8266 juga suatu modul turunan pengembangan dari modul *platform IoT (Internet of Things)* keluarga ESP8266 tipe ESP-12[8]. ESP8266 merupakan salah satu mikrokontroler yang dapat terhubung secara langsung menggunakan Wi-Fi.

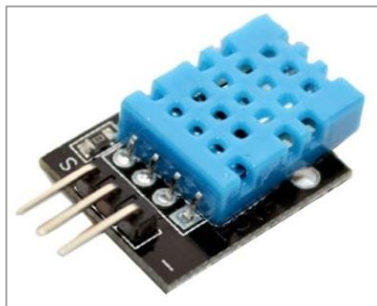


Gambar 5 ESP8266

Pada pengujian ini ESP8266 digunakan untuk mengatur keseluruhan perangkat yang terhubung pada pengujian ini. Dalam kata lain ESP8266 merupakan otak yang dapat mengatur melalui program yang dimasukkan ke dalamnya.

6. DHT11

DHT11 adalah singkatan dari *Digital Humadity and Temperatur*. DHT11 adalah salah satu jenis modul yang berfungsi untuk mengukur suhu dan kelembaban udara yang ada pada lingkungan sekitar[3], [8], [9], [10]. Modul ini merupakan pilihan sensor yang lazim digunakan pada pengembangan aplikasi *embedded* dan solusi IoT dengan kompleksitas yang rendah.



Gambar 6 DHT11

Pada pengujian ini, DHT11 diimplementasikan sebagai elemen untuk validasi fungsional sistem setelah pembaruan *firmware* dilakukan.

7. Arduino IDE

Arduino IDE adalah *software Integrated Development Environment (IDE)* yang digunakan untuk membuat kode program atau sebagai media untuk melakukan pemrograman pada *board* dan pengembangan. Arduino IDE biasanya juga digunakan untuk edit program, *upload* ke *board* yang ditentukan secara *offline*. Dalam Arduino memiliki beberapa library yang digunakan untuk mempermudah pada saat melakukan penulisan kode program untuk perangkat.

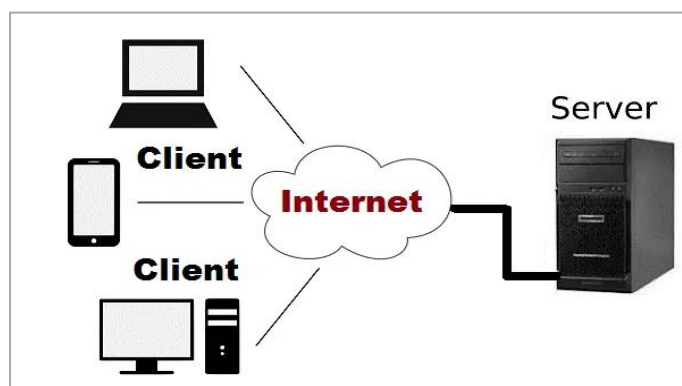


Gambar 7 Aplikasi Arduino IDE

Arduino IDE pada pengujian ini digunakan fasilitator utama pada proses pengembangan, *debugging*, dan validasi secara keseluruhan.

8. Server

Server adalah sebuah perangkat yang digunakan sebagai layanan penyedia perangkat lain atau yang sering disebut dengan klien[11]. Server juga merupakan media yang memiliki peranan penting digunakan untuk menyediakan layanan akses internet bagi *client*. Dalam penelitian ini server digunakan agar jaringan tetap dapat digunakan secara optimal.



Gambar 8 Server

9. Google Drive

Google Drive merupakan *software* yang efektif untuk melakukan penyimpanan, pengelolaan dan juga kolaborasi dokumen secara *online*. Google Drive juga dokumen yang ada dapat disinkronkan secara otomatis sehingga memungkinkan untuk diakses dari berbagai perangkat kapan saja. Pada penelitian ini Google Drive digunakan sebagai *platform* untuk membantu *developer* dalam proses *update firmware*.



Gambar 9 Google Drive

10. GitHub

GitHub merupakan sebuah *platform* berbasis web yang digunakan untuk menyimpan, mengelola dan mengontrol kode program secara kolaboratif. GitHub juga merupakan *platform* yang populer di kalangan *developer* karena didalamnya para *programmer* bisa saling sharing mengenai kode program[12]. Selain itu juga *programmer* dapat menyimpan data program dengan aman, sehingga dapat dikatakan GitHub ialah “media sosial” bagi para *programmer*.

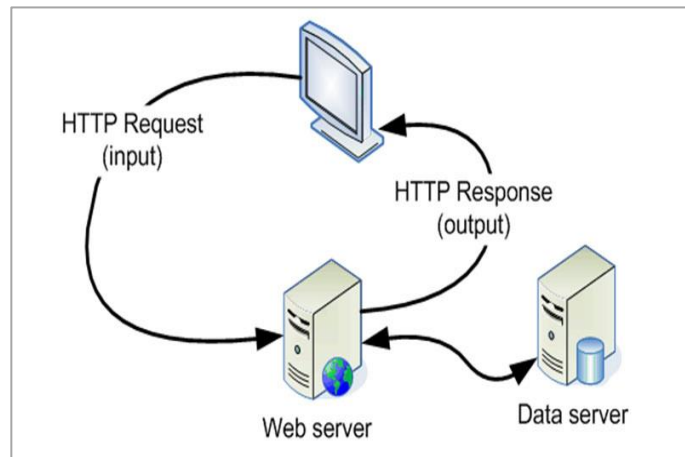


Gambar 10 GitHub

11. Web Server

Web Server merupakan layanan *online* yang menyediakan ruang penyimpanan di server untuk menyimpan semua data dari suatu website agar dapat dengan mudah untuk diakses siapapun di internet. Web Server dapat

berupa perangkat lunak atau perangkat keras yang memiliki fungsi untuk menerima permintaan dari klien biasanya dari browser web kemudian mengirim kembali respon berupa halaman web, file atau data lainnya melalui protokol HTTP atau HTTPS.



Gambar 11 Web Server