

# MONITORING SMART HOME MENGGUNAKAN RASPBERRY PI 3 B+

Wilsa Ananda Selbi<sup>1</sup>, Periyadi<sup>2</sup>, Devie Ryana Suchendra<sup>3</sup>

<sup>1,2,3</sup> Universitas Telkom, Bandung

wilziananda@student.telkomuniversity.ac.id<sup>1</sup>, periyadi@staff.telkomuniversity.ac.id<sup>2</sup>,

deviesuchendra@staff.telkomuniversity.ac.id<sup>3</sup>

---

## Abstrak

Seiring dengan berkembangnya teknologi yang semakin pesat setiap harinya tentu saja mempermudah setiap sarana yang diperlukan bagi kelangsungan hidup manusia. Teknologi semakin tidak melibatkan manusia dalam beberapa aktifitas tertentu. Salah satu teknologi tersebut adalah Monitoring Smart Home. Sistem Monitoring Smart Home adalah perangkat yang memiliki fungsi sebagai sistem kontrol terhadap lingkungan sekitar rumah. Sistem Monitoring ini menampilkan keadaan suhu, kelembapan udara, tekanan udara, kualitas udara dan hujan di sekitar rumah. Sistem monitoring ini dibuat dengan menggunakan JavaScript untuk pembuatan frontend, bahasa pemrograman PHP untuk pembuatan backend dan menggunakan Visual Studio Code sebagai text editor.

**Kata Kunci :** Sistem Informasi Raspberry Pi, Monitoring Smart Home, Monitoring, JavaScript, PHP, Visual Studio Code

---

## Abstract

*Along with development of technology that is growing rapidly everyday of course it makes every facility that we need more easier. Technology increasingly and not involving humans in certain activities. One of these technologies is Smart Home Monitoring. Smart Home Monitoring is a device that has a fuction as a contol system for the environment around the house. This monitoring system displays the state of temperature, humidity, air pressure, air quality and rain around the house. This monitoring system is made by using JavaScrips for frontend, PHP programming language for backend and Visual Studio Code as a text editor.*

**Keywords:** Information System of Raspberry Pi, Smart Home Monitoring, JavaScript, PHP, Visual Studio Code

---

## 1. Pendahuluan

### 1.1 Latar Belakang

Seiring dengan berkembangnya teknologi yang semakin pesat setiap harinya tentunya mempermudah setiap sarana yang diperlukan bagi kelangsungan hidup manusia. Teknologi tentu saja semakin tidak melibatkan manusia dalam beberapa aktifitas tertentu. Muncul sebuah permasalahan yaitu manusia sering kali merasa kerepotan dalam memantau keadaan sekitar rumah saat tidak berada di rumah. Untuk mempermudah user dalam pemantauan kondisi dirumah maka disinilah peran Monitoring Smart Home bekerja. Monitoring Smart Home ini berupa sistem yang dibuat berdasarkan data yang telah diolah hardware. Sistem ini menampilkan keadaan lingkungan sekitar rumah dan di dalam rumah yang berupa keadaan suhu, kelembapan udara, tekanan udara, kualitas udara dan hujan. Sistem ini mempunyai beberapa fitur yaitu, menampilkan suhu, kelembapan udara, kualitas udara, tekanan udara dan hujan.

menampilkan pesan singkat pada saat suhu dan kualitas udara berada dalam kondisi tertentu, menampilkan notifikasi untuk keadaan kipas dan humidifier berada dalam keadaan mati atau menyala, mengetahui data kondisi dalam bentuk grafik, dan menampilkan history dalam bentuk tabel untuk memudahkan user ketika ingin melihat history berdasarkan tanggal tertentu. Sistem pada Proyek Akhir ini dibangun untuk memudahkan aktifitas didalam rumah dalam membantu user untuk memantau keadaan sekitar rumah dan memudahkan beberapa aktifitas user. Seperti mengetahui keadaan kipas dan humidifier dalam keadaan mati atau menyala.

### 1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, rumusan masalah yang didapat adalah pada era saat ini *user* akan melakukan pengecekan suhu, kelembapan udara, tekanan udara, kualitas udara, dan hujan secara terpisah-pisah karena belum adanya layanan yang

dapat menampilkan informasi mengenai keadaan tersebut secara lengkap.

### 1.3 Tujuan

Adapun tujuan yang ingin dicapai dalam pembuatan laporan tugas akhir ini adalah:

1. Membuat website yang dapat menampilkan informasi suhu, kelembapan udara, tekanan udara, kualitas udara dan hujan.
2. Mampu menampilkan informasi mengenai history dalam bentuk tabel.
3. Mampu menampilkan informasi suhu, kelembapan udara dan tekanan udara dalam bentuk grafik.

### 1.4 Batasan Masalah

Adapun batasan masalah yang akan dilakukan yaitu sebagai berikut.

1. Output dari sistem ini dapat ditampilkan di smartphone hanya dengan menggunakan website.
2. Tampilan aplikasi tidak kompatible pada beberapa device karena aplikasi tidak mendukung UI yang responsiv.
3. Membutuhkan daya listrik untuk menampilkan output pada layar monitor.

## 2. Tinjauan Pustaka

### 2.1 Tinjauan Pustaka

Tinjauan Pustaka digunakan sebagai referensi untuk pembuatan proyek akhir ini.

Seiring dengan berkembangnya teknologi yang semakin pesat setiap harinya tentu saja mempermudah setiap sarana yang diperlukan bagi kelangsungan hidup manusia. Teknologi tentu saja semakin tidak melibatkan manusia dalam beberapa aktifitas tertentu[1]. Otomatisasi rumah atau biasa dikenal dengan sebutan smart home dapat dideskripsikan sebagai pengenalan teknologi dengan lingkungan rumah yang menyediakan kenyamanan, keamanan dan efisiensi energi bagi penghuni[2]. Rancang bangun Sistem Monitoring Smart Home berbasis pada data yang dikirimkan dari Raspberry Pi. Pada tampilan yang dibangun peneliti memanfaatkan database untuk melakukan komunikasi antara hardware dan software yang ada pada alat Smart Home yang dimana akan

terintegrasi langsung dengan aplikasi database[3]. Monitoring Smart Home menggunakan Raspberry Pi 3 B+ adalah sebuah sistem yang memiliki fungsi sebagai sistem kontrol dan monitoring terhadap lingkungan sekitar rumah. Sistem monitoring ini memanfaatkan sensor – sensor yang dikirimkan ke Raspberry Pi untuk diproses menjadi output pada layar monitor 29". Sistem ini dibuat berdasarkan data yang diolah hardware yang akan menampilkan keadaan lingkungan sekitar rumah yang berupa keadaan suhu, kelembapan udara, tekanan udara, kualitas udara dan hujan[4]. Sistem ini mempunyai beberapa fitur yaitu menampilkan history dalam bentuk tabel untuk memudahkan user ketika ingin melihat history berdasarkan tanggal tertentu, menampilkan pesan singkat pada saat suhu dan kualitas udara berada dalam kondisi tertentu, menampilkan notifikasi untuk keadaan kipas dan humidifier berada dalam keadaan mati atau menyala, mengetahui data kondisi dalam bentuk grafik, dan mengetahui keadaan sekitar yang sedang berlangsung menggunakan kamera Raspberry Pi. Sistem ini menggunakan JavaScript untuk pembuatan frontend, bahasa pemrograman PHP untuk pembuatan backend dan menggunakan Visual Studio Code sebagai text editor[5].

### 2.2 Dasar Teori

#### 2.2.1 Monitoring

Monitoring adalah pemantauan yang dapat dijelaskan sebagai kesadaran tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu[6].



Gambar 2. 1 Monitoring

#### 2.2.2 Raspberry PI 3

Raspberry Pi 3 adalah sebuah Single Board Computer yang merupakan generasi ketiga dari keluarga

Raspberry Pi. Raspberry Pi 3 memiliki RAM 16GB dan grafis Broadcom VideoCore IV pada frekuensi clock yang lebih tinggi dari sebelumnya yang berjalan pada 250MHz[7].



Gambar 2.2 Raspberry Pi 3

**2.2.3 Visual Studio Code**

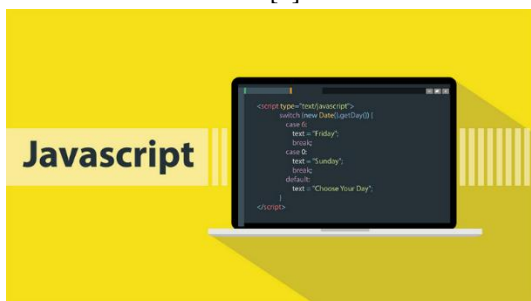
Visual Studio Code merupakan sebuah perangkat lunak yang digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasi lainnya dalam bentuk aplikasi console, aplikasi windows ataupun aplikasi web[8].



Gambar 2.3 Visual Studio Code

**2.2.4 JavaScript**

JavaScript adalah bahasa script yang bisa dijalankan di browser, dan biasa disebut dengan client side programming. Client disini adalah browser, seperti: Google Chrome, Internet Explorer, Firefox, Safari dan sebagainya. Kode javascript biasanya disisipkan diantara kode-kode HTML[9].



Gambar 2.4 JavaScript

**2.2.5 PHP**

PHP merupakan singkatan dari “Hypertext Preprocessor”, yang merupakan sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML. Sebagian besar sintaks dalam PHP mirip dengan bahasa C, Java dan Perl, namun pada PHP ada beberapa fungsi yang lebih spesifik. Sedangkan tujuan utama dari penggunaan bahasa ini adalah untuk memungkinkan perancang web yang dinamis dan dapat bekerja secara otomatis[10].



Gambar 2.5 PHP

**2.2.6 Back-end**

Backend merupakan tempat di mana proses pada suatu sistem informasi atau aplikasi berjalan, data dapat ditambahkan, diubah maupun dihapus. Backend biasanya mengurus segala jenis proses yang tidak berhubungan langsung dengan pengguna, seperti server dan basis data. Backend dibutuhkan dalam pengembangan sistem dan manajemen data pada sistem[11].

**2.2.7 Front-end**

Front end adalah pengembangan antarmuka pengguna grafis dari sebuah situs web, melalui penggunaan HTML, CSS dan JavaScript, sehingga pengguna dapat melihat dan berinteraksi dengan situs web tersebut[12].

**2.2.8 MySQL**

MySQL merupakan jenis program database server yang dapat menerima dan mengirim data dengan cepat menggunakan perintah – perintah SQL. MySQL mempunyai tiga sub bahasa, yaitu yang pertama Data Definition Language (DDL), DDL biasanya digunakan untuk membangun objek – objek dalam database, seperti table dan index. Bahasa yang kedua yaitu Data Manipulation Language (DML), DML ini digunakan untuk menambahkan, mencari, mengubah, dan menghapus baris dalam table. Dan yang terakhir adalah Data Control Language (DCL) yang digunakan untuk menangani security dalam database[13].



Gambar 2.6 MySQL

**2.2.9 XAMPP**

XAMPP adalah sebuah aplikasi web server instan dan lengkap dikarenakan segala yang anda butuhkan untuk membuat sebuah situs web dengan Content Management System (Joomla) bisa dicoba di dalam aplikasi ini. XAMPP adalah sebuah paket installer AMP (Apache, MySQL dan PHP) yang sangat mudah untuk diaplikasikan dalam komputer anda yang belum memiliki server untuk dapat melihat situs yang anda buat menggunakan bahasa server dan database server tersebut[14].



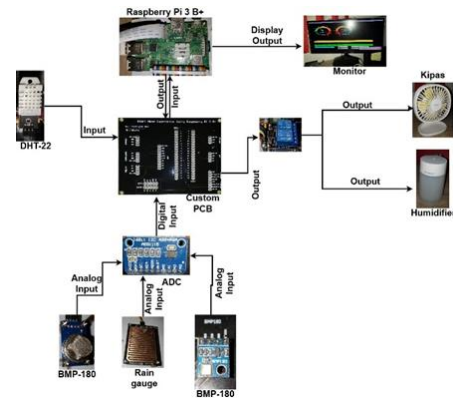
Gambar 2. 7 XAMPP

**3. Analisis dan perancangan**

**3.1 Gambaran Sistem Saat Ini (atau produk)**

Sistem monitoring cuaca di lingkungan rumah saat ini masih bersifat manual yang mana user harus melihat atau mengecek satu per satu pada berbagai layanan monitoring cuaca.

Saat user ingin mengetahui kondisi tertentu di sekitar rumah maka user akan membuka layanan yang dibuat untuk memeriksa keadaan tersebut, misalnya ingin mengetahui suhu maka user akan membuka layanan untuk mengetahui suhu, atau menggunakan alat pengukur suhu tetapi jika user ingin melihat kondisi lain seperti kelembapan udara, kualitas udara, tekanan udara dan hujan maka masih sangat sedikit layanan yang menampilkan monitoring keadaan tersebut.



Gambar 3. 1 Gambaran sistem saat ini

**3.2 Identifikasi Kebutuhan Sistem (atau Produk)**

Melihat kondisi atau gambaran sistem yang ada saat ini, maka diperlukan kebutuhan non fungsional dan fungsional sebagai syarat dibangunnya sistem pada Proyek Akhir ini. Adapun kebutuhan non fungsional yang diperlukan oleh sistem dalam Proyek Akhir ini.

Tabel 3. 1 Tabel Kebutuhan Non Fungsional

No.	Kebutuhan Non Fungsional
1	Raspberry Pi 3 B+
2	Layar Monitor 19"
3	Aplikasi pendukung seperti Visual Studio Code, Xampp, PHP,
4	MySQL untuk aplikasi web dan basis data sebagai media penyimpanan

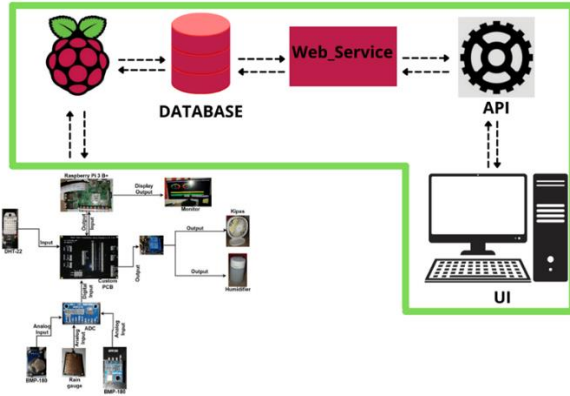
Untuk membangun sistem Proyek Akhir ini membutuhkan kebutuhan fungsional yaitu sebagai berikut.

Tabel 3. 2 Kebutuhan Fungsional

No.	Kebutuhan Fungsional
1	Membuat aplikasi monitoring
2	User dapat mengetahui keadaan cuaca secara keseluruhan
3	Membuat aplikasi yang dapat menampilkan history data kondisi cuaca dalam bentuk table
4	Membuat aplikasi yang dapat menampilkan grafik berupa perubahan kondisi cuaca pada hari itu.

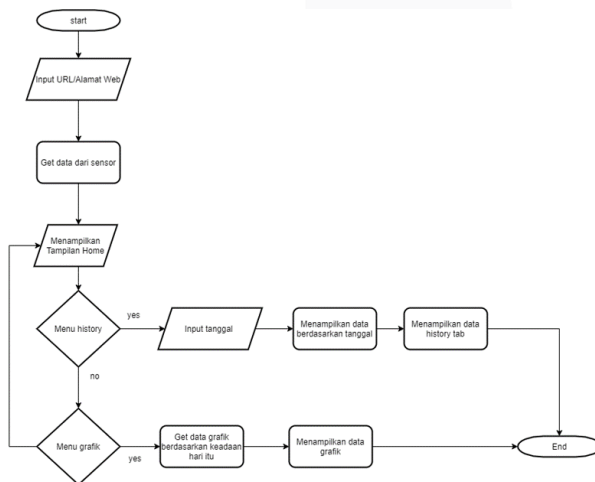
**3.3 Perancangan Sistem**

**3.3.1 Blok Diagram**



Gambar 3. 2 Blok Diagram Rancangan sistem yang akan dibangun yaitu membuat *User Interface* menggunakan HTML dan JavaScript untuk menghubungkan ke *service* dimana *service* tersebut mengakses ke *database* secara online yang mendapatkan data dari *hardware smart home* yang telah diolah oleh Raspberry Pi. Kelebihan dari sistem yang akan dibangun yaitu dapat mempermudah *user* dalam melakukan pemantauan kondisi cuaca di sekitar rumah menggunakan layar monitor 19”.

**3.3.2 Flowchart Sistem Alat**



Gambar 3. 3 Flowchart Sistem Alat

Pada Gambar 3.3 merupakan cara kerja *flowchart* sistem alat sebagai berikut :

1. Sistem alat ini dimulai dari mengakses halaman web yang tersedia.
2. Sistem akan menampilkan tampilan monitoring pada halaman home yang terdapat keterangan waktu, hari tanggal tahun, lokasi, keterangan suhu, *humadity*,

*rain, pressure, kualitas udara* dan notifikasi kondisi kipas dan *humidifier*.

3. Terdapat fitur *history* yang diakses dengan cara input tanggal awal dan tanggal akhir, setelah itu akan menampilkan data berdasarkan tanggal yang di input yang ditampilkan dalam bentuk tabel.
4. Pada fitur grafik ini akan menampilkan data berupa grafik yang menunjukkan kondisi cuaca pada hari itu.

**3.3.3 Metode Pengerjaan**

Metode pengerjaan Proyek Akhir ini menggunakan Metode Penelitian SDLC (*System Development Life Cycle*). Tahapan pengerjaian dilakukan sebagai berikut.

1. Analisa kebutuhan  
Pada tahap ini peneliti melakukan pencarian dan menganalisis kebutuhan user dalam melakukan pemantauan kondisi cuaca di rumah.
2. Desain sistem  
Pada tahap ini peneliti melakukan perancangan user interface, perancangan untuk basis data dan perancangan alur.
3. Implementasi dan pengujian unit  
Melakukan pengkodean atau coding untuk membuat aplikasi sistem monitoring ini menggunakan JavaScript untuk pembuatan front-end, pemrograman PHP untuk pembuatan back-end dan menggunakan Visual Studio Code sebagai text editor.
4. Integrasi dan pengujian aplikasi  
Melakukan pengujian mengenai fungsionalitas fitur pada aplikasi yang telah dibuat apakah sudah sesuai dengan kebutuhan yang telah dirancang.

**3.4 Kebutuhan Perangkat Keras dan Perangkat Lunak**

**3.4.1 Perangkat Keras**

Adapun spesifikasi perangkat keras dalam pengerjaan Proyek Akhir ini sebagai berikut :

No.	Perangkat Keras	Jumlah	Spesifikasi

1	Laptop	1 Buah	Processor Intel Core i5, RAM 8Gb
2	Layar Monitor 19"	1 Buah	Display Mode 1366 x 768 (32 bit) (59Hz)
3	Pendukung	1 Buah	Mouse, dan kabel HDMI to VGA
4	Raspberry Pi	1 Buah	3 B+

Tabel 3. 3 Perangkat keras

**3.4.2 Perangkat Lunak**

Adapun spesifikasi perangkat lunak dalam pengerjaan Proyek Akhir ini sebagai berikut :

No	Perangkat Lunak	Spesifikasi
1	Visual Studio Code	Versi 1.60.0
2	XAMPP	Versi 3.2.4
3	Browser seperti, Google Chrom	Versi 93.0.4577.63

Tabel 3. 4 Perangkat Lunak

**4. Implementasi dan Pengujian**

**4.1 Implementasi**

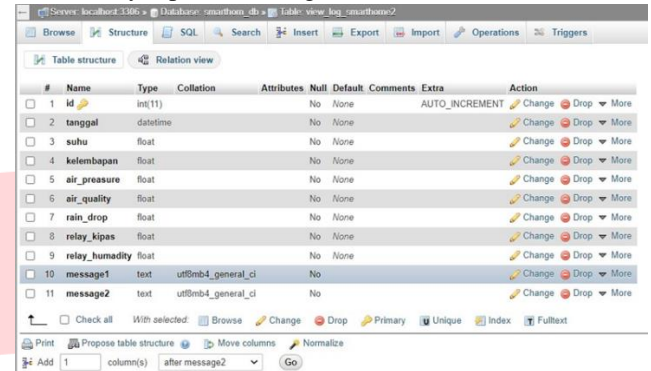
Pada implementasi proyek akhir ini dijelaskan tahap-tahap perancangan dan pembuatan aplikasi web. Berikut merupakan tahap-tahap implementasi yang dilakukan.

**4.1.1 Implementasi Database**

Berikut merupakan implementasi database untuk mendukung sistem Monitoring Smart Home menggunakan Raspberry Pi

1. Tabel View\_log\_smarthome2

Tabel 4-1 merupakan implementasi database tabel view\_log\_smarthome2 pada MySQL untuk menyimpan data dengan id tersebut.

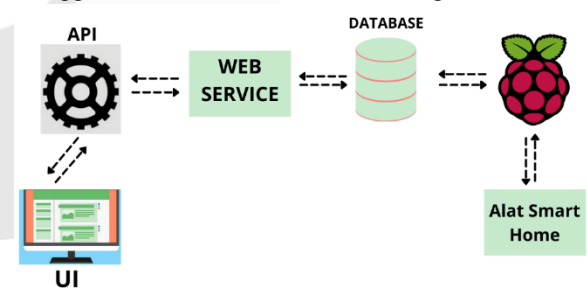


Gambar 4-1 View\_log\_smarthome2

**4.1.2 Pengiriman Data**

Dalam proses pengiriman data, alat smart home yang didalamnya terdapat sensor suhu, kelembapan udara, tekanan udara, kualitas udara dan sensor hujan akan mengirimkan data ke raspberry pi.

Setelah data berhasil di input ke raspberry pi maka selanjutnya adalah raspberry pi akan input data ke database lalu web service akan mengambil data dari database untuk dikirimkan ke API dan ditampilkan pada user interface. *User interface* ini dibangun dengan menggunakan JavaScript untuk pembuatan *front-end*, pemrograman PHP untuk pembuatan *back-end* dan menggunakan Visual Studio Code sebagai *text editor*.



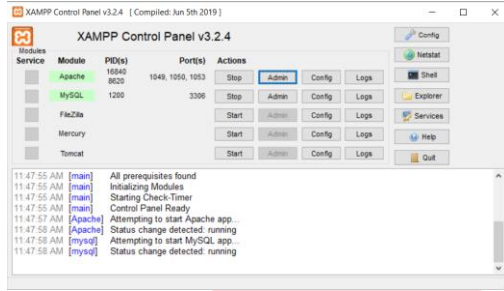
Gambar 4-2 Pengiriman Data

**4.1.3 Perancangan Aplikasi Monitoring**

Untuk melakukan pengkodean aplikasi web menggunakan Visual Studio Code dan memanfaatkan aplikasi XAMPP untuk membuat server lokal.

Membuat pengkodean untuk menampilkan kondisi suhu, kelembapan udara, tekanan udara, kualitas udara dan hujan pada tampilan monitoring untuk memudahkan *user* memantau keadaan sekitar.

Membuat pengkodean untuk menampilkan fitur *history* dan grafik menggunakan *library* untuk membuat tampilan yang menarik dan mudah dimengerti.



Gambar 4-3 XAMPP

```

211 <script type="text/javascript"> // logic
212 setInterval(function() {
213   $.ajax({
214     url: "web_service.php",
215     success: function(result){
216       var json = $.parseJSON(result);
217       var suhu = parseFloat(json['suhu']);
218       var kelembapan = parseFloat(json['kelembapan']);
219       var air_pressure = parseFloat(json['air_pressure']);
220       var air_quality = parseFloat(json['air_quality']);
221       var rain_drop = parseFloat(json['rain_drop']);
222       var relay_kipas = parseFloat(json['relay_kipas']);
223       var relay_humidity = parseFloat(json['relay_humidity']);
224       var relay_kipas_message = json['relay_kipas_message'];
225       var relay_humidity_message = json['relay_humidity_message'];
226       var message1 = 'Suhu udara terlalu dingin';
227       var message2 = 'Suhu udara normal';
228       var message3 = 'Suhu udara terlalu panas';
229       var udaraBaik = 'Kualitas udara baik';
230       var udaraTidakBaik = 'Kualitas udara tidak baik';

```

Gambar 4-4 Kodingan menampilkan keadaan cuaca

```

234 console.log('suhu : ',suhu)
235 if (suhu < 20) {
236   $("#message1").html(message1);
237 } else if (suhu < 30 && suhu > 20) {
238   $("#message2").html(message2);
239 } else {
240   $("#message3").html(message3);
241 }
242
243 if (air_quality > 0 && air_quality < 100){
244   $("#udaraBaik").html(udaraBaik);
245 } else {
246   $("#udaraTidakBaik").html(udaraTidakBaik);
247 }
248
249 if (relay_kipas == 1) {
250
251 }
252 $("#suhu").html(suhu+ "°C");
253 $("#humidity").html(kelembapan+ "%");
254 $("#rain_drop").html(rain_drop+ "%");
255 $("#air_pressure").html(air_pressure+ "kPa");
256 //document.getElementById("message1").innerHTML = message1;
257 //$("#air_pressure").html("Air Pressure : "+air_pressure);
258 $("#relay_kipas_message").html(relay_kipas_message);
259 $("#relay_humidity_message").html(relay_humidity_message);
260 }
261 });

```

Gambar 4-5 kodingan untuk menampilkan notifikasi kondisi suhu dan kualitas udara

```

111 <div class="table" style="background-color: #eee; text-align:center; padding-top:10px;">
112 <input type="text" id="tanggalAwal" data-date-format="yyyy-mm-dd">
113 <input type="text" id="tanggalAkhir" data-date-format="yyyy-mm-dd">
114 <button class="btn btn-primary" id="btnHistory" style="margin-top:10px;">Submit/Buttom</button>
115 </div>
116 <table id="table-history">
117 <thead>
118 <tr>
119 <th>Tanggal</th>
120 <th>Suhu</th>
121 <th>Kelembapan</th>
122 <th>Air Pressure</th>
123 <th>Air Quality</th>
124 <th>Rain Drop</th>
125 </thead>
126 <tbody>
127 <tr>
128 <td></td>
129 <td></td>
130 <td></td>
131 <td></td>
132 <td></td>
133 <td></td>
134 </tr>
135 </tbody>
136 </table>

```

Gambar 4-6 Kodingan tabel history

```

208 <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

```

Gambar 4-7 Library grafik

```

15 $sql = "SELECT * FROM view_log_smartHome2 ORDER BY tanggal DESC LIMIT 1";
16 $result = $conn->query($sql);

```

Gambar 4-8 Kodingan get database

## 4.2 Pengujian

Pengujian dilakukan untuk melihat perubahan data dari setiap kondisi cuaca untuk mengetahui keberhasilan dari fitur-fitur pada proyek akhir.

### 4.2.1. Pengujian Fitur Aplikasi

#### 1. Tujuan Pengujian :

Memastikan bahwa semua fitur aplikasi dapat digunakan dengan baik.

#### 2. Skenario Pengujian :

Dengan cara melakukan input tanggal pada fitur *history* agar bisa tau bahwa fitur dapat bekerja dengan semestinya, memantau keadaan suhu pada fitur grafik dan memantau apakah notifikasi pada tampilan home apakah bekerja dengan dengan semestinya.

#### 3. Hasil Pengujian

Hasil pengujian ditampilkan dalam bentuk tabel sebagai berikut :

No.	Fitur	Gambar	Keterangan
1	Menampilkan Suhu, Kelembapan, Hujan, Tekanan udara, Kualitas udara		Berhasil
2	History		Berhasil
3	Grafik		Berhasil

### 4. Analisis Hasil Pengujian

Pengujian	Hasil
Pengujian menampilkan suhu, kelembapan udara, hujan, tekanan udara dan kualitas udara	Berhasil
input tanggal untuk mengetahui history kondisi cuaca	Berhasil
Menampilkan grafik kondisi cuaca pada hari ini	Berhasil

### 4.2.2. Pengujian Kondisi Cuaca

#### 1. Tujuan Pengujian :

Mengetahui kondisi cuaca yang didapatkan apakah bekerja dengan baik.

2. Skenario Pengujian

Merekam kondisi setiap 3 jam dan merekam kondisi cuaca dengan durasi 10 menit

3. Hasil Pengujian

- Hasil Pengujian 3 jam Pertama selama 10 menit

Tanggal	Waktu	Kondisi				
		Suhu	Kelembapan Udara	Hujan	Tekanan Udara	Kualitas Udara
12-Sep-21	12:46:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:47:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:48:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:49:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:50:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:51:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:52:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:53:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:54:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:55:20	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	12:56:17	27.6 °C	95.50%	-59.36%	117 kPa	Baik

Gambar 4. 1 Hasil Pengujian 3 Jam Pertama

- Hasil pengujian 3 jam kedua selama 10 menit

Tanggal	Waktu	Kondisi				
		Suhu	Kelembapan Udara	Hujan	Tekanan Udara	Kualitas Udara
12-Sep-21	16:35:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:36:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:37:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:38:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:39:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:40:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:41:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:42:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:43:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik
12-Sep-21	16:44:12	27.6 °C	95.50%	-59.36%	117 kPa	Baik

Gambar 4. 2 Hasil Pengujian 3 Jam Kedua

- Hasil pengujian 3 jam ketiga

Tanggal	Waktu	Kondisi				
		Suhu	Kelembapan Udara	Hujan	Tekanan Udara	Kualitas Udara
13-Sep-21	1:26:27	27.5 °C	91.30%	-59.04%	117 kPa	Baik
13-Sep-21	1:27:27	27.5 °C	91.30%	-59.36%	117 kPa	Baik
13-Sep-21	1:28:28	27.5 °C	91.30%	-59.20%	117 kPa	Baik
13-Sep-21	1:29:27	27.4 °C	91.30%	-59.20%	117 kPa	Baik
13-Sep-21	1:30:27	27.5 °C	91.40%	-59.20%	117 kPa	Baik
13-Sep-21	1:31:27	27.5 °C	91.30%	-59.20%	117 kPa	Baik
13-Sep-21	1:32:27	27.5 °C	91.40%	-59.20%	117 kPa	Baik
13-Sep-21	1:33:27	27.5 °C	91.40%	-59.20%	117 kPa	Baik
13-Sep-21	1:34:27	27.4 °C	91.40%	-59.20%	117 kPa	Baik
13-Sep-21	1:35:27	27.5 °C	91.40%	-59.36%	117 kPa	Baik
13-Sep-21	1:36:27	27.5 °C	91.50%	-59.04%	117 kPa	Baik

Gambar 4. 3 Hasil Pengujian 3 Jam Ketiga

4. Analisis Hasil Pengujian

Waktu Pengujian	Hasil
3 jam pertama	Tidak ada perubahan selama 10 menit pada 3 jam pertama, kon cuaca stabil
3 jam kedua	Tidak ada perubahan selama 10 menit pada 3 jam kedua, kon cuaca stabil
3 jam ketiga	Terjadi perubahan suhu pada menit 1:29:27, 1:30:27, 1:34:27. Se itu juga terjadi perubahan kondisi kelembapan udara pada m 1:30:27, 1:31:27, 1:32:27, 1:36:27. Perubahan pada Hujan pada m 1:27:27, 1:28:28, 1:35:27, 1:36:27.

Gambar 4.4 Analisis Hasil Pengujian Kondisi Cuaca

1. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian pada proyek akhir ini, dapat ditarik kesimpulan sebagai berikut.

1. Aplikasi monitoring smart home dapat menampilkan kondisi cuaca berupa suhu, kelembapan udara, hujan, tekanan udara dan kualitas udara secara praktis pada satu layar monitor, namun terdapat delay pada sensor hujan selama 5 detik.
2. User dapat melihat history data keadaan cuaca berdasarkan tanggal yang dipilih dan akan ditampilkan dalam bentuk tabel pada aplikasi ini yang di input dari 40 data terbaru dari database dengan jarak waktu 20 detik.
3. User dapat memantau kondisi suhu, air pressure dan kelembapan udara dalam bentuk grafik yang dapat dipilih dengan klik pada bagian yang tidak ingin ditampilkan.

5.2 Saran

Berdasarkan hasil dari penelitian dan pengujian alat pada Proyek Akhir ini, terdapat beberapa saran yang dapat dijadikan pengembangan selanjutnya, yaitu :

1. Pada ukuran tampilan aplikasi buat agar dapat responsive terhadap perbagai ukuran layar monitor.
2. Tambahkan sensor gerak pada layar monitor untuk menambahkan fungsi agar saat ada gerakan pada jarak tertentu maka layar monitor bisa mati dan menyala guna menghemat pemakaian enegi.



## Referensi

- [1] kemenkes R. 2020, "Bab 1 pendahuluan," *Pelayanan Kesehat.*, no. 2015, pp. 3–13, 2010.
- [2] R. Piyare and R. L. Seong, "Smart Home-Control and Monitoring System Using Smart Phone," *Proceedings, 1st Int. Conf. Converg. it's Appl.*, vol. 24, pp. 83–86, 2013.
- [3] S. N. Ishak, N. N. N. A. Malik, N. M. A. Latiff, N. E. Ghazali, and M. A. Baharudin, "Smart home garden irrigation system using Raspberry Pi," in *2017 IEEE 13th Malaysia International Conference on Communications (MICC)*, 2017, pp. 101–106, doi: 10.1109/MICC.2017.8311741.
- [4] J. C. Co-, *About the authors*, vol. 27, no. 3. 2002.
- [5] V. Patchava, H. B. Kandala, and P. R. Babu, "A Smart Home Automation technique with Raspberry Pi using IoT," in *2015 International Conference on Smart Sensors and Systems (IC-SSS)*, 2015, pp. 1–4, doi: 10.1109/SMARTSENS.2015.7873584.
- [6] S. Santoso, D. A. Rahman, and D. E. Purnama, "Aplikasi Sistem Monitoring Keadaan Komputer Di Laboratorium Pada Perguruan Tinggi Raharja," *ICIT J.*, vol. 2, no. 2, pp. 184–195, 2016, doi: 10.33050/icit.v2i2.33.
- [7] A. A. Meidyan Putri, "Rancang Bangun Sistem Smart Class Berbasis Web," pp. 4–29, 2011, doi: 10.1360/zd-2013-43-6-1064.
- [8] D. Umagapi and S. Hasan, "Sistem Informasi Pengendalian Internal Prosedur Pencatatan Akuntansi Penggajian Pada Pt. Halmahera Karya Timur Persada Menggunakan Visual Studio," *IJIS - Indones. J. Inf. Syst.*, vol. 3, no. 2, pp. 76–84, 2018, doi: 10.36549/ijis.v3i2.45.
- [9] I. Explorer and P. Web, "Modul 5 - Javascript-," p. 1.
- [10] My PHP.net, "Hypertext Preprocessor (PHP)," *PHP Gr.*, pp. 4–16, 2021.
- [11] R. Pangestika and R. T. Dirgahayu, "Pengembangan Back-end Sistem Informasi Komunitas Pendar Foundation Yogyakarta," *Automata*, 2020.
- [12] D. A. N. R. Ternak, "St Ay St Ay," no. September, p. 2011, 2010.
- [13] C. K. Bandung, B. Irawan, S. Si, A. B. Osmond, and S. T. Mt, "Aplikasi sistem informasi pada puskesmas," pp. 1–16.
- [14] yolana dan mansuri, "Sistem Informasi Pariwisata Propinsi Nangroe Aceh Darussalam Berbasis Web," *Jupiter*, vol. 1, pp. 32–39, 2015.