

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Penelitian ini berhasil mengimplementasikan sistem IoT hidroponik yang dapat di monitoring pada sebuah halaman pemantauan berbasis web. Sistem ini dibangun menggunakan bahasa pemrograman C untuk kode program di ESP32 dan bahasa pemrograman *javascript* di *framework ReactJS*. Data sensor disimpan dalam database untuk ditampilkan di halaman web. Database yang digunakan adalah *firebase*.

Sistem ini mempunyai fitur monitoring dan kontrol. Terdapat grafik yang menampilkan data dari sensor-sensor yang di aplikasikan ke media hidroponik. Juga terdapat indikator berupa lampu dan *circle progress bar* dan kontrol manual berupa *selector switch*.

4.2 IMPLEMENTASI DESAIN INTERFACE

Pada halaman web disajikan tiga *selector switch* dan satu *setting time* di sebelah kiri. Di bagian kanan atas terdapat indikator berupa dua lampu dan dua *circle progress bar* atau indikator persentase sensor. Dan terdapat grafik yang menampilkan data sensor di beberapa waktu. Berikut kode pemrograman untuk implementasi halaman monitoring:

1. Kode pemrograman untuk menampilkan selector switch auto/manual,

```
<h2>Auto - Manual</h2>
<ToggleSwitch
  key={writeData["test/Switch1"]}
  active={writeData["test/Switch1"]}
  onClick={() => handleToggleSwitch("test/Switch1")}
/>
```

2. Kode pemrograman untuk menampilkan selector switch pompa sirkulasi,

```
<h2>Saklar Pompa Sirkulasi</h2>
<ToggleSwitch
  key={writeData["test/Switch2"]}
  active={writeData["test/Switch2"]}
  onClick={() => handleToggleSwitch("test/Switch2")}
/>
```

3. Kode pemrograman untuk menampilkan selector switch pompa semprot,

```
<h2>Saklar Pompa Semprot</h2>
<ToggleSwitch
  key={writeData["test/Switch3"]}
  active={writeData["test/Switch3"]}
  onClick={() => handleToggleSwitch("test/Switch3")}
/>
```

4. Kode pemrograman untuk menampilkan setting waktu,

```
<h2>Setting Time (Menit)</h2>
<input
  type="number"
  value={inputNumber}
  onChange={handleInputNumberChange}
/>
```

5. Kode pemrograman untuk menampilkan indikator motor pompa,

```
<div className="grid-item2">
  <h2>Indikator Pompa Sirkulasi</h2>
  <LedIndicator switchState={led1 ? "ON" : "OFF"} />
</div>
<div className="grid-item2">
  <h2>Indikator Pompa Semprot</h2>
  <LedIndicator switchState={led2 ? "ON" : "OFF"} />
</div>
```

6. Kode pemrograman untuk menampilkan indikator persentase,

```
<div className="grid-item2">
  <h2>Indikator Sensor TDS</h2>
  <CircleProgressBar percentage={percentage1} circleWidth="200"/>
</div>
<div className="grid-item2">
  <h2>Indikator Sensor Water Level</h2>
  <CircleProgressBar percentage={percentage2} circleWidth="200"/>
</div>
```

7. Kode pemrograman untuk menampilkan grafik data sensor,

```
<div className="dataCard revenueCard">
  <Line
    data={{
      labels: sensorData.map((data) =>
        new Date(data.timestamp).toLocaleTimeString()
      ),
      datasets: [
        {
          label: "Sensor TDS",
          data: sensorData.map((data) => data.sensor1),
          fill: true,
          backgroundColor: "rgba(6, 79, 240, 0.2)",
          borderColor: "rgba(6, 79, 240, 1)",
          borderWidth: 1,
        },
        {
          label: "Sensor Water Level",
          data: sensorData.map((data) => data.sensor2),
```

```

    fill: true,
    backgroundColor: "rgba(255, 48, 48, 0.2)",
    borderColor: "rgba(255, 48, 48, 1)",
    borderWidth: 1,
  },
  />
</div>

```

Dari kode program diatas sehingga dapat di tampilkan sebuah halaman web monitoring seperti pada Gambar 4.1.



Gambar 4.1 Tampilan halaman web.

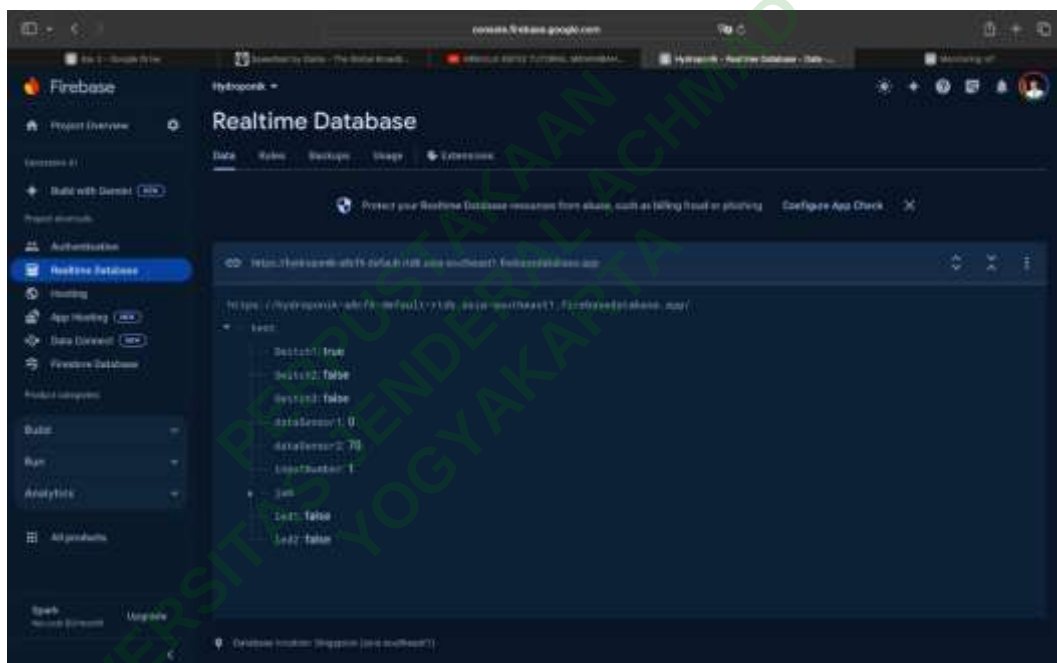
Dari Gambar 4.1 fungsi *toggle switch* pada program adalah untuk menampilkan *selector switch* pada halaman monitoring web. Kemudian *setting time* pada kode program berfungsi untuk mengatur waktu jeda pada mode otomatis output motor semprot. Selanjutnya *led indicator switch state* berfungsi sebagai indikator yang dibuat untuk menandakan motor pompa hidup/mati. *Circle progress bar* pada kode pemograman berfungsi sebagai indikator persentase hasil dari sensor yang tampil pada halaman web monitoring. Yang terakhir ada fungsi *chart* yang digunakan untuk menampilkan grafik pada halaman web monitoring. Fungsi *line* pada kode program diatas adalah untuk membuat garis yang terhubung antar data.

4.3 IMPLEMENTASI DATABASE

Dalam penelitian ini, sistem *database* yang digunakan yaitu *firebase*. *Firebase* merupakan salah satu *database NoSQL* yang di-hosting di *cloud*. Sistem

IoT hidroponik ini mengimplementasikan *database* nilai sensor yang di tampilkan pada grafik dan indikator. Untuk kode konfigurasi *framework ReactJs* dan *firebase* terhubung adalah sebagai berikut:

```
const firebaseConfig = {
  apiKey: "AIzaSyBDch4oNkNn5SA_dHI5japSMypDyxRGW5Q",
  authDomain: "hydroponik-a8cf6.firebaseio.com",
  databaseURL: "https://hydroponik-a8cf6-default-rtdb.asia-southeast1.firebaseio.com",
  projectId: "hydroponik-a8cf6",
  storageBucket: "hydroponik-a8cf6.appspot.com",
  messagingSenderId: "44975630903",
  appId: "1:44975630903:web:98998d36b967a51f5e2db2",
};
```



Gambar 4.2 Contoh *database* nilai sensor.

Firebase mempunyai jenis *realtime database* dan *cloud firestore*. Pada Gambar 4.2. menjelaskan bahwa *firebase* yang digunakan adalah *realtime database*. Alasan menggunakan *realtime database* adalah penggunaannya yang lebih mudah. Cara kerja *realtime database* yaitu data disimpan sebagai JSON dan disinkronkan secara *realtime* lintas platform.

4.4 HASIL PENGUJIAN

Setelah sistem selesai dibangun, perlu dilakukan pengujian fitur-fitur yang ada untuk mengetahui fungsionalitasnya.

1. Pengujian pada tombol selector switch auto. Fitur otomatis dimulai dari selector switch auto, saat fitur otomatis berjalan maka kontrol manual tidak dapat difungsikan. Yang berfungsi pada mode auto adalah sensor LDR dan timer. Apabila sensor LDR mendeteksi cahaya kurang atau menandakan gelap maka secara otomatis motor pompa sirkulasi mati. Akan Kembali hidup setelah sensor LDR mendeteksi cahaya yang cukup. Secara sederhana prinsip LDR adalah semakin banyak cahaya, nilai resistansi sensor semakin kecil. Nilai resistensi LDR dapat mencapai 1 Megaohm ($1.000.000\Omega$) saat kondisi sangat gelap. Pada sistem yang di buat sensor LDR yang di gunakan jenis analog. Sensor jenis tersebut telah dikoversi dengan sistem modul Arduino mempunyai nilai 0-1023. Kemudian nilai tersebut dikonversi ke persen. Pengujian sensor LDR dapat di lihat pada Tabel 4.1

Tabel 4.1 Tabel hasil pengujian fungsi sensor LDR

Waktu	Nilai sensor	Kondisi pompa sirkulasi
Siang	10%	Hidup
Sore	50%	Hidup
Malam	$\geq 60\%$	Mati

Skema pengujian pada sensor LDR seperti Tabel 4.1. Pada siang hari nilai sensor LDR sebesar 10% sehingga fungsi yang digunakan kondisi motor pompa sirkulasi hidup. Saat malam hari nilai sensor LDR lebih dari 60% sehingga motor pompa sirkulasi mati. Dari skema pengujian tersebut dapat disimpulkan bahwa fungsi sensor LDR sudah sesuai.

Kemudian untuk timer berfungsi untuk mengontrol jalannya motor pompa semprot. Apabila timer belum memberikan sinyal maka motor semprot tidak bisa hidup. Dan akan hidup dikondisi sebaliknya. Fungsi dari fitur otomatis ini dapat berjalan dengan baik.

2. Selector switch pompa sirkulasi akan berfungsi jika mode auto mati. Tombol tersebut dapat mengontrol hidup/matinya motor pompa sirkulasi sesuai aksi pengguna.
3. Selector switch pompa semprot sama sama seperti selector switch pompa sirkulasi. Kedua switch tersebut tidak akan berfungsi saat selector switch mode auto menyala. Pengujian switch dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian Swicth manual

Selector switch motor sirkulasi	Selector switch motor sirkulasi	Motor pompa sirkulasi	Motor pompa semprot
off	off	off	off
on	off	on	off
off	on	off	on
on	on	on	on

Skema pengujian yang di lakukan untuk menguji switch manual adalah dengan pengujian respon output terhadap inputan yang berbeda. Contoh seperti Tabel 4.2 pengujian yang dilakukan adalah ketika *switch* sirkulasi hidup maka outputnya juga hidup. Kemudian saat *switch* semprot hidup maka outputnya juga hidup. Saat kedua *switch* dimatikan outputnya juga mati. Dari skema pengujian tersebut dapat disimpulkan bahwa *switch* manual sudah sesuai.

4. Pada sistem ini, timer berfungsi sebagai kontrol otomatis pompa semprot. Timer akan berfungsi saat mode auto hidup. Cara kerja timer yaitu apabila diatur 1 menit maka motor pompa semprot akan hidup selama 10 detik lalu mati. 1 menit setelah motor mati, maka motor akan nyala dan mati sesuai waktu yang di atur.
5. Indikator dan grafik mempunyai fungsi masing-masing. Terdapat indikator lamp dan indikator persentase. Indikator lamp berfungsi menandai apakah motor pompa hidup/mati. Apabila indikator lamp berwarna hijau tandanya motor pompa hidup dan jika berwarna merah tandanya motor posisi mati.

Indikator persentase dan grafik fungsinya menampilkan nilai yang dibaca sensor. Perbedaan indikator persentasi update setiap detik sedangkan grafik update 1 menit sekali. Dapat dilihat pada Tabel 4.3 untuk perbedaan indikator persentase dan grafik.

Tabel 4.3 Perbedaan indikator persentase dan grafik

Durasi pengujian	Waktu	Nilai Indikator persentase	Nilai grafik
1 Jam	15.15.10	45%	45%
	15.15.15	45%	45%
	15.15.55	43%	45%
	16.16.00	44%	45%
	16.16.10	42%	42%
	16.16.20	40%	42%

Skema yang dilakukan untuk pengujian nilai indikator persentase dan grafik adalah dengan pengambilan data berdasarkan waktu permenit dengan respon nilai persentase yang dikeluarkan di indikator maupun grafik. Contohnya seperti pada Tabel. 4.3 bahwa pada jam 15.15.10 nilai indikator dan grafik sama 45% kemudian pada jam 15.15.55 nilai indikator menunjukkan 43% dan nilai grafik tetap 45%. Setelah jam 15.16.10 atau satu menit setelah nilai indikator dan grafik sama pada nilai keduanya juga sama 42%. Dapat disimpulkan fungsional indikator dan grafik sudah sesuai.

Untuk hasil pengujian fungsionalitas fitur secara garis besarnya dapat dilihat pada Tabel 4.4.

Tabel 4.4 Tabel hasil pengujian fungsional

No	Fungsi	Skenario	Hasil	Keterangan
1	<i>Selector Auto</i>	Setelah dipilih auto maka pompa akan bekerja sesuai sensor dan <i>setting time</i>	Pompa sirkulasi mati apabila sensor LDR mendeteksi kurang cahaya. Pompa semprot akan nyala pada waktu yang sudah ditentukan	Berrhasil
2	<i>Selector pompa sirkulasi</i>	Saat mode auto <i>off</i> dan <i>selector</i> pompa sirkulasi <i>on</i> maka pompa sirkulasi <i>on</i> dan indikator pompa sirkulasi <i>on</i>	Pompa sirkulasi dikendalikan manual	Berhasil
3	<i>Selector pompa semprot</i>	Saat mode auto <i>off</i> dan <i>selector</i> pompa semprot <i>on</i> maka pompa semprot <i>on</i> dan indikator pompa semprot <i>on</i>	Pompa semprot dikendalikan manual	Berhasil
4	<i>Setting time</i>	Saat mode auto <i>on</i> maka pompa semprot akan <i>on</i> sesuai settingan waktu	Pompa semprot otomatis	Berhasil
5	Indikator dan Grafik	Menampilkan nilai sensor	Nilai sensor tampil	Berhasil

4.5 PEMBAHASAN

Pada pengembangan sistem IoT hidroponik dengan metode *prototyping* mempunyai kelebihan diantaranya dapat langsung komunikasi dengan calon

pengguna sistem sesuai fungsionalitas yang dibutuhkan. Dengan sistem pengguna dapat melakukan monitoring media tanam hidroponik khususnya nutrisi dan air juga dapat melakukan kontrol manual maupun otomatis terhadap beberapa penanganan dalam perawatan tanaman hidroponik. Sistem ini dapat melakukan kontrol otomatis menggunakan sensor dan timer juga dapat melakukan kontrol manual sehingga pengguna dapat mengatur fungsi secara manual dengan *selector switch* yang ada di layar monitor. Selain itu, sistem ini menampilkan indikator dari nutrisi dan volume air pada tampungan hidroponik. Pengaplikasiannya dipasangkan sensor tds dan water level pada tampungan air hidroponik yang fungsinya dapat mengambil data sehingga data tersebut di tampilkan dalam wujud indikator dan grafik *realtime*. Juga terdapat sensor LDR yang fungsinya sebagai alat kontrol otomatis untuk motor pompa sirkulasi hidroponik. Di Bagasi Hidroponik sirkulasi air nutrisi pada pipa media tanam hidroponik hanya di lakukan saat matahari terbit sampai terbenam. Untuk pengujian sistem IoT hidroponik ini, penulis menggunakan metode Blackbox Testing. Secara rencana yang dibutuhkan pengguna, fungsionalitas sistem ini berhasil sesuai rencana.