

## **BAB II DASAR TEORI**

### **2.1 KAJIAN PUSTAKA**

Penelitian M. Nasir, M. Ali dan Nurhadi tahun 2010 yang berjudul “Perancangan Sistem *Monitoring Volume* Bahan Bakar pada *Prototype Sephull Bubble Vessel*” yang membahas tentang perancangan sistem *monitoring* volume bahan bakar berbasis IoT. Penelitian ini bertujuan untuk efisiensi pemakaian bahan bakar dengan kecepatan maksimal dengan menggunakan teknologi sistem pelumasan udara. Sistem *monitoring* bahan bakar secara otomatis dan bisa diamati secara *real time*. Pada perancangan ini akan menggunakan sensor *Universal Fuel Sender*, sensor ini berbasis *resistance* dengan menggunakan sistem pelampung dimana ketinggian level bahan bakar sebanding dengan *output resistensi* sensor. Namun hasil data *monitoring* yang dilakukan hanya ditunjukkan melalui *display* LCD belum melalui *software* atau *web* yang dapat digunakan sebagai proses penampilan hasil *monitoring*. Hasil dari kalibrasi sensor tersebut menunjukkan bahwa sensor mempunyai sensitivitas 30,2 *mVolt/cm*. Dari hasil tersebut dapat diketahui bahwa sistem berkerja dengan baik dan dengan *error* sekitar 3,3% [3].

Penelitian Riki Ruli, dan Rifky Raymond pada tahun 2015 membahas tentang “Model Sistem *Monitoring* Tangki Bahan Bakar Minyak SPBU dengan Menggunakan *Web* Aplikasi dan *SMS Gateway*”. Tujuan dari sistem ini dibuat untuk mengembangkan sistem untuk memantau bahan bakar minyak pada SPBU secara telemetri dengan memanfaatkan mikrokontroler *Arduino* berbasis *web* yang terintegrasi dengan *sms gateway*. Karena pada sistem *monitoring* yang dilakukan sebelumnya pihak SPBU masih menggunakan sistem secara manual dengan menggunakan galah, sehingga menyebabkan ketidak efektifan dan efisiennya proses *monitoring*. Pengujian sistem dilakukan dalam dua tahapan yaitu pengujian tampilan halaman *web* yang digunakan oleh pengelola SPBU pada *SPBU Client* dan pengujian *monitoring* dan pengukuran pada tangki penda SPBU. Namun pada perancangan ini sesnsor suhu yang digunakan tidak menggunakan sensor anti air (cairan). Hasil peneitian ini dihasilkan dengan interval jarak yang digunakan yaitu 5 cm, tingkat rata-rata yaitu 3,02% [4].

Penelitian Yuda Indra dan Muchlas pada tahun 2020 yang meneliti tentang “Simulasi Sistem Telementri Pemantauan Level Ketinggian BBM Berbasis SMS Gateway” Sistem ini dibuat untuk mengetahui level ketinggian BBM pada tangki pendam SPBU dengan menggunakan teknologi SMS sebagai proses *monitoring*nya. Sistem ini diharapkan untuk mempermudah *monitoring* dengan kebutuhan pemantauan dari jarak jauh. Sistem ini dibuat menggunakan sensor ultrasonik sebagai sensor jarak yang digunakan untuk membaca ketinggian BBM dan *Arduino Uno* sebagai mikrokontrolernya. Penelitian ini masih belum menggunakan sensor suhu sebagai sistem *monitoring* di dalam tangki pendam tersebut, sehingga masih menjadi kekurangan dalam penelitian ini. Namun dari hasil pengujian sistem, setelah dilakukan percobaan sebanyak 6 kali dinyatakan sensor ultrasonik mampu bekerja dengan baik. Hasil akurasi sensor 100%, hasil tersebut diambil dari perbandingan pembacaan sensor dan hasil ukur secara manual menggunakan penggaris [5].

Penelitian Dian Yayan, Ferabuta, Ery dan Chairil Idrus pada tahun 2019 yang meneliti tentang “Teknologi Informasi dan Otomatisasi Pengukuran Ketinggian BBM dalam Tangki Minyak Berbasis *Arduino Uno*”. Sistem ini dibuat untuk mengembangkan teknologi informasi dan pengukuran otomatis ketinggian BBM dalam tangki minyak berbasis *Arduino Uno*. Cara kerja sistem ini menggunakan sensor Ultrasonik PING. Sinyal pengukuran dalam bentuk digital diolah dan diproses menggunakan mikrokontroler *Arduino Uno* menghasilkan sinyal informasi. Sinyal informasi yang dihasilkan langsung ditampilkan menggunakan LCD. Pada kondisi tertentu yang dianggap informasi penting, sinyal informasi diteruskan GSM untuk dikirimkan ke telepon selular pemilik melalui media pesan singkat (SMS). Hasil pengukuran tinggi BBM secara otomatis memiliki akurasi 100% hingga ketinggian 25 cm. Namun di atas 25 cm akurasi ketinggian BBM memiliki rata-rata sebesar 97,9% [6].

Penelitian Cyntia, Putri Insani, dan M. Diono pada tahun 2019 yang meneliti tentang “Sistem *Monitoring* Tangki dan Perhitungan *RunHour* Genset Otomatis Berbasis *Internet of Things*”. Sistem ini bertujuan untuk melakukan *monitoring* bahan bakar minyak pada genset secara *realtime* pada saat genset digunakan. Sistem tersebut terdiri dari sensor ultrasonik sebagai sensor pembaca BBM genset,

sensor *servo* sebagai pengaturan kran pada genset dan menggunakan NodeMCU ESP8266 sebagai mikrokontrolernya. Data yang dihasilkan akan ditampilkan pada personal *computer* saja. Sistem ini belum menggunakan aplikasi *Android* dalam proses *monitoring*nya. Dari pengujian yang dilakukan presentase *error* kinerja sensor *ultrasonic* sebesar 1,9% dalam satuan *centimetre* (jarak) dan 0,034 dalam satuan liter (volume) [7].

## **2.2 BAHAN BAKAR MINYAK (BBM)**

Bahan Bakar Minyak (BBM) merupakan salah satu komoditas dari sumber daya alam minyak dan gas bumi. Minyak dan gas bumi merupakan sumber daya alam strategis tidak terbarukan yang dikuasai oleh Negara. Minyak dan gas bumi merupakan komoditas vital yang menguasai ajat hidup orang banyak. Minyak dan gas bumi mempunyai peranan penting dalam perekonomian nasional. Bahan Bakar Minyak (BBM) merupakan bahan bakar yang berasal dan atau diolah dari minyak bumi. Minyak bumi adalah hasil proses alami berupa hidrokarbon yang dalam kondisi tekanan dan *temperature* berupa fasa cairan atau padat [8].

## **2.3 STASIUN PENGISIAN BAHAN BAKAR UMUM (SPBU)**

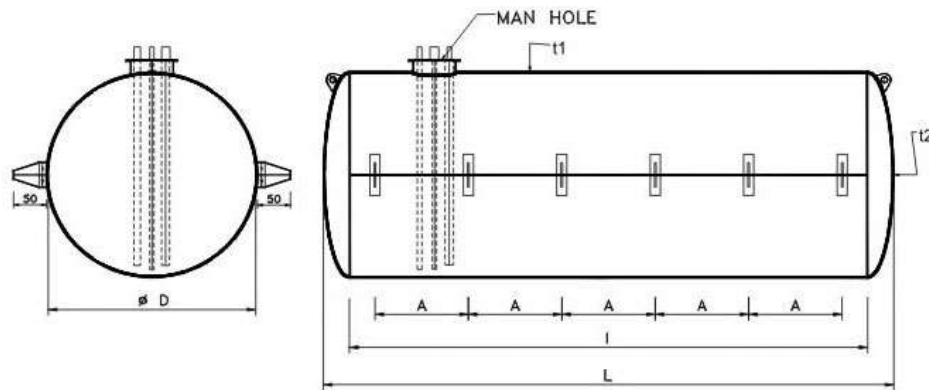
Stasiun Pengisian Bahan Bakar merupakan tempat dimana kendaraan bermotor dapat memperoleh bahan bakar. Di Indonesia, Stasiun Pengisian Bahan Bakar dikenal dengan nama SPBU, namun masyarakat juga memiliki sebutan bagi SPBU, misalnya saja di kebanyakan daerah SPBU disebut juga sebagai Pom Bensin. Stasiun Pengisian Bahan Bakar, pada umumnya menyediakan beberapa jenis bahan bakar sebagai berikut:

- a. Bensin dan beragam varian produk bensin.
- b. Solar.
- c. LPH dalam berbagai ukuran tabung.
- d. Minyak tanah.

Dalam beberapa Negara termasuk Indonesia, Stasiun Pengisian Bahan Bakar dijaga oleh petugas-petugas yang mengisikan bahan bakar kepada pelanggan. Pelanggan kemudian membayarkan biaya pengisian kepada petugas. Namun di Negara lainnya, misalnya saja Amerika atau Eropa, pompa-pompa bensin tidak dijaga oleh petugas, pelanggan mengisikan bahan bakar sendiri kemudian membayarkannya kepada petugas di sebuah loket. Di samping itu pada setiap Stasiun Pengisian

Bahan Bakar tentunya memiliki tempat penyimpanan bahan bakar seperti bensin dan solar, bahan bakar tersebut disimpan pada sebuah tangki pendam atau *underground tank* yang berada pada setiap SPBU.

Tangki pendam merupakan suatu tempat penyimpanan bahan bakar yang ada disetiap SPBU yang umumnya berupa bak yang berada dibawah permukaan tanah. Pengukuran bahan bakar yang dilakukan saat ini kurang efisien, hal ini dikarenakan pengukuran kapasitas bahan bakar yang ada di dalam tangki pendam SPBU dilakukan secara manual. Pengukuran dengan menggunakan sensor merupakan salah satu alternatif yang dapat digunakan dalam proses pengukuran kapasitas di dalam tangki pendam tersebut [2].



Gambar 2.1 *Underground Tank* [2]

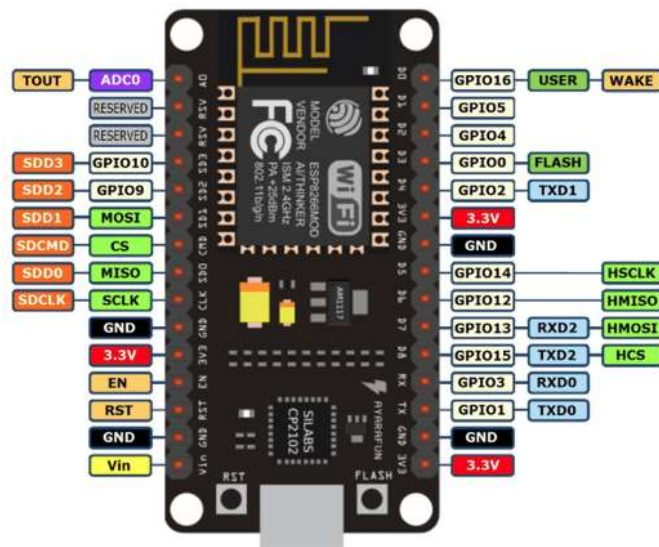
Tabel 2.1 Tipe Tangki Pendam SPBU [9]

NO	TYPE TANGKI	$\phi$ (DIAMETER)	l mm	L mm	t1 mm	t2 mm	A mm
1	10.000 LITER	2070	2540	3040	6	8	970
2	10.000 LITER	2380	2275	2775	6	8	825
3	15.000 LITER	2000	4750	5210	6	8	1030
4	20.000 LITER	2500	4000	4540	8	10	1130
5	30.000 LITER	2500	6100	6700	8	10	1110
6	45.000 LITER	2500	9150	9950	8	10	1200
7	45.000 LITER	2750	7500	8138	8	10	1230
8	60.000 LITER	2860	9450	10050	8	10	1230

## 2.4 NODEMCU ESP8266

NodeMCU adalah sebuah *board* modul *embedded system* yang mempunyai *feature WiFi*, yang menggunakan *chip ESP8266* dengan *firmware* berbasis *Lua*. Pada NodeMCU ini dilengkapi dengan *port Micro USB* yang berfungsi untuk

melakukan pemrograman sekaligus *power supply*. Karena banyak pihak seperti pelajar, *enginerrm developer* lebih familiar dengan bahasa C dan *Arduino*, maka pengguna maupun pengembang ESP8266 melakukan *porting board* supaya dapat berjalan dan dapat diprogram dengan menggunakan *Arduino IDE*. Spesifikasi Modul NodeMCU ESP8266 sebagai berikut [10].



Gambar 2.2 NodeMCU ESP8266 [10]

Tabel 2.2 Spesifikasi Modul NodeMCU ESP8266 [10]

Spesifikasi	NodeMCU
Mikrokontroler	ESP8266-12E
Tegangan Input	3.3 ~ 5V
GPIO	13 Pin
Kanal PWM	10 Kanal
10 bit ADC Pin	1 Pin
Flash Memory	4 MB
Clock Speed	40/26/24 MHz
WiFi	IEEE 802.11 b/g/n
Frekuensi	2.4 GHz – 22.5 GHz
USB Port	Micro USB
USB Chip	CH340G

## 2.5 SENSOR ULTRASONIK HC-SR04

Sensor ultrasonik secara umum dapat digunakan sebagai suatu pengungkapan tak sentuh yang beragam seperti aplikasi pengukuran jarak. Alat ini secara umum juga memancarkan gelombang suara ultrasonik menuju suatu target yang memantulkan balik gelombang kearah sensor. Kemudian sistem mengukur waktu yang diperlukan untuk pemancaran gelombang sampai kembali ke sensor dan menghitung jarak target dengan menggunakan kecepatan suara dalam medium. Rangkaian penyusun sensor ultrasonik ini terdiri dari *transmitter*, *receiver*, dan komparator. Sensor ultrasonik ini juga ada beberapa jenis, namun untuk perancangan alat kali ini akan menggunakan sensor ultrasonik jenis *Ultrasonic Ranging Module HC-SR04*. Sinyal yang dipancarkan sensor *ultrasonic* yang berupa gelombang bunyi dengan kecepatan 340 m/s. Sinyal yang dipantulkan dan diterima kembali oleh sensor akan dihitung dengan rumus sebagai berikut :

$$S = \frac{340\left(\frac{100}{1000000}\right).t}{2}$$
$$S = \frac{0,034 . t}{2} \quad [11]$$

Dimana S merupakan jarak sensor dengan objek pantul dan t merupakan selisih waktu pada saat gelombang dipancarkan dan diterima. Mikrokontroler dapat bekerja pada *microsecond* (1s =1.000.000  $\mu$ s) dan satuan jarak bisa diubah dari meter ke *centimetre* (1m =100 cm).



Gambar 2.3 Sensor Ultrasonik HC-SR04 [11]

Sedangkan untuk data set sensor *Ultrasonic Ranging Module HC-SR04* ini dapat dilihat pada tabel berikut [11].

Tabel 2.3 *Male Connector Side* [11].

<b>Pin No.</b>	<b>Signal</b>
1.	VCC (5V supply)
2.	Trigger Pulse Input
3.	Echo Pulse Output
4.	GND (0V)

Tabel 2.4 *Electric Spesification* [11]

<b>Parameter</b>	<b>Spesification</b>
VCC	4.8V to 5V (typical) 5.5V (absolute maximum)
<i>Operating Current</i>	15mA
<i>Temperature</i>	0° to 60° Ambient room temperature
<i>Trigger Input Pulse</i>	10uS TTL pulse (5V, ± 2%)
<i>Echo Input Pulse</i>	TTL (0V/5V, ± 2%)

Tabel 2.5 *Physical Spesification* [11]

<b>Parameter</b>	<b>Spesification</b>
<i>Dimensions</i>	45 x 20 x 15 mm
<i>Pin-out Pitch</i>	2.54mm male berg
<i>Interface</i>	VCC, GND, SDA, SCL

## 2.6 SENSOR SUHU DS18B20

DS18B20 merupakan sensor suhu digital dari *Maxim IC*. Sensor ini mampu membaca suhu dengan ketelitian 9 hingga 12-bit, rentang -55 C hingga 125 C dengan ketelitian (+/-0.5C). Setiap sensor yang diproduksi pasti memiliki sebuah kode unik sebesar 64 bit yan di sematkan pada masing-masing *chip*, sehingga memungkinkan penggunaan sensor dalam jumlah besar hanya melalui satu kabel saja (*single wire data bus/1 wire-protocol*). Hal tersebut merupakan komponen yang luar biasa, dan merupakan patokan dari banyak proyek-proyek data *logging* dan kontrol berbasis *temperature* di luar sana.



Gambar 2.4 Sensor suhu DS18B20 [12].

Fitur dari sensor suhu DS18B20, yaitu :

- a. Antarmuka hanya menggunakan satu kabel sebagai komunikasi (menggunakan *protocol Unique 1-Wire*).
- b. Setiap sensor memiliki kode pengenalan unik *64-bit* yang tertanam di *onboard ROM*.
- c. Kemampuan *multidrop* yang menyederhanakan aplikasi penginderaan suhu terdistribusi.
- d. Tidak memerlukan komponen tambahan.
- e. Juga bisa diumpungkan daya melalui jalur datanya. Rentang dayanya adalah 3.0V hingga 5.5V.
- f. Bisa mengukur temperatur mulai dari -55C hingga +125 C
- g. Memiliki akurasi +/-0.5C pada rentang -10C hingga +85C
- h. Resolusi sensor bisa dipilih mulai dari 9 hingga 12 bit.
- i. Bisa mengkonversi data suhu ke 12 bit digital *word* hanya dalam 750 milidetik.
- j. Memiliki konfigurasi *alarm* yang bisa disetel (*nonvolatile*) [12].

## 2.7 INTERNET OF THINGS (IOT)

*Internet of Thing* (IoT) merupakan suatu konsep yang bertujuan untuk memperluas manfaat dari konektivitas *internet* yang terhubung secara terus menerus yang memungkinkan kita untuk menghubungkan mesin, peralatan dan benda fisik lainnya dengan sensor jaringan dan *actuator* untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperbolehkan secara independen. *Internet of Things* ini juga merupakan sebuah gagasan dimana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan



internet sebagai penghubungnya. Pada dasarnya perangkat IoT terdiri dari sensor sebagai media pengumpulan data, sambungan *internet* sebagai media komunikasi dan *server* sebagai pengumpulan informasi yang diterima sensor dan untuk analisa. Banyak yang memprediksi bahwa pengaruh *Internet of Things* ini adalah “*the next big thing*” di dunia teknologi informasi, hal ini karena IoT menawarkan banyak potensi yang bisa digali dan dapat bermanfaat untuk segala kebutuhan manusia [13].

## 2.8 GOOGLE FIREBASE

*Firebase* merupakan suatu layanan dari *Google* untuk memberikan kemudahan bahkan mempermudah para *developer* aplikasi dalam mengembangkan aplikasinya. *Firebase* atau *BaaS (Backend as a Service)* adalah solusi yang ditawarkan *Google* untuk mempercepat pekerjaan *developer*. Dengan menggunakan *Firebase* ini, *apps developer* bisa fokus dalam mengembangkan aplikasi tanpa memberikan *effort* yang besar untuk urusan *backend*. *Firebase* pertama kali didirikan pada tahun 2011, produk *firebase* yang pertama kali adalah *Realtime Database*. *Realtime Database* tersebut digunakan *developer* untuk menyimpan data yang *synchronize* ke banyak user. Kemudian kini berkembang sebagai layanan pengembang aplikasi. Mengenai segi layanan, dulu *Firebase* ini memeberikan *service trial* (percobaan), namun saat ini dapat dimanfaatkan dan digunakan layanan ini secara gratis. Tentu saja dengan adanya batasan-batasan tertentu. Layanan yang tersedia pada *Firebase* ada 2 pilihan, yaitu :

- a. SPARK
- b. BLAZE [14].



Gambar 2.5 *Firebase Realtime Database*

## 2.9 APP INVENTOR

*App Inventor* merupakan sistem perangkat lunak untuk membuat aplikasi pada perangkat *Android*. *App Inventor* dibuat tidak seperti sistem pengembangan aplikasi pada biasanya, di mana seorang *programmer* harus menuliskan baris-baris kode program, melainkan dengan interaksi visual berbasis grafis. Dalam hal ini *App Inventor* dapat disebut sebagai sistem terpadu untuk mengembangkan aplikasi berbasis *blog-blog* grafis. *App Inventor* bekerja untuk *platform* perangkat bergerak pada yang berbasis *Android*. Istilah *App Inventor* dan *inventor* digunakan sebagai sinonim. *Inventor* diuji coba pertama kali pada kalangan terbatas juli 2010, kemudian dirilis ke *public* pada desember 2010. Pengembangan *inventor* dimodifikasi oleh keyakinan dan persepektif edukasi yang kuat bahwa pembelajaran aktif pemrograman (secara visual) dapat menjadi wadah untuk memicu ide-ide baru dan kreatif [15].



Gambar 2.6 *App Inventor*

## 2.10 QUALITY OF SERVICE

*Quality of Service* (QoS) merupakan suatu kemampuan jaringan untuk menyediakan layanan yang baik dengan menyediakan *bandwidth*, mengatasi *jitter* dan *delay*. Parameter QoS antara lain yaitu *throughput*, *delay*, *jitter*, dan *packet loss*. QoS ini sangat ditentukan oleh kualitas jaringan yang digunakan. Terdapat juga beberapa faktor yang mampu mempengaruhi nilai dari QoS, seperti redaman, *distorsi* dan *noise*.

### 2.10.1 Parameter *Quality of Service*

#### a. *Throughput*

*Throughput* adalah kecepatan dari transfer data. *Throughput* merupakan jumlah total kedatangan paket yang sukses diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut.

Tabel 2.6 Standarisasi *Throughput* [16]

Kategori <i>Throughput</i>	<i>Throughput</i> (bps)	Indeks
Sangat Bagus	100	4
Bagus	75	3
Sedang	50	2
Jelek	>25	1

Persamaan Perhitungan *Throughput* :

$$\text{Throughput} = \frac{\text{Jumlah Data yang Dikirim}}{\text{Waktu Pengiriman Data}} \times 8 \quad [16]$$

#### b. *Delay*

*Delay* merupakan waktu yang dibutuhkan pada sebuah data yang menempuh jarak dari asal ke tujuannya. *Delay* mampu dipengaruhi oleh jarak media fisik, kongesti atau proses yang lama.

Tabel 2.7 Standarisasi *Delay* [16]

Kategori <i>Delay</i>	Besar <i>Delay</i>	Indeks
Sangat Bagus	<150 ms	4
Bagus	150 s/d 300 ms	3
Sedang	300 s/d 450 ms	2
Jelek	>450 ms	1

Persamaan Perhitungan *Delay* :

$$\text{Delay (s)} = \frac{\text{Total Delay}}{\text{Total Paket yang Diterima}} \quad [16]$$

#### c. *Pakectloss*

*Packet loss* merupakan banyaknya paket yang gagal mencapai tempat tujuan paket tersebut dikirimkan [16].

Persmaan Perhitungan *Packetloss*:

$$\text{Packetloss} = \frac{\text{Paket Data Dikirim} - \text{Paket Data Diterima}}{\text{Paket Data yang Dikirim}} \times 100\% \quad [16]$$

Tabel 2.8 Standarisasi *Packetloss* [16]

<b>Kategori Packetloss</b>	<b>Packetloss</b>	<b>Indeks</b>
Sangat Bagus	0%	4
Bagus	3%	3
Sedang	15%	2
Jelek	25%	1