

# Pendeteksi Volume Air pada Galon Berbasis *Internet of Things* dengan Menggunakan Arduino dan Android

Denny Kuriando<sup>1</sup>, Agustinus Noertjahyana<sup>2</sup>, Resmana Lim<sup>3</sup>

<sup>1,2</sup> Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

<sup>3</sup> Program Studi Teknik Elektro Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-Mail: dennykuriando@yahoo.com<sup>1</sup>, agust@petra.ac.id<sup>2</sup>, resmana@petra.ac.id<sup>3</sup>

## ABSTRAK

*Internet of things* memiliki konsep yang bertujuan untuk memperluas manfaat yang tersambung dalam koneksi internet secara terus-menerus. Metode yang digunakan oleh *Internet of Things* adalah pengendalian secara otomatis tanpa mengenal jarak. Implementasi Saat ini pengontrolan galon air masih dilakukan dengan melihat keadaan galon kemudian pergantian galon apabila air pada galon sudah habis. Dengan adanya *internet of things* dapat di implementasikan pendeteksi volume air yang dapat memantau keadaan galon dari jarak jauh melalui *smartphone*. Untuk melakukan pembuatan pendeteksi volume air diperlukan alat seperti *water flow* sensor, mikrokontroler atau alat bantu lainnya yang kemudian dihubungkan dengan internet.

Untuk hubungan interaksi *water flow* sensor, mikrokontroler, ataupun alat bantu lainnya melalui internet digunakan aplikasi Android. Aplikasi Android ini digunakan untuk mempermudah pengguna untuk mengakses internet dimanapun. Aplikasi Android ini dibuat dengan menggunakan pemrograman Android Studio dan mikrokontroler dibuat dengan menggunakan bahasa C. Mikrokontroler yang digunakan adalah WeMos D1 R2 yang telah terdapat ESP8266 didalamnya.

Hasil akhir dai pengembangan pendeteksi volume air ini adalah mikrokontroler dapat mengirimkan data sensor ke server Ubidots sehingga aplikasi dapat mengambil data sensor di server Ubidots. Pengguna dapat melihat keadaan galon secara real time dan juga diberikan notifikasi apabila volume air galon berada dibawah alert level. Pengguna juga dapat melihat laporan pemakaian galon per bulan.

**Kata Kunci:** *Internet of things*, Pendeteksi volume air, Mikrokontroler ESP8266, Ubidots, *Water flow* sensor, Android.

## ABSTRACT

*Internet of things* has a concept that aims to expand the benefits connected in a continuous internet connection on an ongoing basis. The method used by *Internet of Things* is automatic control without distance. Implementation currently gallon control of water is still done by looking at the state of gallons and then gallons turn when the water in the gallon is already empty. With the *internet of things* can be implemented a water volume detector that can monitor the state of the gallon from a distance through a *smartphone*. To make the manufacture of water volume detection required tools such as *water flow* sensors, microcontroller or other tools that are then connected to the internet. For interaction relationship of *water flow* sensor, microcontroller, or other tool of help through internet

*used Android application. This Android application is used to facilitate users to access the internet anywhere. This Android application is made by using Android studio and microcontroller made with Arduino programming using C language. Microcontroller used is WeMos D1 R2 which has been contained ESP8266 in it.*

*The result of this water volume detection development is the microcontroller can transmit sensor data to the Ubidots server so that the application can retrieve sensor data on the Ubidots server. Users can view the gallon state in real time and are also given a notification when the gallon water volume is below alert level. Users can also view reports of gallons per month.*

**Keywords:** *Internet of things, Water volume detector, ESP8266 Microcontroller, Ubidots, Water flow sensor, Android.*

## 1. PENDAHULUAN

*Internet of things* memiliki konsep yang bertujuan untuk memperluas manfaat yang tersambung dalam koneksi internet secara terus-menerus. Metode yang digunakan oleh *Internet of Things* adalah pengendalian secara otomatis tanpa mengenal jarak. Pengimplementasian *Internet of Things* biasanya mengikuti keinginan *developer* dalam mengembangkan sebuah aplikasi yang dibuatnya.

Kebutuhan air minum sangatlah dibutuhkan oleh manusia. Tanpa air minum pastilah badan terasa lemah, letih, lesu, kurang bersemangat, dan mudah sekali capek walaupun tidak melakukan aktifitas yang berat. Dengan perkembangan jaman, masyarakat sekarang banyak yang beralih ke penggunaan galon dan dispenser sebagai tempat penyimpanan dan pengambilan air minum. Mengisi galon merupakan kegiatan yang harus dilakukan apabila isi air galon telah habis. Hal ini tak lepas dari kebiasaan masyarakat yang harus melihat dahulu habis atau tidaknya isi air galon.

Konsep *internet of things* dapat di implementasikan dalam pemantauan isi air galon dari jarak jauh dan memberikan notifikasi pada aplikasi Android melalui internet. Dalam menerapkan *internet of things* ini, alat sensor yang digunakan untuk mengukur volume air yang keluar ini harus dapat berkomunikasi dengan *smartphone* Android. Saat ini hanya implementasi pengukur volume air yang keluar sudah diterapkan pada dalam artikel [2]. Tetapi untuk teknologi pendekteksi volume air yang dapat dipantau melalui *smartphone* masih belum diterapkan dalam kehidupan sehari-hari sehingga tidak ada pemberitahuan mengenai habisnya air galon. Dengan adanya referensi dari artikel [2], maka dibuatnya pendeteksi volume air galon berbasis Arduino yang dapat

diaplikasikan dengan *smartphone* Android yang dapat memantau *volume* air tiap galon dari jarak jauh dan memberikan notifikasi ketika *volume* air galon lebih rendah dari batas minimal. Hal ini bisa dikatakan sebagai *prototype* di masa yang akan datang, teknologi ini diharapkan akan berguna untuk perusahaan – perusahaan yang memiliki jumlah pegawai yang besar seperti pabrik, rumah sakit, perkantoran, dan lain - lain.

## 2. LANDASAN TEORI

### 2.1. Internet of Things

*Internet of Things* (IOT) adalah konsep dan paradigma yang menganggap meresap kehadiran di lingkungan berbagai hal / benda yang melalui koneksi nirkabel dan kabel dan unik skema pengalamatan dapat berinteraksi satu sama lain dan bekerja sama dengan hal-hal lain / benda untuk membuat aplikasi baru / jasa dan mencapai tujuan bersama. Dalam konteks ini penelitian dan pengembangan tantangan untuk menciptakan dunia yang cerdas sangat besar. Sebuah dunia di mana nyata, digital dan virtual berkumpul untuk menciptakan lingkungan pintar yang membuat energi, transportasi, kota dan banyak daerah lain yang lebih cerdas [5].

### 2.2. Ubidots

Ubidots adalah sebuah IoT *cloud* atau *platform internet of things* yang berasal dari Boston, Amerika Serikat. Platform ini bertujuan untuk mempermudah pembuatan dalam pengambilan data dari sensor dan mengubahnya menjadi Informasi. Ubidots digunakan juga untuk mengatur aksi yang dilakukan atau memberikan peringatan berdasarkan data yang didapat. Ubidots menyediakan API untuk membantu proses pengumpulan data dan menjadikannya sebagai informasi. API adalah sebuah cara atau solusi agar benda dapat berinteraksi dengan *web*. API ini memungkinkan para pembuat untuk melakukan perubahan dan pengambilan data dari server. API ini mendukung penggunaan *HTTP* maupun *HTTPS*. Untuk melakukan pertukaran data diperlukan API *key*. Untuk mengambil sebuah nilai atau variabel dari server maka diperlukan *ID* dari variabel tersebut dan *token* akun *user* yang telah diberikan server. Untuk memudahkan *developer* Android, Ubidots telah menyediakan *library* yang berfungsi untuk mempersingkat penulisan *HTTP request* dalam bahasa *JAVA* [4].

### 2.3. Modul $G\frac{1}{2}$ Water Flow Sensor

Sensor aliran banyak digunakan untuk pemantauan dan pengendalian aplikasi, untuk mengukur aliran berupa udara atau cairan. *Water flow* sensor ini terdiri atas katup plastic, komponen rotor aliran air, dan sebuah sensor *half-effect*. Rotor pin memiliki magnet kecil yang terpasang, dan ada ruang-efek sensor magnetik di sisi lain dari tabung plastic yang dapat mengukur berapa banyak putaran kincir yang dilakukan melalui dinding plastic. Metode ini memungkinkan sensor untuk tetap aman dan kering [2].

Dalam sistem ini dapat menghitung aliran rotor yang dikelilingi oleh magnet bersama dengan penggunaan sensor *Hall Effect*. Hal ini dikenal sebagai  $G\frac{1}{2}$  *water flow* sensor. Turbin yang berputar pada medan magnet yang dihasilkan dan hasil dari AC *pulse* akan diubah menjadi *output digital* dengan bantuan sensor *Hall Effect* yang ditempatkan setelah turbin. Jumlah pulsa yang dihasilkan per liter dapat dihitung dengan perangkat lunak pemrograman. Jadi pulsa akan menghasilkan *output* frekuensi yang berbanding lurus dengan laju aliran *volumetric* / total laju aliran melalui meter

## 2.4. Mikrokontroler dan WiFi module ESP8266

Mikrokontroler adalah sebuah mikroprosesor yang memiliki bagian-bagian tambahan yang memungkinkan untuk mengatur atau mengontrol benda lain. Dapat diartikan bahwa mikrokontroler menjalankan program yang telah dibuat oleh *user* dan disimpan dalam *memory*. Mikrokontroler merupakan alat yang sangat berguna yang dapat membantu perancang untuk memanipulasi data yang didapat dari input ataupun *output*. Biasanya mikrokontroler tersusun atas mikroprosesor, *memory* dan I/O. *Microprocessor* sendiri tersusun atas *Central Processing Unit* (CPU) dan *Control Unit* (CU). CPU merupakan otak utama yang melakukan proses aritmatik dan logic. Sementara CU mengontrol operasi dari mikroprosesor dan mengirimkan sinyal ke bagian lain dari mikroprosesor untuk melakukan instruksi [3].

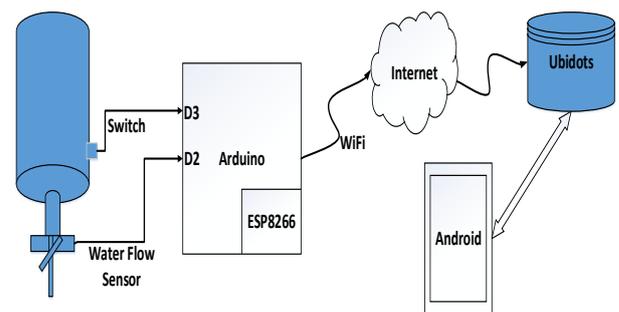
## 2.5. Arduino IDE

Arduino IDE adalah sebuah software open-source yang membantu pembuatan script atau code dan mengupload script atau code tersebut pada board atau *microprocessor*. Arduino IDE memiliki *environment* yang ditulis dalam *Java*. Arduino IDE juga memiliki compiler untuk bahasa C atau C++. Arduino IDE memang bertujuan untuk membantu pembuatan script atau code untuk berbagai macam board atau *microprocessor* [1].

## 3. ANALISIS DAN DESAIN SISTEM

### 3.1. Desain Perangkat Pendeteksi Volume Air

Pada gambar 1 merupakan blok diagram dari perangkat pendeteksi *volume* air galon dimana menjelaskan alur kerja sistem secara keseluruhan. Dalam pendeteksi *volume* air ini memiliki 3 komponen utama yang memiliki fungsinya masing – masing. Komponen utama tersebut adalah mikrokontroler WeMos D1 R2, *water flow* sensor, dan *switch*. Adapun komponen tambahan yang ada di desain adalah kabel jumper dan resistor.

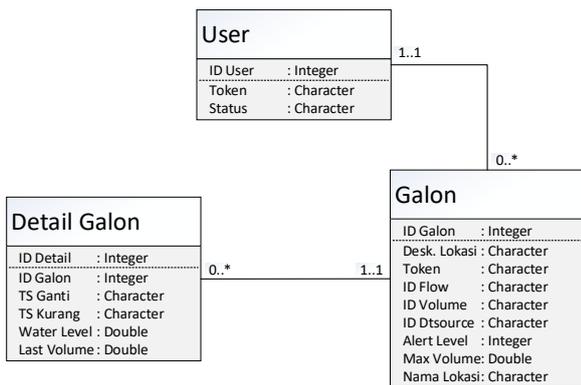


Gambar 1. Block Diagram Sistem

### 3.2. Class Diagram

Terdapat *class diagram* yang dijelaskan pada bab ini yaitu *class diagram* untuk SQLite. Pada gambar 2 ditampilkan *class diagram* untuk SQLite. Terdapat 3 *class* yaitu *user*, galon dan detail galon. Dimana pada *class user* terdapat atribut *id user*, token dan status. Sedangkan pada *class galon* terdapat atribut *id galon*, token, nama lokasi, deskripsi lokasi, *id flow*, *id volume*, *id datasource* dan *alert level*. Sedangkan pada *class detail galon* terdapat atribut *id detail galon*, *id galon*, *timestamp* ganti, *timestamp* kurang, *water level* yang merupakan total aliran air yang keluar dan *last volume* yang merupakan total *volume* air terakhir. *Class user* berelasi dengan *class galon*

yaitu relasi *one to many*. *Class* galon berelasi dengan *class* detail galon yaitu relasi *one to many*.



Gambar 2. Class Diagram

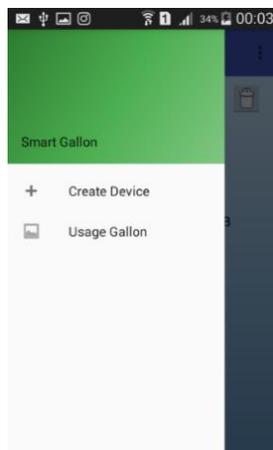
## 4. PENGUJIAN SISTEM

### 4.1. View Halaman Utama

Pada saat pengguna aplikasi berhasil melakukan *login* maka aplikasi akan menampilkan halaman *home* seperti pada gambar 3. Halaman ini akan menampilkan semua data galon di *database* SQLite yang memperlihatkan kondisi volume air galon berupa gambar. Terdapat 2 tombol yaitu tombol *delete* untuk menghapus *device* galon dan tombol “+” untuk *insert device* galon. Dan juga terdapat *menu bar*.



Gambar 3. Tampilan View Home

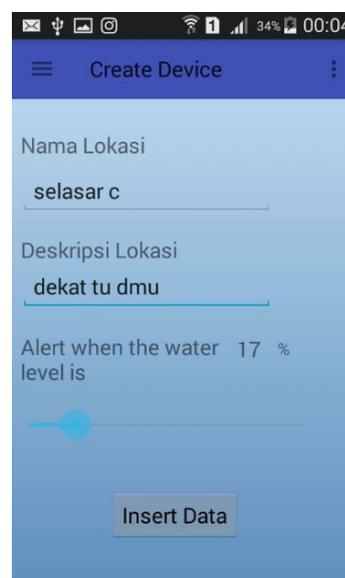


Gambar 4. Tampilan Menu Bar

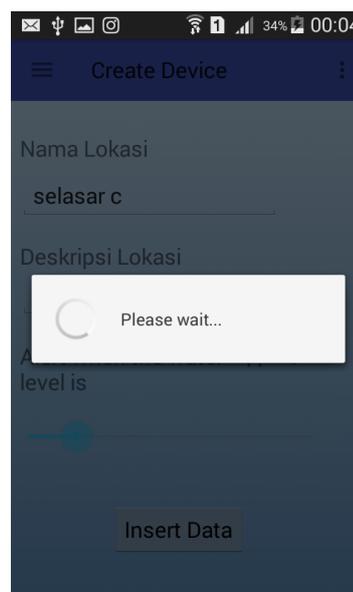
### 4.2. Create Galon

*Create* galon berfungsi untuk menambahkan *device* galon baru dan disimpan dalam *server* Ubidots dan *database* SQLite. Halaman ini dapat diakses oleh pengguna dengan menekan tombol “+” yang ada pada halaman *home* dan menekan *create device* pada *menu bar*. Halaman *create* galon dapat dilihat pada Gambar 5.

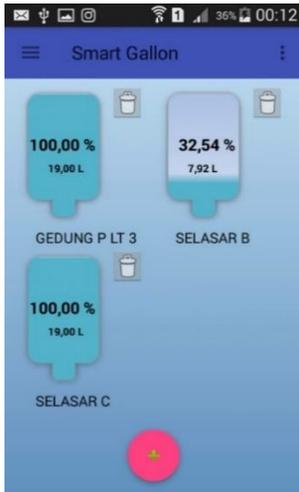
Pada *form* tersebut pengguna dapat mengisi nama lokasi, deskripsi lokasi dan *alert level* kemudian menekan tombol *insert* data, maka sistem akan membuatkan variabel baru yaitu *Flow, Volume*, dan koneksi di *server* Ubidots kemudian sistem mengambil masing - masing id variabel yang kemudian sistem akan menyimpan semua data form beserta id variabel ke dalam *database* SQLite. Dalam proses penambahan *device* galon sistem akan menampilkan *progress dialog* seperti pada gambar 6. Setelah data telah tersimpan maka sistem akan kembali ke halaman *home* seperti pada gambar 7.



Gambar 5. Tampilan Halaman Add Device



Gambar 6. Progress Dialog



Gambar 7. Tampilan Setelah Insert Berhasil

### 4.3. View Update Galon

Pengguna juga dapat mengubah *device* galon yang ada bila diperlukan dengan menekan gambar galon yang ada pada halaman *home*. Ketika pengguna menekan gambar galon tersebut, maka akan menampilkan *form* detail galon yang dipilih beserta dengan *valuenya* seperti pada gambar 8 dan 9. Kemudian pengguna dapat mengubah deskripsi lokasi dan *alert level* kemudian menekan tombol *update*.



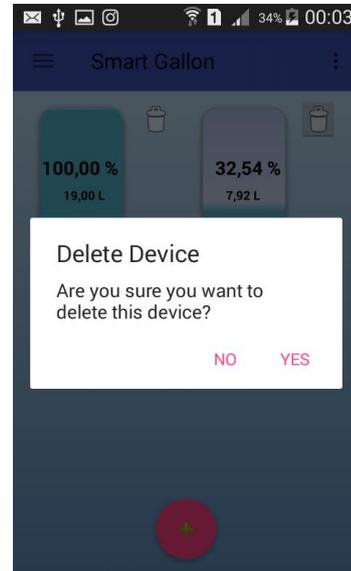
Gambar 8. Tampilan Detail Device



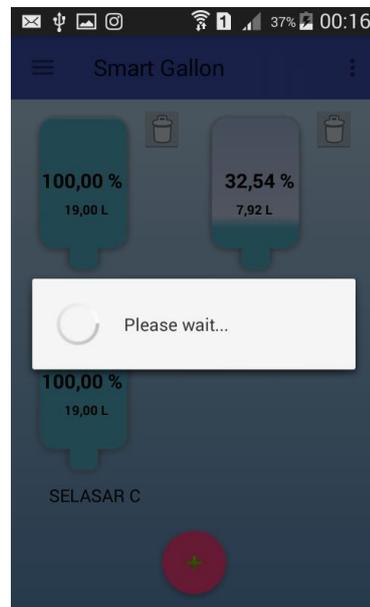
Gambar 9. Tampilan Setelah Update Galon

### 4.4. Delete Galon

Pengguna juga dapat menghapus sebuah *device* galon bila dirasa sudah tidak diperlukan lagi dengan menekan tombol  yang ada disamping gambar galon. aplikasi akan memberikan *alert dialog* "Are you sure you want to delete this device?" seperti yang ditampilkan pada gambar 10. Kemudian sistem akan melakukan *request delete* ke *server* Ubidots untuk menghapus *device* yang sesuai dengan ID *Datasource device*.



Gambar 10. Tampilan Alert Delete Device



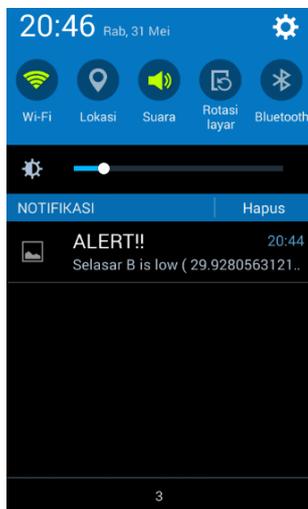
Gambar 11. Delete Progress Dialog

### 4.5. Alert Notifikasi

Pada saat volume air berada dibawah alert level yang telah di *setting* oleh pengguna maka sistem aplikasi akan memberikan notifikasi pada *smartphone* pengguna dan pengguna dapat menekan alert notifikasi tersebut agar aplikasi menampilkan kondisi galon seperti pada gambar 13 dan 14. Notifikasi yang diberikan berupa nada dering dan getaran. Apabila pengguna tidak mengubah *setting-an alert level* air pada detail galon maka sistem memberikan notifikasi selama 2 menit sekali.



Gambar 12. Tampilan Setelah Delete Device



Gambar 13. Tampilan Notifikasi



Gambar 14. Tampilan Setelah Menekan Notifikasi

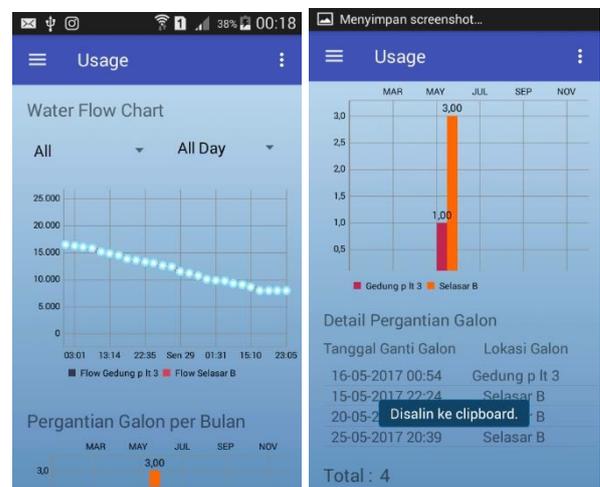
#### 4.6. Uji Pergantian Galon

Pengujian pergantian galon ini dilakukan 2 kali percobaan. Percobaan pertama adalah pengguna melakukan penekanan pada *switch* sekali saja. Dalam percobaan pertama, pergantian galon dalam penekanan pada *switch* sekali ini mengalami halangan pada proses perhitungan *flow* air. Hal ini dikarenakan

proses perhitungan *flow* tidak diberhentikan akibatnya adalah ketika galon air dituangkan diatas guci air minum maka aliran air itu mengenai rotor pada *water flow* sensor sehingga secara otomatis mikrokontroler akan menghitung aliran air tersebut. Solusi yang tepat pada percobaan yang kedua yaitu menekan tombol *switch* sebanyak 2 kali. Untuk penekanan *switch* pertama, mikrokontroler akan menghentikan proses perhitungan *flow*. Setelah galon air sudah diganti baru, maka pengguna menekan sekali lagi agar mikrokontroler akan mengirimkan *volume* air yang baru beserta menjalankan kembali proses perhitungan *flow*.

#### 4.7. View Laporan Pemakaian Galon

Pada saat pengguna menekan *usage gallon* pada *menu bar* maka aplikasi akan menampilkan halaman laporan pemakaian galon seperti pada gambar 15. Halaman laporan pemakaian galon ini terdapat 2 *chart* yaitu *LineChart* untuk menampilkan kondisi pengurangan volume air galon dan *BarChart* untuk menampilkan pergantian galon per bulan.



Gambar 15. Tampilan LineChart dan BarChart

#### 4.8. Pengujian Akurasi Water Flow Sensor

Pengujian akurasi keluar masuknya air dari galon ini ditujukan pada tingkat akuratnya *water flow* sensor dan perhitungan laju alir air di mikrokontroler dengan persamaan  $(Pulse\ frequency \times 60\ min) / 7.5Q = flowrate\ in\ L/hour$ . Untuk mengetahui tingkat akurasinya maka diperlukannya pengamatan terhadap hasil pembacaan dari laju alir air yang dapat dilihat mikrokontroler. Hal pertama yang dilakukan adalah menyiapkan gelas ukur dengan ukuran *volume* 0-750 ml dan botol *aqua* 1,5 liter. Setelah itu wadah tersebut diisi dengan air yang kemudian hasil pembacaan sensor tersebut akan dicatat ke dalam tabel 1. Data yang telah diambil untuk pengujian ini dapat dilihat pada tabel 1 akan ditampilkan perbandingan *volume* air dengan *volume* pemakaian dengan hasil pembacaan sensor. Percobaan yang dilakukan secara 5 kali untuk mengetahui perbedaan hasil pembacaan sensor.

Tabel 1. Perbandingan antara Volume Pemakaian dengan Hasil Pembacaan Sensor

Volume Pemakaian	Hasil Pembacaan Sensor				
	1	2	3	4	5
1500 ml	1484.46	1502.24	1502.24	1497.79	1501.16
750 ml	762.22	755.32	752.4	750	750.8
330 ml	324.4	331.34	342.2	330.67	334.53
250 ml	260	250.79	240	251.12	253.33

Dari data yang diperoleh dari tabel 1 ini dapat dihitung rata – rata dan tingkat ketidaksamaannya dalam bentuk persentase yang dapat dilihat pada tabel 2.

**Tabel 2. Tabel Perhitungan Rata-Rata dan Error pada Volume Air**

Volume Pemakaian	Rata-rata Hasil Pembacaan Sensor	Error (%)
1500 ml	1497.58	0.16
750 ml	754.15	0.55
330 ml	332.63	0.79
250 ml	251.05	0.42
Total rata-rata persentase error		0.48

## 5. KESIMPULAN

Dari hasil pembuatan pendeteksi *volume* air pada galon berbasis *internet of things* dengan menggunakan Android dan Arduino, dapat diambil kesimpulan antara lain:

- *Water flow* sensor dapat berkomunikasi dengan mikrokontroler dengan baik dan mikrokontroler juga dapat mengirimkan data ke *server* Ubidots.
- Aplikasi Android dapat memantau keadaan galon dan dapat menginformasikan pada pengguna melalui notifikasi apabila *volume* air galon berada dibawah *alert level*.

- Aplikasi dapat memudahkan pengguna dalam mencari informasi seperti laporan pergantian galon per bulannya.
- Aplikasi dapat menampilkan keadaan galon secara *real time*.
- Berdasarkan pengujian pergantian galon adalah pengguna harus menekan tombol *switch* sebanyak 2 kali.
- Berdasarkan pengujian akurasi laju alir air yang dilakukan, *water flow* sensor memiliki presentase error adalah 0.48%. maka *water flow* sensor ini layak untuk digunakan pada pembuatan pendeteksi *volume* air.

## 6. DAFTAR REFERENSI

- [1] Arduino. 2015. *Arduino - Environment*. URI=<https://www.arduino.cc/en/guide/environment>
- [2] Hosamani, R., & Bagade, R. 2015. Arduino based water billing system for domestic purpose. *International journal of modern trends in engineering and research*, 2(6), 424-432.
- [3] Ibrahim, D. 2002. *Microcontroller based temperature monitoring and control*. Woburn: Biddles Ltd.
- [4] Ubidots. 2016. *REST API Reference*. URI=<https://ubidots.com/docs/api/#what-is-a-restful-api>
- [5] Vermesan, O., & Friess, P. 2013. *Internet of Things: Converging technologies for smart environments and integrated ecosystems*. Aalborg: River Publisher.