

SISTEM PENGENDALIAN DAN PENGAWASAN AQUAPONIK PINTAR MENGUNAKAN APLIKASI *MOBILE* BERBASIS ANDROID

SMART AQUAPONICS CONTROLING AND MONITORING SYSTEM USING ANDROID BASED MOBILE APPLICATION

Ramadhika Halim Nurbed¹, Ahmad Tri Hartanto², Sevierda Raniprima³

^{1,2,3} Universitas Telkom, Bandung

ramadhikahalimnurbed@student.telkomuniversity.ac.id¹, athanuranto@telkomuniversity.ac.id²,
raniprimas@gmail.com³

Abstrak

Padatnya pemukiman di daerah perkotaan membuat lahan yang digunakan untuk menghasilkan pangan bagi kebutuhan manusia juga berkurang. Dengan demikian diperlukan sebuah sistem pertanian modern yang tidak banyak memakan tempat dan waktu yang terbuang, serta menghasilkan panen maksimal. Akan tetapi sistem akuaponik konvensional masih banyak kekurangannya dalam menjaga kualitas air sehingga mengakibatkan banyak akuaponik yang gagal panen. Disini dirancang sistem akuaponik pintar yang terdiri dari alat dengan beberapa sensor kualitas air (sensor suhu, sensor pH, sensor ketinggian air, serta sensor kekeruhan dan TDS), *webservice* dan *database* serta untuk *user interfacenya* yaitu aplikasi android untuk memudahkan pengguna akuaponik dengan memanfaatkan IoT (*Internet of Things*). Dan dengan sistem ini pengguna tidak perlu untuk *memonitor* dan mengontrol langsung kualitas air secara manual atau dengan cara yang konvensional. Hasil pengujian fungsionalitas, seluruh fitur yang terdapat di aplikasi *mobile* dapat dijalankan dengan baik. Untuk pengujian non-fungsionalitas, aplikasi dapat dijalankan di berbagai android yang mempunyai spesifikasi *smartphone* yang berbeda. Untuk hasil pengujian *delay* didapatkan nilai rata-rata pada proses *read database* sebesar 290,05 ms dan untuk proses *write database* sebesar 247,81 ms, sehingga nilai *delay* yang didapat bagus. Untuk hasil pengujian *availability* dan *reliability* didapatkan nilai sebesar 99,58% yang diuji selama 8 jam.

Kata Kunci: Akuaponik, Kualitas Air, IoT, Aplikasi *mobile*, Android, *Smartphone*

Abstract

Density of settlements in urban areas makes the land used to produce food for human needs is also reduced. Thus we need a modern agricultural system that does not take up much space and time wasted, and produces maximum yields, one of the popular modern farming models is aquaponics, but the conventional aquaponics system still has many shortcomings in maintaining water quality, resulting in many failed aquaponics. Here a smart aquaponics system is designed which consists of tools with several water quality sensors (temperature sensors, pH sensors, water level sensors, as well as turbidity sensors and TDS), a web server and database, as well as for the user interface using an android application to facilitate aquaponics users by utilizing IoT (Internet of Things). And with this system, users do not need to directly monitor and control water quality manually or by conventional. For this final project, the author focuses more on designing and testing mobile applications android-based. The results of functionality testing, all the features contained in the mobile application can be run properly. For non-functionality testing, the application can be run on various androids that have different smartphone specifications. For the delay test results, the average value for the database read process is 290.05 ms and for the database write process is 247.81 ms, so the delay obtained is good. For the results of availability and reliability testing, a value of 99.58% was obtained which was tested for 8 hours.

Keyword: *Aquaponics, Water Quality, IoT, Mobile Application, Android, Smartphone*

1. Pendahuluan

Pertanian merupakan salah satu sektor yang penting di Indonesia, karena sektor pertanian mempunyai peranan dari keseluruhan perekonomian nasional guna mewujudkan kesejahteraan masyarakat. Namun dengan berkembangnya laju pertumbuhan dan teknologi pada kawasan perkotaan menyebabkan penurunan lahan

pertanian sebesar 14,18% dari 5.218 hektar ditahun 2011 menjadi 4.570 pada tahun 2016 [1]. Alhasil kebutuhan rumah tangga untuk pangan pun juga menjadi tidak tercukupi [2].

Dengan laju pertumbuhan yang meningkat dampak lainnya dari kepadatan penduduk yaitu limbah. Masih banyak orang tidak menghiraukan akibat dari limbah ini terkhusus limbah cair yang langsung dibuang kelaut yang menyebabkan laut menjadi kotor [3]. Dengan demikian diperlukan sebuah sistem pertanian modern yang tidak banyak memakan tempat dan waktu yang terbuang, serta menghasilkan panen maksimal, salah satu model pertanian modern yang sangat digemari sekarang yakni Akuaponik [4].

Pada sistem ini parameter yang dipakai adalah suhu air, pH air, temperatur air, ketinggian air dan total zat terlarut dalam air, demi menjaga kualitas air pada sistem akuaponik, keakuratan data dari parameter ditambah dengan metode *Fuzzy Logic*. Dengan data yang ada akan diolah secara otomatis selanjutnya data akan diunggah ke *database* dan dapat dimonitor dan dikontrol melalui aplikasi *mobile* berbasis *android*. Dengan aplikasi *mobile* dan sistem yang akan dirancang, Pemilik dapat melihat dan mengontrol kualitas air secara jarak jauh dan tidak perlu sering kali langsung ke tempat untuk sekedar mengecek kondisi. Maka dari itu diharapkan sistem yang dibuat dapat menjadi solusi untuk efisiensi dalam budidaya ikan dan tumbuhan yang dibuat dalam suatu sistem pintar aquaponik, seperti sampel yang dipakai yaitu ikan lele dan tumbuhan kangkung, demi meningkatkan produktifitas kegiatan pembudidaya ikan dan tumbuhan.

2. Dasar Teori

2.1 Aquaponik Pintar (*Smart Aquaponics*)

Aquaponik yang biasanya tradisional dibuat menjadi aquaponik pintar atau biasa disebut *smart aquaponics*. Yakni sebuah sistem budidaya gabungan antara perikanan dan tanaman dalam satu wadah secara otomatis. Hal terpenting dari sebuah sistem aquaponik ini yaitu penyediaan air yang optimum terhadap masing-masing komoditas dan memanfaatkan resirkulasi air yang membutuhkan perawatan intensif untuk me-monitor sistem tersebut seperti yang dapat mengawasi kadar pH air, suhu air dan lainnya [4]. Sistem ini juga menghemat penggunaan air dalam budi daya ikan sampai 97%. [6]

2.2 Ikan Lele (*Clarias sp*)

Ikan lele merupakan salah satu komoditas yang sangat potensial untuk dikembangkan, dibalik harganya yang relatif murah serta mempunyai nilai gizi yang tinggi, perawatan dari ikan lele sangat lah mudah, tidak perlu memiliki tempat atau pun air bersih. Selain itu keunggulan yang lain yakni pertumbuhan yang relatif cepat dan cukup adaptif terhadap kondisi lingkungan [7].

2.3 Kangkung (*Ipomea aquatic forsk sp*)

Hampir semua jenis tanaman air bisa digunakan dalam sistem aquaponik. Tapi yang dipilih disini yaitu tanaman kangkung. Kangkung sendiri bisa menjadi *biofilter* yang dapat menyerap nitrogen dalam bentuk amonium (NH_4^+) dan nitrat (NO_3^-) sehingga nitrogen didalam air yang menjadi faktor kualitas air dalam budi daya perikanan. Dan pada tanaman sendiri nitrat berfungsi sebagai nutrisi yang dimanfaatkan sebagai hara serta pemanfaatan ammonia oleh tanaman sebagai filter biologis akan mengurangi konsentrasi racun pada budi daya ikan itu sendiri [6].

2.4 Kualitas Air

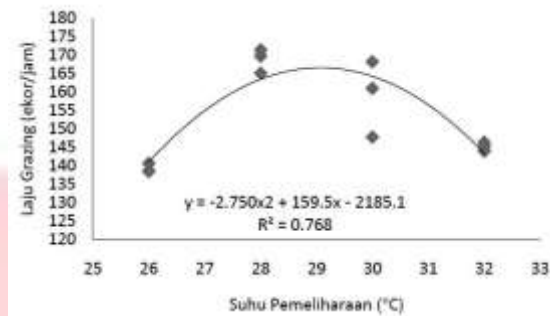
Kualitas air merupakan kandungan zat, energi atau bisa saja komponen lain yang ada di dalam air. Kualitas air dapat dinyatakan dalam parameter fisika seperti suhu, kekeruhan, padatan terlarut dan sebagainya. Serta parameter kimi seperti pH, Besi, Nitrit dan sebagainya [8]. Dan pada penelitian ini parameter yang diukur yakni pH, suhu air, kekeruhan air dan juga padatan terlarut dalam air.

2.4.1. pH Air

Dalam budidaya ikan lele, kestabilan pH air merupakan salah satu parameter kualitas air sehingga dikatakan baik. Pada budidaya lele pH air kolam itu harus dijaga dalam rentang 5,5-7,5 atau bisa dikatakan air yang netral. Karena ketika air itu dikondisi < 5,5 bakteri dan jamur akan cepat berkembang biak dibanding kondisi air yang netral [9].

2.4.2. Suhu Air

Suhu air adalah salah satu parameter yang menunjang konsumsi pakan oleh ikan lele sendiri. Tingkat makanan yang akan dikonsumsi ikan lele ini akan meningkat setiap kenaikan suhu sebesar 10°C dan diatas suhu 13-15°C akan meningkat dua kali lebih banyak, oleh sebab itu suhu ideal untuk pemeliharaan ikan lele adalah 25-30°C [10].



Gambar 2. 1 Grafik hubungan antara suhu dan pemangsaan selama 15 hari [10]

2.4.3. Kekeruhan dan Zat Terlarut dalam Air (TDS)

Tingkat kekeruhan air dipengaruhi oleh kadar partikel yang terlarut di dalamnya. Secara umum, kondisi ini disebut dengan Total Dissolved Solids pada air. Parameter kualitas air bersih juga diatur oleh Peraturan Menteri Perindustrian RI No. 78 tahun 2016 dengan ketentuan tingkat kekeruhan air bersih sebesar 25 NTU dan Total Dissolved Solids (TDS) sebesar 1500 mg/L. Secara fisis, indikator air bersih yaitu air yang bening, tidak berwarna, dan tidak berbau. Sedangkan secara optis, air yang tercampur oleh bahan pengotor keadaannya akan berubah warna, dan tingkat kekeruhannya [11].

2.5 IoT (Internet of Things)

Internet of Things (IoT) bisa dikatakan sebuah struktur di mana objek, orang disediakan dengan identitas eksklusif dan kemampuan untuk pindah data melalui jaringan tanpa memerlukan dua arah antara manusia ke manusia yaitu sumber ke tujuan atau interaksi manusia ke komputer. Iot merupakan perkembangan teknologi yang menjanjikan, IoT dapat mengoptimalkan kehidupan dengan sensor sensor cerdas dan benda yang memiliki jaringan dan bekerjasama dalam internet [12]

2.6 Aplikasi Mobile

Aplikasi *mobile* merupakan program komputer yang dirancang dalam bentuk *mobile* atau *software mobile* pada platform sistem operasi seperti *android*, *iOS* dan lainnya. Aplikasi dalam platform tersebut dirancang secara khusus sehingga dapat berjalan diperangkat seperti *smartphone*, tablet dan sebagainya. Umumnya aplikasi *mobile* ini memiliki *user interface* (UI) atau sebagai *front-end* yang terintegrasi dengan *web server* sebagai pengolahan datanya atau biasa disebut *back-end* [13].

2.7 UI/UX (User Interface & User Experience)

Jika membahas tentang aplikasi *mobile*, maka komponen utama yang membentuk sebuah aplikasi yaitu, *user interface* (UI) dan *user experience* (UX). Ketika pengguna melihat dan merasakan tampilan visual ketika memakai aplikasi itulah fungsi dari UI/UX sendiri. UI disini mencakup tata desain sebuah aplikasi seperti *button*, gambar, teks, serta semua *item* yang berinteraksi dengan pengguna. Sedangkan untuk UX sendiri adalah cara meningkatkan kepuasan pengguna pada saat interaksi dengan produk/aplikasi agar mudah digunakan [14].

2.8 Android

Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Sistem operasinya sendiri bersifat *opensource*, sehingga sangat terbuka bagi para pengembang dari seluruh dunia. Dan Android sendiri memfasilitasi dengan aplikasi yang dapat mengembangkan Android itu sendiri yakni Android Studio yang sudah terintegrasi dengan pemrograman Android [15].

2.9 Android Studio IDE (Integrated Development Environment)

Android Studio adalah sebuah aplikasi pengembangan Android yang resmi untuk mengembangkan aplikasi, yang didasarkan pada IntelliJ IDEA. Selain sebagai editor kode, Android Studio memiliki banyak fitur yang menunjang pengembangan aplikasi Android, seperti sistem build berbasis *Gradle*, emulator, framework dan fitur pengujian yang lengkap serta masih banyak lagi fitur yang memudahkan pengembangan aplikasi Android [16].

2.10 Kotlin

Kotlin merupakan bahasa pemrograman berbasis *Java Virtual Machine (JVM)* yang pragmatis untuk android yang mengkombinasikan *object oriented (OO)* dan pemrograman fungsional. Kotlin juga bahasa

pemrograman yang interoperabilitas yang membuat bahasa ini dapat digabungkan dalam satu *project* dengan bahasa pemrograman Java [17].

2.11 MySQL – phpMyAdmin

MySQL merupakan perangkat lunak sistem manajemen basis data yang memiliki kelebihan kecepatan serta keaamanan datanya, walaupun tidak secepat perangkat lunak lainnya. SQL sendiri adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data [18].

2.12 Parameter Pengujian Aplikasi

Pengujian aplikasi sendiri yaitu mengetahui kesalahan atau bug pada aplikasi, terdapat pengujian secara fungsionalitas dan juga non – fungsionalitas, Berikut parameter pengujian aplikasi yang digunakan pada tugas akhir ini agar dapat menjaga kualitas dari aplikasi serta memberikan kepuasan yang diinginkan oleh pengguna:

2.12.1. Black Box Testing

Pengujian ini termasuk kedalam pengujian fungsional aplikasi, yakni bertujuan melihat program tersebut apakah *user interface* aplikasi sesuai dengan fungsi yang diinginkan tanpa mengetahui kode program yang dipakai [19].

2.12.2. Compatibility Testing

Pengujian ini menggunakan metode *interoperability* yakni pengujian kemampuan dari dua atau lebih komponen perangkat lunak untuk melakukan pertukaran informasi dan juga melakukan fungsi yang dipakai ketika digunakan pada *hardware* ataupun lingkungan *software* yang sama ataupun berbeda [20].

2.12.3. User Experience Questionnaire (UEQ)

UEQ merupakan sebuah alat kuesioner yang digunakan untuk mengukur User Experience (UX). Data yang didapat digunakan untuk evaluasi setiap versi desainnya, didapatkan bahwa pada desain baru terjadi peningkatan rata-rata tiap skala UEQ. Evaluasi pengukuran berisi enam skala (aspek) antara lain Attractiveness (daya tarik), Perspicuity (kejelasan), Efficiency (efisiensi), Dependability (ketepatan), Stimulation (stimulasi), Novelty (inovasi). Dan dari keenam skala tersebut terdapat 26 pertanyaan singkat sebagai kuesioner pengujian user experience [21].

Tabel 2. 1 Skala penilaian UEQ

No.	Kiri	Kanan	Skala
1	Menyusahkan	Menyenangkan	Daya tarik
2	Tak dapat dipahami	Dapat dipahami	Kejelasan
3	Kreatif	Monoton	Kebaruan
4	Mudah dipelajari	Sulit dipelajari	Kejelasan
5	Bermanfaat	Kurang bermanfaat	Stimulasi
6	Membosankan	Mengasyikkan	Stimulasi
7	Tidak menarik	Menarik	Stimulasi
8	Tak dapat diprediksi	Dapat diprediksi	Ketepatan
9	Cepat	Lambat	Efisiensi
10	Berdaya cipta	Konvensional	Kebaruan
11	Menghalangi	Mendukung	Ketepatan
12	Baik	Buruk	Daya tarik
13	Rumit	Sederhana	Kejelasan
14	Tidak disukai	Menggembirakan	Daya tarik
15	Lazim	Terdepan	Kebaruan
16	Tidak nyaman	Nyaman	Daya tarik
17	Aman	Tidak aman	Ketepatan
18	Memotivasi	Tidak memotivasi	Stimulasi
19	Memenuhi ekspektasi	Tidak memenuhi ekspektasi	Ketepatan
20	Tidak efisien	Efisien	Efisiensi
21	Jelas	Mbingungkan	Kejelasan
22	Tidak praktis	Praktis	Efisiensi
23	Terorganisasi	Berantakan	Efisiensi
24	Atraktif	Tidak atraktif	Daya tarik

25	Ramah pengguna	Tidak ramah pengguna	Daya tarik
26	Konservatif	Inovatif	Kebaruan

2.13 Parameter Pengujian Jaringan

Pada parameter ini yang diuji kualitas baik atau buruknya jaringan pada saat berjalannya aplikasi. Berikut parameter pengujian jaringan yang digunakan pada tugas akhir ini untuk menjaga kualitas jaringan dari aplikasi *mobile* sendiri:

2.13.1. Delay

Delay atau *latency* merupakan waktu yang dibutuhkan untuk sebuah paket yang dikirimkan dari pengirim ke tujuan. Pada penelitian ini *delay* dihitung dari mulai aplikasi *mobile* mengirimkan perintah hingga data tersebut terkirim ke database maupun sebaliknya mengambil data dari database hingga sampai data ditampilkan pada aplikasi *mobile* [22].

Perhitungan *delay* ditunjukkan pada persamaan berikut :

$$\text{Rata - rata delay} = \frac{\text{Jumlah delay Seluruh paket}}{\text{Jumlah paket diterima}}$$

Tabel 2. 2 Spesifikasi Delay ITU-T G.1010 [23]

Medium	Application	Delay
Data	Bulk data transfer/retrieval	Preferred < 15 s Acceptable < 60s

2.13.1. Availability

Pengujian *availability* adalah suatu kemampuan sebuah sistem agar selalu tersedia ketika sistem tersebut digunakan [23]. Perhitungan *availability* ditunjukkan pada persamaan berikut:

$$\text{Availability} = \frac{\text{Uptime}}{(\text{Uptime} + \text{Downtime})} \times 100\%$$

2.13.1. Reliability

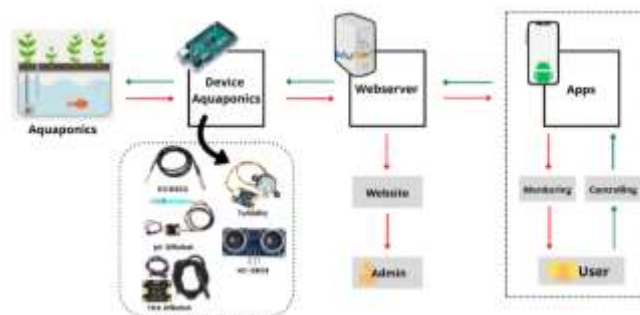
Pengujian *reliability* disini diperlukan untuk mengetahui kemungkinan suatu sistem terbebas dari suatu kegagalan pada saat sistem tersebut dijalankan [23]. Perhitungan *reliability* ditunjukkan pada persamaan berikut:

$$\text{Reliability} = \frac{(\text{Uptime} - \text{Downtime})}{\text{Uptime}} \times 100\%$$

3. Perancangan dan Simulasi Sistem

3.1 Desain Sistem

Dalam pengerjaan sistem ini, penulis akan fokus untuk mengembangkan aplikasi *mobile* yang merupakan satu segmen dari sistem *smart aquaponics*

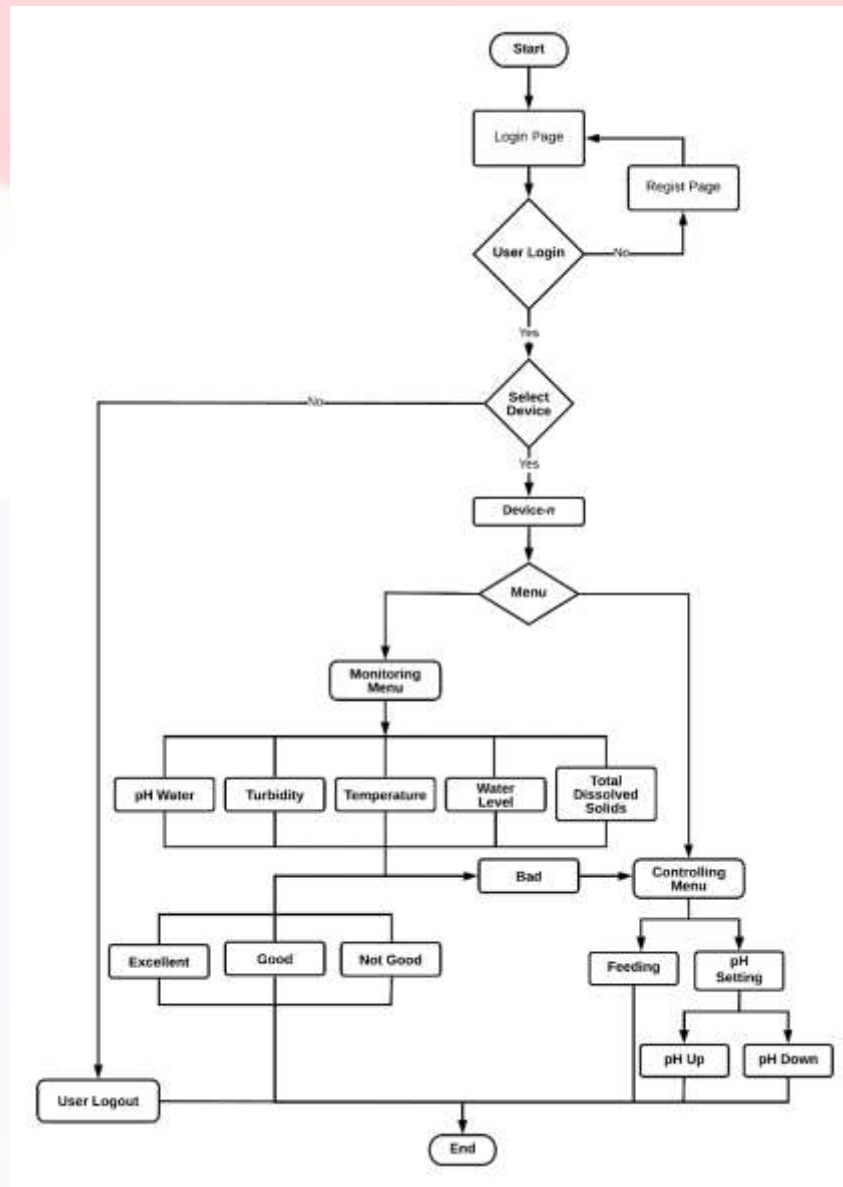


Gambar 3. 1 Desain sistem smart aquaponics

Sistem *smart aquaponics* disini merupakan sebuah sistem yang dapat mengawasi dan mengontrol aquaponik secara jarak jauh melalui aplikasi *mobile*. Sistem ini dapat melihat dan mengontrol parameter kualitas pada sistem aquaponik ini, seperti temperatur, ketinggian air, pH air, kekeruhan air dan juga zat

terlarut air. Selanjutnya pada sistem ini terdapat modul yang akan mengumpulkan dan mengirimkan data pada *web server*. Tahapan selanjutnya data yang sudah terunggah pada *webserver* akan diunduh dan ditampilkan dalam UI (*user interface*) aplikasi *mobile* agar pemilik dapat memonitor kondisi dari aquaponik dan selain itu proses sebaliknya melalui UI yang mengirimkan aksi ke *web server* untuk mengendalikan perangkat yang terdapat pada sistem aquaponik pintar ini serta pengontrolan bisa dilakukan manual agar lebih optimal dibandingkan secara otomatis.

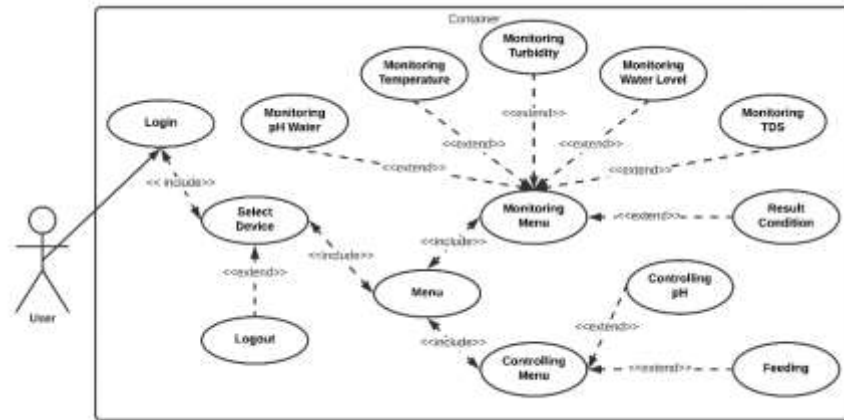
3.1.1. Diagram Alir Sistem Aplikasi



Gambar 3. 2 Diagram blok sistem aplikasi

. Dalam perancangan aplikasi *mobile* ini, untuk diawal sendiri terdapat menu *login* untuk masuk ke aplikasi. Setelah login terdapat pilihan device dari pemilik serta *logout* jika ingin keluar dari akun , ketika pemilik memiliki banyak device maka pada menu pilihan device akan terdeteksi device yang dimiliki. Setelah itu terdapat menu utama yang terdiri dari *monitoring* dan *controlling*, pertama untuk menu *monitoring* sendiri yang mana bila diakses oleh pemilik dapat melihat kondisi kualitas air perharinya mengenai pH, ketinggian air, kekeruhan air, temperatur air, dan kadar zat terlarut air, dan ketika hasil dari kualitas air itu kurang baik ataupun buruk, maka *user* harus masuk ke menu *controlling* untuk mengontrol pH dengan cara menaikkan atau menurunkan pH sesuai dengan kebutuhan pH tanaman yang di tanam oleh pemilik, serta memberikan pakan ikan secara manual dari aplikasi. Dan ketika ingin keluar dari akun maka kembali ke menu pilihan device dan *logout*.

3.1.2. Use case diagram

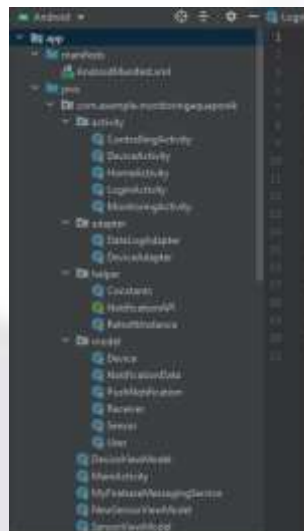


Gambar 3. 3 Diagram alir pengujian sistem aplikasi

Pada gambar 3.3 dijelaskan mengenai *use case* diagram digunakan untuk mengilustrasikan dari sistem yang dibuat. Ketika *user* menjalankan aplikasi akan masuk pada *login* saat pertama kali membuka aplikasi, kemudian setelah berhasil masuk *user* akan masuk ke pilihan *device*, disini *user* diberikan dua opsi, yaitu memilih *device* yang dia punya untuk di *monitoring* dan *controlling*, atau *logout* ketika tidak memiliki *device*. Setelah memilih *device* yang ada maka akan langsung diarahkan pada menu utama yang terdapat dua menu yakni menu *monitoring* dan menu *controlling*. Untuk menu *monitoring* sendiri digunakan untuk melihat kualitas air yang meliputi pH air, suhu air, kekeruhan air, ketinggian air, dan juga total zat terlarut dalam air, ditambah dengan kondisi keseluruhan dari kualitas air. Untuk menu *controlling* sendiri terdapat dua opsi fitur yakni *controlling* pH air dan juga pemberian pakan ikan secara manual.

3.2 Perancangan Aplikasi *Mobile*

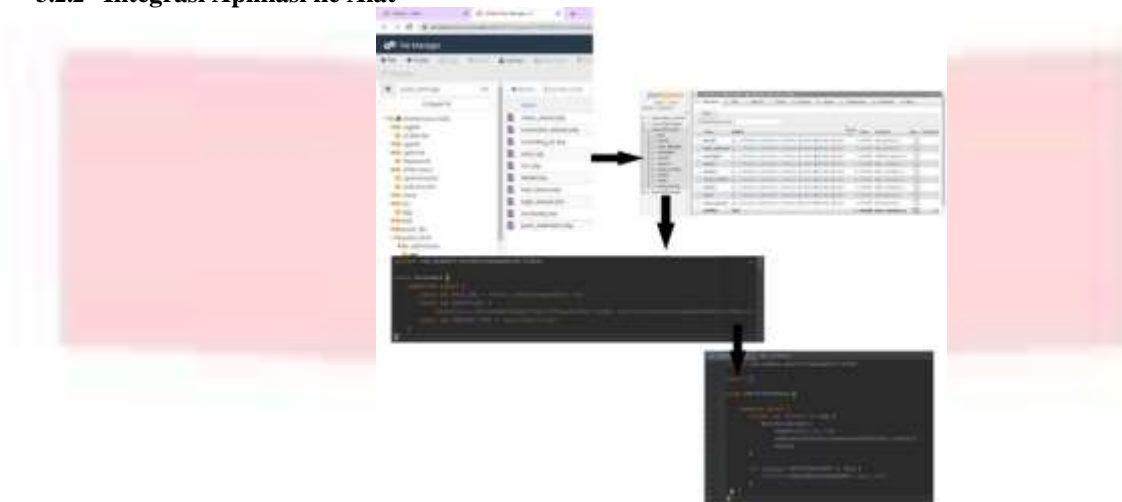
3.2.1 Konfigurasi Java Class



Gambar 3. 4 Konfigurasi Java Class

Java *Class* yaitu rancangan yang mendefinisikan variabel dan *method* pada suatu objek tertentu. *Class* dapat menampung isi program yang akan dijalankan, didalamnya terdapat berbagai *metgod* aktivitas aplikasi. Pada Gambar 3. 4 menunjukkan dalam perancangan aplikasi *mobile* kali ini terbagi beberapa *package* yang didalamnya terdapat Java *Class*.

3.2.2 Integrasi Aplikasi ke Alat



Gambar 3. 5 Integrasi Aplikasi dengan Database

Untuk intergrasi aplikasi ke alat melalui perantara database menggunakan REST API (*Represntional State Transfer Application Program Interface*). REST API merupakan arsitektur komunikasi data untuk menerima dan merespon data. Untuk API sendiri berada di *cpanel* atau kontrol panel hosting. Untuk di android dideklasikan untuk *url* dari hosting sendiri di *constant.kl* dan *RetrofitInstance.kl* untuk *library Rest Client* untuk android dan java.

3.3 Desain UI

3.3.1. Menu Login



Gambar 3. 6 Menu login

Pada menu *login* yang bisa dilihat pada Gambar 3.6 diatas, pengguna masuk menggunakan email serta password yang sudah terdaftar pada *database*, akan tetap ketika pengguna belum memiliki akun untuk masuk, terdapat pilihan “Daftar” dibawah tombol “Masuk”, ketika “Daftar” di pencet maka langsung pindah ke *browser* untuk melakukan registrasi pada halaman *website*. Dan ketika telah berhasil masuk selanjutnya akan berpindah ke menu pilihan *device*.

3.3.2. Menu Registrasi



Gambar 3. 7 Menu Registrasi

Pada menu registrasi yang bisa dilihat pada Gambar 3.7 diatas, untuk daftar agar bisa masuk ke aplikasi, pengguna wajib mengisi nama lengkap, email, password, alamat, dan juga nomor handphone. Dan ketika pengguna berhasil mengisi sesuai dengan yang diminta maka pengguna bisa balik ke menu *login* untuk mengakses aplikasinya

3.3.3. Menu Pilihan Device



Gambar 3. 8 Ketika terdapat pilihan *device*



Gambar 3. 9 Ketika tidak terdapat pilihan *device*

Pada menu pilihan *device* terdapat pilihan ketika akun yang terdaftar tersebut memiliki lebih dari satu *device* yang terdaftar pada *database*, seperti yang terlihat pada Gambar 3.8. Sebaliknya ketika akun yang terdaftar tetapi tidak memiliki *device* maka tidak akan menampilkan pilihan *device* seperti yang terlihat pada Gambar 3.9. Selanjutnya ketika pengguna telah memilih *device* yang akan di monitor dan dikontrol, beralih ke menu utama.

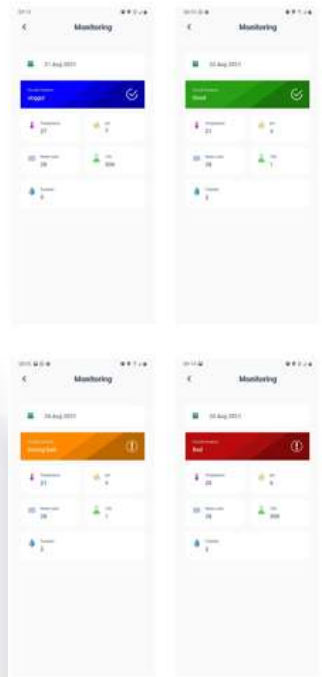
3.3.4. Menu Utama



Gambar 3. 10 Menu utama

Pada menu utama, terdapat pilihan menu *monitoring* dan *controlling*, yang mana pada menu *monitoring* tersebut berfungsi untuk memantau kondisi dari sistem akuaponik dengan menampilkan parameter-parameter yang menunjang kualitas air dari sistem akuaponik. Selanjutnya untuk menu *controlling* sendiri terdapat fitur mengatur pH naik dan pH turun untuk menjaga kestabilan pH air serta pemberian pakan ikan manual.

3.3.5. Menu Monitoring



Gambar 3. 11 Menu monitoring

Pada menu *monitoring* menampilkan tanggal, kondisi, serta parameter penunjang kualitas air antara lain temperatur air, pH, ketinggian air, kekeruhan air, serta total zat terlarut dalam air. Untuk tanggal sendiri bisa diganti sesuai dengan tanggal yang ingin dimonitoring oleh pengguna. Dan untuk *overall condition* adalah hasil dari kualitas air yang berdasarkan pada parameter-parameter kualitas air. Selanjutnya untuk parameter-parameter kualitas air didapat dari data *fuzzy* yang dikirimkan oleh alat ke aplikasi.

3.3.6. Menu *Controlling*



Gambar 3. 12 *Menu controlling*

Untuk menu *controlling* ini terdapat fitur untuk menjaga kestabilan pH air, untuk menjaga kestabilan pH sendiri terdapat pilihan untuk menaikkan ataupun menurunkan pH. Sedangkan untuk fitur lainnya yaitu pemberian pakan ikan melalui aplikasi tanpa harus ke kolam akuaponik.

3.3.7. Menu *Logout*



Gambar 3. 13 *Menu logout dari aplikasi*

Untuk menu *logout* sendiri diselipkan pada menu pilihan device yang terdapat pada simbol titik 3 di kanan atas aplikasi.

3.4 Spesifikasi Sistem

Berikut spesifikasi perangkat yang dipakai mulai dari perangkat lunak (*software*) ataupun perangkat keras (*hardware*) yang digunakan untuk perancangan aplikasi *mobile* pada penelitian ini:

- Perangkat Keras (*Hardware*)
 - Laptop :
 - Intel® Core™ i3-3217U CPU @ 1.80GHz (4 CPUs)
 - RAM 6144 MB
 - Sistem Operasi Windows 10 Pro 64-bit
 - VGA NVIDIA GeForce GT 625M
 - *Smartphone 1* :
 - Samsung Galaxy A32
 - Versi Android 11.0 (*Red Velvet Cake*)
 - RAM 6 GB & ROM 128 GB
 - *Smartphone 2* :
 - Oppo Reno 1
 - Versi Android 10.0 (*Queen Cake*)
 - RAM 8 GB & ROM 256 GB
 - *Emulator 1* :
 - MEmu Play
 - Versi Android 7.1.2 (*Nougat*)
 - RAM 6 GB (spesifikasi laptop)
- Perangkat Lunak (*Software*)
 - Android Studio IDE
 - MEmu Play

4. Hasil dan Analisis

4.1. Pengujian Aplikasi

Pada pengujian fungsi dari aplikasi menggunakan 3 parameter yakni, *black-box test*, *compatibility test*, serta *user experience quistionare*. Pengujian tersebut dilakukan untuk menjaga kualitas dari aplikasi serta kepuasan pengguna saat menggunakan aplikasi tersebut.

4.1.1. *Black-Box Testing*

Dalam pengujian ini dilakukan untuk mencoba semua fungsionalitas serta output dari aplikasi *mobile* yang diimplementasikan melalui *user interface*. Pengujian ini mengambil sampel pada menu yang ada yakni menu *login*, menu pilih *device*, menu utama, menu *monitoring*, dan juga menu *controlling*.

1. Pengujian pada menu *login*

Berikut ini adalah pengujian validasi *black-box* yang dilakukan pada fungsionalitas login. Pada Tabel 4.1 dilakukan pengujian *black-box* dengan cara memasukkan username dan password dari pengguna. Terdapat beberapa tes data yang akan dilakukan pada pengujian ini. Pertama, pengujian dilakukan dengan mengisi username dan password seara benar. Hasil pengujian pertama, pengguna dapat masuk ke sistem dan berpindah ke pilih *device*. Kedua, pengujian dilakukan dengan mengisi username dan password yang belum terdaftar. Hasil pengujian kedua, pengguna gagal masuk kedalam sistem dan muncul pesan "*Username dan Password Salah*". Ketiga, pengujian dilakukan dengan mengosongkan username dan password. Hasil pengujian ketiga, kolom username dan password memunculkan pesan untuk mengisi username dan password, agar bisa masuk ke dalam sistem.

2. Pengujian pada menu pilih device

Berikut ini adalah pengujian validasi *black-box* yang dilakukan pada fungsionalitas menu pilih device. Pada Tabel 4.2 dilakukan dengan cara memilih pilihan *device* yang tersedia pada akun pengguna. Untuk hasil yang diharapkan yaitu dapat menampilkan menu utama *monitoring* dan *controlling*. Dan dari hasil pengujian yang didapatkan yakni sistem dapat menampilkan menu *monitoring* dan *controlling* pada *device* yang sudah dipilih.

3. Pengujian pada menu utama

Berikut ini adalah pengujian validasi *black-box* yang dilakukan pada fungsionalitas menu utama. Pada Tabel 4.3 dilakukan pengujian untuk menu utama. Disini terdapat dua pengetesan yaitu mencoba fungsionalitas memilih menu *monitoring* dan menu *controlling*. Pertama untuk pengujian fungsionalitas dilakukan untuk memilih menu *monitoring* dengan cara menekan menu *monitoring* untuk menampilkan halaman *monitoring*. Untuk hasil yang diharapkan menampilkan halaman *monitoring*. Dan dari hasil pengujian yang didapatkan yakni sistem dapat menampilkan menu *monitoring*. Untuk kedua pengujian fungsionalitas dilakukan untuk menu *controlling* dengan cara menekan menu *controlling* untuk menampilkan halaman *controlling*. Untuk hasil yang diharapkan menampilkan halaman *controlling*. Dan dari hasil pengujian yang didapatkan yakni sistem dapat menampilkan menu *controlling*.

4. Pengujian pada menu monitoring

Berikut ini adalah pengujian validasi *black-box* yang dilakukan pada fungsionalitas menu *monitoring*. Pada Tabel 4.4 dilakukan pengujian menu *monitoring*. Disini terdapat pengetesan untuk memilih tanggal yang berbeda untuk melihat hasil *monitoring* pada hari yang berbeda. Untuk pengujian fungsionalitas dilakukan menekan tanggal yang ingin dipilih oleh pengguna. Untuk hasil yang diharapkan menampilkan halaman *monitoring* pada tanggal yang dipilih *user*. Dan dari hasil pengujian yang didapatkan yakni sistem dapat menampilkan tanggal yang ingin ditampilkan oleh *user*.

Tabel 4. 1 Pengujian *black-box* menu login

Deskripsi	Spesifikasi	Langkah	Input	Tes Data	Hasil yang diharapkan	Hasil
Melakukan login pada aplikasi.	Sistem dapat melakukan autentikasi terhadap user yang melakukan proses login pada aplikasi.	1. User memasukan email dan password 2. User menekan tombol login.	Email dan password	Email : iqbaladifm@gmail.com Password : admin	Berhasil masuk dan berpindah ke "Pilih Device".	Sesuai
				Email : aplikasirama@gmail.com Password : rama123	Gagal masuk kedalam sistem dan ditampilkan Toast: "username dan password salah".	Sesuai
				Email dan Password dikosongkan	Kolom username dan password bertanda merah, serta muncul pesan untuk mengisi username dan password agar bisa masuk ke dalam sistem.	Sesuai

Tabel 4. 2 Pengujian *black-box* menu pilih device

Deskripsi	Spesifikasi	Langkah	Input	Tes Data	Hasil yang diharapkan	Hasil
Memilih <i>device</i> yang tersedia pada akun <i>user</i> .	Sistem dapat memilih <i>device</i> yang terdaftar pada akun <i>user</i> .	<i>User</i> menekan pilihan <i>device</i> yang tersedia.	-	-	Menampilkan menu <i>monitoring</i> dan <i>controlling</i> pada <i>device</i> yang dipilih.	Sesuai

Tabel 4. 3 Pengujian *black-box* menu utama

Deskripsi	Spesifikasi	Langkah	Input	Tes Data	Hasil yang diharapkan	Hasil
Memilih menu <i>monitoring</i> dan <i>controlling</i>	Sistem dapat menampilkan menu yang dipilih <i>user</i>	Menekan menu <i>monitoring</i> .	-	-	Menampilkan halaman <i>monitoring</i> .	Sesuai
		Menekan menu <i>controlling</i> .	-	-	Menampilkan halaman <i>controlling</i> .	Sesuai

Tabel 4. 4 Pengujian *black-box* menu utama

Deskripsi	Spesifikasi	Langkah	Input	Tes Data	Hasil yang diharapkan	Hasil
Memilih tanggal yang berbeda untuk melihat hasil <i>monitoring</i> pada hari yang berbeda	Sistem dapat menampilkan tanggal yang dipilih <i>user</i>	Menekan tanggal yang dipilih	-	-	Menampilkan halaman <i>monitoring</i> pada tanggal yang dipilih <i>user</i> .	Sesuai

4.1.2. Compatibility Testing

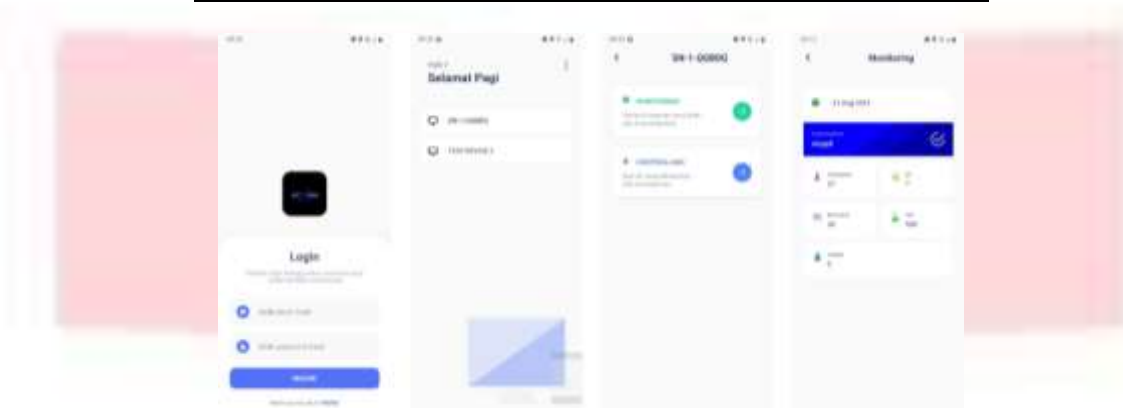
Pengujian ini bertujuan untuk mengetahui suatu sistem dapat dijalankan dengan baik ketika berada di lingkungan sistem yang berbeda. Pada pengujian ini dilakukan pada beberapa perangkat Android antara lain : Samsung Galaxy A32, Oppo Reno 1, dan MEmu Play.

1. Pengujian Compatibility pada Smartphone Samsung Galaxy A32

Pada pengujian ini, dilakukan dengan cara mengakses aplikasi menggunakan perangkat Samsung Galaxy A32 dengan versi android 11.0 serta memiliki RAM 6 GB DAN ROM 128 GB, berikut langkah-langkah pengujian pada perangkat ini :

Samsung Galaxy A32	
Langkah- Langkah	<ul style="list-style-type: none"> • Aplikasi <i>mobile</i> dijalankan • <i>Login</i> pada aplikasi <i>mobile</i> • Masuk ke pilihan <i>device</i> • Masuk ke menu utama • Membuka menu <i>monitoring</i>
Hasil yang diharapkan	Menu <i>monitoring</i> dapat ditampilkan

Hasil yang didapatkan	Menu <i>monitoring</i> dapat ditampilkan
Status	Valid



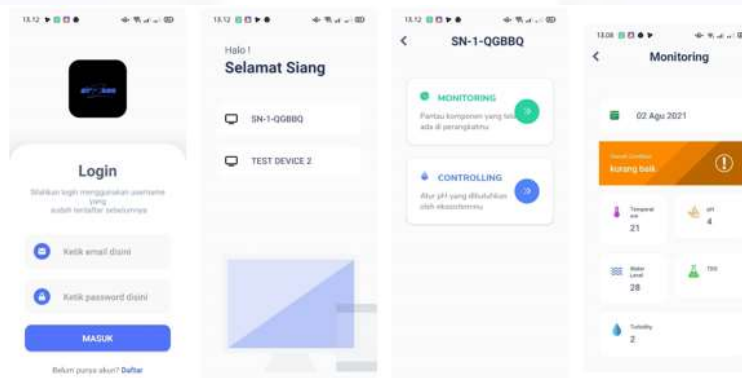
Gambar 4. 1 Pengujian *compatibility* aplikasi pada *smartphone* Samsung Galaxy A32

Pada Gambar 4.1 didapatkan hasil menu *login*, menu pilihan *device*, dan menu *monitoring* berhasil ditampilkan pada *smartphone* Samsung Galaxy A32. Hasil yang ditampilkan menunjukkan aplikasi dapat dijalankan tanpa ada kendala sesuai dengan hasil yang diharapkan.

2. Pengujian *Compatibility* pada *Smartphone* Oppo Reno 1

Pada pengujian ini, dilakukan dengan cara mengakses aplikasi menggunakan perangkat Oppo Reno 1 dengan versi android 10.0 serta memiliki RAM 8 GB DAN ROM 256 GB, berikut langkah-langkah pengujian pada perangkat ini :

Oppo Reno 1	
Langkah- Langkah	<ul style="list-style-type: none"> • Aplikasi <i>mobile</i> dijalankan • <i>Login</i> pada aplikasi <i>mobile</i> • Masuk ke pilihan <i>device</i> • Masuk ke menu utama • Membuka menu <i>monitoring</i>
Hasil yang diharapkan	Menu <i>monitoring</i> dapat ditampilkan
Hasil yang didapatkan	Menu <i>monitoring</i> dapat ditampilkan
Status	Valid



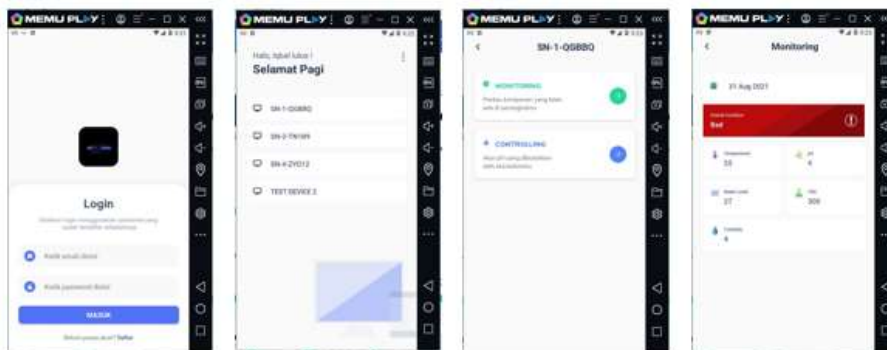
Gambar 4. 2 Pengujian *compatibility* aplikasi pada *smartphone* Oppo Reno 1

Pada Gambar 4.2 didapatkan hasil menu *login*, menu pilihan *device*, dan menu *monitoring* berhasil ditampilkan pada *smartphone* Oppo Reno 1. Hasil yang ditampilkan menunjukkan aplikasi dapat dijalankan tanpa ada kendala sesuai dengan hasil yang diharapkan.

3. Pengujian *compatibility* aplikasi pada *emulator* MEMu Play

Pada pengujian ini, dilakukan dengan cara mengakses aplikasi menggunakan emulator MEMu Play dengan versi android 7.1.2 serta memiliki RAM mengikuti laptop 6 GB, berikut langkah-langkah pengujian pada perangkat ini :

MEMu Play	
Langkah- Langkah	<ul style="list-style-type: none"> • Aplikasi <i>mobile</i> dijalankan • <i>Login</i> pada aplikasi <i>mobile</i> • Masuk ke pilihan <i>device</i> • Masuk ke menu utama • Membuka menu <i>monitoring</i>
Hasil yang diharapkan	Menu <i>monitoring</i> dapat ditampilkan
Hasil yang didapatkan	Menu <i>monitoring</i> dapat ditampilkan
Status	Valid

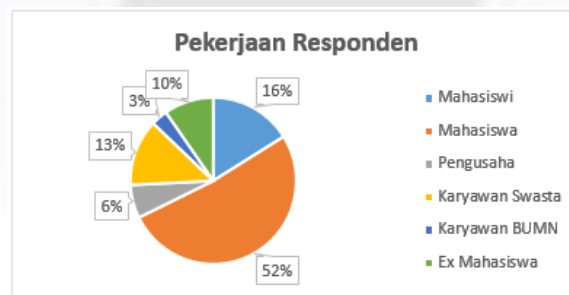


Gambar 4. 3 Pengujian *compatibility* aplikasi pada *emulator* MEMu Play

Pada Gambar 4.3 didapatkan hasil menu *login*, menu pilihan *device*, dan menu *monitoring* berhasil ditampilkan pada android *emulator* MEMu Play. Hasil yang ditampilkan menunjukkan aplikasi dapat dijalankan tanpa ada kendala sesuai dengan hasil yang diharapkan.

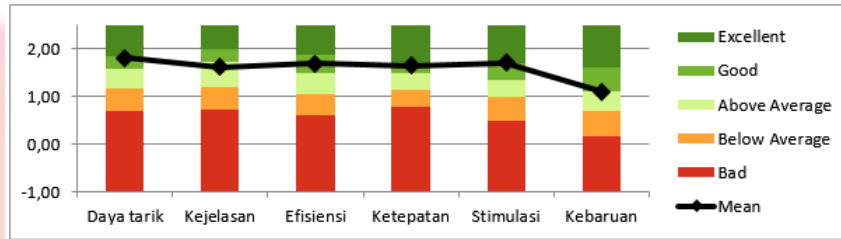
4.1.3. *User Experience Qustionnaire*

Kuesioner *User Experiiece Qutionnaire* (UEQ) yaitu alat kuesioner yang digunakan untuk menguji *user experience* sebuah produk dan menentukan perbaikan. Pada pengujian ini dilakukan dengan mengambil sampel dari 30 lebih responden yang beragam latar belakang dengan menggunakan kuesioner Google Form yang selanjutnya kuesioner diolah dalam *UEQ Data Analysis Tool*.



Gambar 4. 4 Presentase pekerjaan dari responden

Berikut dibawah terdapat grafik pada Gambar 4.7 dan Tabel 4.5 untuk hasil dari pengujian *User Experience* menggunakan *UEQ Tools*



Gambar 4. 5 Grafik hasil pengujian *user experience* dengan *UEQ tools*

Dari hasil pengujian didapatkan nilai daya tarik 1,80, kejelasan 1,62, efisiensi 1,68, ketepatan 1,65, stimulasi 1,70, dan kebaruan 1,10. Sehingga dapat disimpulkan bahwa untuk *user experience* dari aplikasi *mobile* ini sudah baik.

Tabel 4. 5 Hasil pengujian *user experience* dengan *UEQ tools*

Scale	Lower Border	Bad	Below Average	Above Average	Good	Excellent	Mean
Daya tarik	-1.00	0,69	0,49	0,4	0,26	0,66	1,81
Kejelasan	-1.00	0,72	0,48	0,53	0,27	0,5	1,62
Efisiensi	-1.00	0,6	0,45	0,45	0,38	0,62	1,69
Ketepatan	-1.00	0,78	0,36	0,34	0,22	0,8	1,65
Stimulasi	-1.00	0,5	0,5	0,35	0,35	0,8	1,70
Kebaruan	-1.00	0,16	0,54	0,42	0,48	0,9	1,10

Dari hasil diatas dapat dilihat untuk poin Kebaruan mendapat nilai *mean* yang paling kecil, maka skala tersebut dapat menjadi prioritas pengembangan untuk *user experience* selanjutnya.

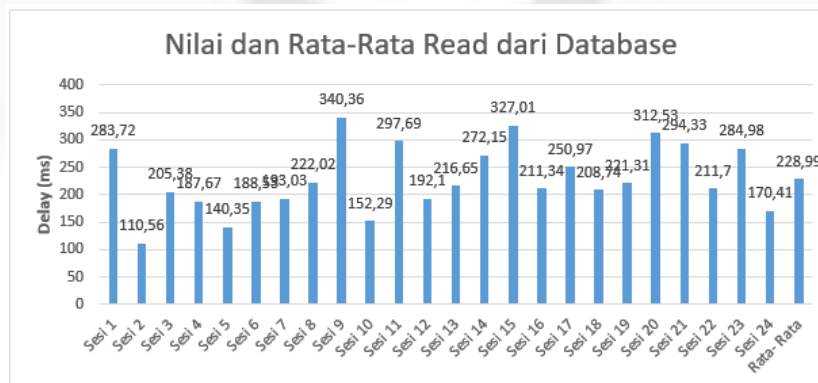
4.2 Pengujian Jaringan

Pada pengujian jaringan dari aplikasi menggunakan 3 parameter yakni, *delay*, *avaialability*, dan *reliability*. Pengujian tersebut dilakukan untuk mengetahui bagaimana performa kualitas jaringan pada aplikasi.

4.2.1 Delay

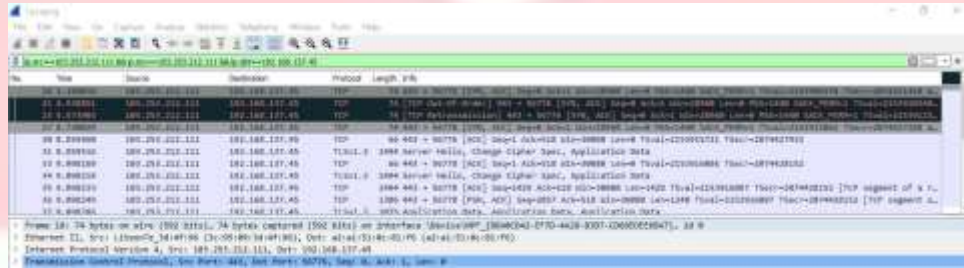
Pada pengujian *delay* ini pengamatan yang dilakukan dengan mengambil dua sampel aktivitas, yakni aktivitas unduh (*read*) dan unggah (*write*) pada database. Pengujian *read* sendiri dilakukan pada menu *monitoring* dan untuk pengujian *write* sendiri dilakukan pada menu *controlling*. Untuk perhitungan *delay* dilakukan sebanyak 24 sesi dengan cara menyambungkan *smartphone* ke *hotspot* dari laptop yang tersambung dengan jaringan internet.

4.2.1.1 Pengujian Delay Pada Proses Read Data dari Database



Gambar 4. 6 Grafik pengujian *delay* pada proses *read* database

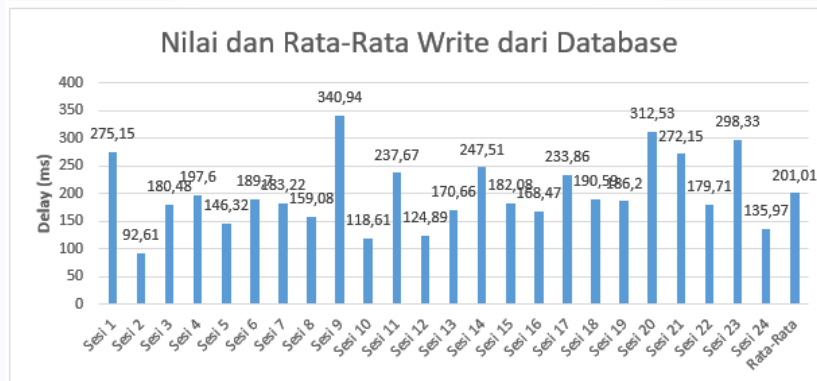
Pada grafik diatas pengukuran *delay* pada proses *read* data dari database memiliki nilai terendah pada sesi ke-2 dengan nilai 110,56 ms. Sedangkan untuk nilai tertinggi pada sesi ke-9 dengan nilai 340,36 ms. Dan untuk rata-rata dari 24 sesi dari delay itu bernilai 231,49 ms dan *delay* termasuk kategori baik. Berikut salah satu *screenshoot wireshark* pada pengambilan data dan untuk lainnya dilampirkan pada lampiran buku tugas akhir.



Gambar 4.7 Screenshoot wireshark pada sesi satu pengujian *delay read database*

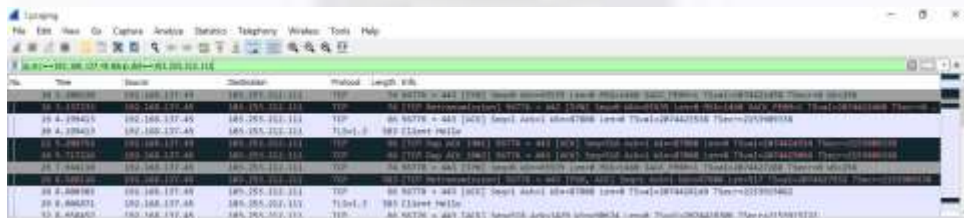
Pada Gambar 4.7 diatas terdapat *time* untuk memperlihatkan waktu yang dibutuhkan untuk melakukan perintah yang dilakukan. Untuk *source*, pengukuran *delay read database* terdapat IP yang bersumber pada server. Dan untuk *destination* pada pengukuran *delay read database* IP nya bersumber pada *smartphone*.

4.2.1.2 Pengujian *Delay* Pada Proses *Write* Data dari Database



Gambar 4.8 Grafik pengujian *delay* pada proses *write database*

Pada grafik diatas pengukuran *delay* pada proses *write* data dari database memiliki nilai terendah pada sesi ke-2 dengan nilai 92,61 ms. Sedangkan untuk nilai tertinggi pada sesi ke-9 dengan nilai 340,94 ms. Dan untuk rata-rata dari 24 sesi dari delay itu bernilai 201,01 ms dan *delay* termasuk kategori baik. Berikut salah satu *screenshoot wireshark* pada pengambilan data dan untuk lainnya dilampirkan pada lampiran buku tugas akhir.



Gambar 4.9 Screenshoot wireshark pada sesi satu pengujian *delay write database*

Pada Gambar 4.9 diatas terdapat *time* untuk memperlihatkan waktu yang dibutuhkan untuk melakukan perintah yang dilakukan. Untuk *source*, pengukuran *delay write database* terdapat IP yang bersumber

pada *smartphone*. Dan untuk *destination* pada pengukuran *delay read database* IP nya bersumber pada server.

4.2.2 Availability

Pada pengujian *availability* ini dilakukan pengamatan terhadap sistem aplikasi menggunakan android *emulator*, untuk durasi pengamatan sendiri dilakukan sebanyak 8 jam dengan durasi pertama 3 jam dan selanjutnya ditambah 1 jam sampai total durasi menjadi 8 jam agar mendapat hasil yang lebih akurat dan melihat apakah ada kesalahan pada sistem. Berikut tabel hasil pengujian dari *availability*.

Tabel 4. 6 Hasil pengujian dari *availability*.

Pengujian		Downtime		Penyebab	Availability (%)
Waktu	Durasi	Waktu	Total Durasi		
10:00 – 13:00	3 jam (180 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	98,91 %
10:00 – 14:00	4 jam (240 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,17 %
10:00 – 15:00	5 jam (300 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,34 %
10:00 – 16:00	6 jam (360 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,45 %
10:00 – 17:00	7 jam (420 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,52 %
10:00 – 18:00	8 jam (480 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,58 %

Dari hasil pengujian diatas nilai yang diambil adalah pada durasi 8 jam dikarenakan hasilnya yang bisa lebih akurat karena semakin lama durasi pengujian maka semakin akurat nilai yang didapat, jadi untuk nilai *availability* yakni 99,58% dan dikatakan telah layak karena mendekati nilai 100%.

4.2.3 Reliability

Untuk pengujian *reliability* ini dilakukan bersamaan dengan pengujian *availability*, untuk durasi pengamatan sendiri sama dengan pengujian *availability*, Berikut tabel hasil pengujian dari *reliability*.

Tabel 4. 7 Hasil pengujian dari *reliability*

Pengujian		Downtime		Penyebab	Reliability (%)
Waktu	Durasi	Waktu	Total Durasi		
10:00 – 13:00	3 jam (180 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	98 %
10:00 – 14:00	4 jam (240 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,16 %
10:00 – 15:00	5 jam (300 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,33 %
10:00 – 16:00	6 jam (360 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,44 %
10:00 – 17:00	7 jam (420 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,52 %
10:00 – 18:00	8 jam (480 menit)	12:14:29 – 12:16:14	2 menit	<i>Request Time Out</i>	99,58 %

Dari hasil pengujian diatas nilai yang diambil adalah pada durasi 8 jam dikarenakan hasilnya yang bisa lebih akurat karena semakin lama durasi pengujian maka semakin akurat nilai yang didapat, jadi untuk nilai *reliability* yakni 99,58% dan dikatakan telah layak karena mendekati nilai 100%.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Penelitian Tugas Akhir ini menghasilkan kesimpulan sesuai dengan tujuan yang telah tercapai.

1. Pada tugas akhir ini berhasil dibangun *user interface* dalam bentuk aplikasi berbasis android yang bisa melakukan *monitoring* serta pengawasan kualitas air dari sistem pintar akuaponik yang bisa dilakukan dari mana saja dan kapan saja.
2. Hasil dari pengujian aplikasi berdasarkan fungsionalitas menggunakan *black-box test* aplikasi *mobile* dapat berjalan sesuai dengan *user interface* yang dirancang.
3. Pada pengujian aplikasi berdasarkan non-fungsionalitas didapatkan hasil dari *compatibility test* dengan metode *interoperability*, aplikasi dapat dijalankan di *smartphone* Android maupun *emulator* Android dengan spesifikasi berbeda mulai dari Android 7.1.2 hingga Android 11.0 dapat menjalankan aplikasi ini. Hasil pengujian dari *user experience quistionnaire* didapatkan poin tertinggi pada skala penilaian daya tarik sebesar 1,81 dan untuk poin terendah pada skala penilaian kebaruan sebesar 1,10. Dan dapat disimpulkan bahwa *user experience* dari aplikasi sudah baik dan dapat dikembangkan berdasarkan poin skala yang didapatkan.
4. Pada pengujian jaringan hasil pengujian rata-rata *delay* pada proses proses *read* database sebesar 228,99 ms dengan nilai *delay* terendah 110,56 ms dan nilai *delay* tertinggi 340,36. Sedangkan untuk hasil rata-rata *delay* pada proses *write* database sebesar 201,01 ms dengan nilai *delay* terendah 92,61 ms dan nilai *delay* tertinggi 340,94 ms. sehingga hasil *delay* keseluruhan yang didapat bagus. Dan untuk hasil pengujian *availability* dan *reliability* didapatkan nilai yang sama yaitu sebesar 99,58% setelah dilakukan pengujian selama 8 jam.

5.2. Saran

1. Berdasarkan hasil pengujian UEQ, kebaruan adalah skala rata-rata terendah, seingga skala ini dapat lebih di prioritaskan untuk mengembangkan *user experience*.
2. Menambahkan *shortcut* menu *controlling* pada menu *monitoring*, agar ketika hasil *monitoring* kualitas air buruk bisa langsung berpindah ke menu *controlling*.

REFERENSI

- [1] A. Herlindawati, L. Trimo, and T. I. Noor, "ANALISIS TEKANAN PENDUDUK TERHADAP PETANI PADI SAWAH (Suatu Kasus di Kecamatan Cilamaya Kulon, Kabupaten Karawang, Jawa Barat) ," *Mimb. AGRIBISNIS J. Pemikir. Masy. Ilm. Berwawasan Agribisnis*, vol. 4, no. 1, pp. 12–24, 2018.
- [2] G. R. A. Kartika and A. P. W. K. Dewi, "Potensi Penerapan Sistem Budidaya Ikan Kombinasi Akuaponik Pada Skala Rumah Tangga," no. November 2019, pp. 1–5, 2020, [Online]. Available: https://www.researchgate.net/.../340826993_Potensi_Penerapan_Sistem_Budidaya_Ikan_.
- [3] N. A. Maharani and P. N. Sari, "Penerapan Aquaponic Sebagai Teknologi Tepat Guna Pengolahan Limbah Cair Kolam Ikan di Dusun Kergan, Tirtomulyo, Kretek, Bantul, Yogyakarta," *J. Pengabd. Kpd. Masy. (Indonesian J. Community Engag.*, vol. 1, no. 2, pp. 172–182, 2016, doi: 10.22146/jpkm.10603.
- [4] Zulhelman., H.A. Ausha., R.M. Ulfa., "Pengembangan Sistem Smart Aquaponik," *J. Poli-Teknologi*, vol. 15, no. 2, pp. 181–186, 2016.
- [5] R. Muzawi and W. J. Kurniawan, "Penerapan Internet of Things (IoT) Pada Sistem Kendali Lampu Berbasis Mobile," *J-SAKTI (Jurnal Sains Komput. dan Inform.*, vol. 2, no. 2, p. 115, 2018, doi: 10.30645/j-sakti.v2i2.75.
- [6] Z. Hasan, Y. Andriani, Y. Dhahiyat, A. Sahidin, and M. R. Rubiansyah, "Pertumbuhan tiga jenis ikan dan kangkung darat (*Ipomoea reptans* Poir) yang dipelihara dengan sistem akuaponik," *J. Iktiologi Indones.*, vol. 17, no. 2, p. 175, 2018, doi: 10.32491/jii.v17i2.357.
- [7] O. W. B. S. R. Prayogo, "Pengaruh Padat Tebar Ikan Lele Terhadap Laju Pertumbuhan Dan Survival Rate Pada Sistem Akuaponik
<I>[The Effect Of Stocking Density On Survival Rate And Grow Rate Of Aquaponic System]<I>," *J. Ilm. Perikan. dan Kelaut.*, vol. 6, no. 1, p. 55, 2019, doi: 10.20473/jipk.v6i1.11382.
- [8] M. H. D. Barang and S. K. Saptomo, "Analisis Kualitas Air pada Jalur Distribusi Air Bersih di Gedung Baru Fakultas Ekonomi dan Manajemen Institut Pertanian Bogor," *J. Tek. Sipil dan Lingkung.*, vol. 4, no. 1, pp. 13–24, 2019, doi: 10.29244/jsil.4.1.13-24.
- [9] Hermansyah, "RANCANG BANGUN PENGENDALI pH AIR UNTUK PEMBUDIDAYAAN IKAN LELE BERBASIS MIKROKONTROLER ATMEGA16," *J. Tek. Elektro Univ. Tanjungpura*, vol. 2, no. 1, pp. 1–13, 2017.
- [10] T. P. Lestari and E. Dewantoro, "PENGARUH SUHU MEDIA PEMELIHARAAN TERHADAP LAJU PEMANGSAAN DAN PERTUMBUHAN LARVA IKAN LELE DUMBO (*Clarias gariepinus*)," *J. Ruaya J. Penelit. dan Kaji. Ilmu Perikan. dan Kelaut.*, vol. 6, no. 1, pp. 14–22, 2018, doi: 10.29406/rya.v6i1.923.
- [11] R. P. Wirman, I. Wardhana, and V. A. Isnaini, "Kajian Tingkat Akurasi Sensor pada Rancang Bangun Alat Ukur Total Dissolved Solids (TDS) dan Tingkat Kekeruhan Air," *J. Fis.*, vol. 9, no. 1, pp. 37–46, 2019, doi: 10.15294/jf.v9i1.17056.
- [12] W. Wilianto and A. Kurniawan, "Sejarah, Cara Kerja Dan Manfaat Internet of Things," *Matrix J. Manaj. Teknol. dan Inform.*, vol. 8, no. 2, pp. 36–41, 2018, doi: 10.31940/matrix.v8i2.818.
- [13] Y. Prasetiawan, M. Jumansyah, and A. Hamzah, "Pengembangan aplikasi mobile monitoring kemajuan nilai mahasiswa kurikulum 2016 berbasis open learner model," vol. 1, no. 1, pp. 1–8, 2020, [Online]. Available: <https://journal.uui.ac.id/AUTOMATA/article/download/13890/9951>.
- [14] E. Tirtadarma, A. E. B. Waspada, and E. F. Jasjfi, "Kajian Peranan Desain UX (Pengalaman Pengguna) - UI (Antar Muka Pengguna) Mobile Application Kategori Transportasi Online terhadap Gaya Hidup Bertransportasi Masyarakat Urban," *J. Seni dan Reka Ranc. J. Ilm. Magister Desain*, vol. 1, no. 1, p. 181207, 2018, doi: 10.25105/jsrr.v1i1.4046.
- [15] D. I. Maulana and D. Susandi, "Rancang Bangun Aplikasi Silase Pakan Ternak Domba Berbasis Android,"

vol. 5, no. 1, pp. 94–100.

- [16] Google Developers, “Dokumen Android Developer | Android Studio,” *Developer.Android.Com*. 2020, [Online]. Available: <https://developer.android.com/docs>.
- [17] N. S. Sibarani, G. Munawar, and B. Wisnuadhi, “Analisis Performa Aplikasi Android Pada Bahasa Pemrograman Java dan Kotlin.”
- [18] R. E. Standsyah and I. S. Restu, “Implementasi PHP MyAdmin pada Rancangan Sistem Pengadministrasian,” *J. UJMC*, vol. 3, no. 2, pp. 38–44, 2017, [Online]. Available: <http://ejournal.unisda.ac.id/index.php/ujmc/article/download/467/251/>.
- [19] I. Ismail and J. Efendi, “Black-Box Testing : Analisis Kualitas Aplikasi Source Code Bank Programming,” *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 4, no. 2, p. 1, 2020, doi: 10.35870/jtik.v5i1.148.
- [20] T. A. Ghaffur, “Analisis Kualitas Sistem Informasi Kegiatan Sekolah Berbasis Mobile Web Di Smk Negeri 2 Yogyakarta,” *Elinvo (Electronics, Informatics, Vocat. Educ.)*, vol. 2, no. 1, pp. 94–101, 2017, doi: 10.21831/elinvo.v2i1.16426.
- [21] S. V. Izabal, I. Aknuranda, and H. M. Az-zahra, “Evaluasi dan Perbaikan User Experience Menggunakan User Experience Questionnaire (UEQ) dan Focus Group Discussion (FGD) pada Situs Web FILKOM Apps Mahasiswa Fakultas Ilmu Komputer Universitas Brawijaya,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 9, pp. 3224–3232, 2018.
- [22] Hasanul Fahmi, “Analisis Qos (Quality of Service) Pengukuran Delay, Jitter, Packet Lost Dan Throughput Untuk Mendapatkan Kualitas Kerja Radio Streaming Yang Baik,” *J. Teknol. Inf. dan Komun.*, vol. 7, no. 2, pp. 98–105, 2018.
- [23] ITU-T, “G.1010: End-user multimedia QoS categories,” *Int. Telecommun. Union*, vol. 1010, 2001, [Online]. Available: http://scholar.google.com.au/scholar?hl=en&q=ITU-T+Recommendation+G.1010&btnG=&as_sdt=1,5&as_sdt=7.
- [24] D. J. Trujillo and C. J. B. Scharmer, “Reliability , Availability , and Maintainability Considerations in the Design and Evaluation of Physical Security Systems,” pp. 1–12, 2012.