



**UNIVERSITAS INDONESIA**

**JUDUL SKRIPSI/THESIS/DISERTASI**

**TUGAS AKHIR**

**LUQMAN SUNGKAR**

**1106088303**

**FAKULTAS FAKULTAS ILMU KOMPUTER**

**PROGRAM STUDI SISTEM INFORMASI**

**DEPOK**

**JANUARI 2010**



**UNIVERSITAS INDONESIA**

**JUDUL SKRIPSI/THESIS/DISERTASI**

**TUGAS AKHIR**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Ilmu Komputer**

**LUQMAN SUNGKAR**

**1106088303**

**FAKULTAS FAKULTAS ILMU KOMPUTER**

**PROGRAM STUDI SISTEM INFORMASI**

**DEPOK**

**JANUARI 2010**

## HALAMAN PERSETUJUAN

**Judul** : Judul Skripsi/Thesis/Disertasi  
**Nama** : Luqman Sungkar  
**NPM** : 1106088303

Laporan Tugas Akhir ini telah diperiksa dan disetujui.

XX Januari 2010

Prof. XXXX

Pembimbing Tugas Akhir

## **HALAMAN PERNYATAAN ORISINALITAS**

**Tugas Akhir ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun dirujuk  
telah saya nyatakan dengan benar.**

**Nama : Luqman Sungkar**  
**NPM : 1106088303**  
**Tanda Tangan :**

**Tanggal : XX Januari 2010**

## HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh :

Nama : Luqman Sungkar

NPM : 1106088303

Program Studi : Sistem Informasi

Judul Tugas Akhir : Judul Skripsi/Thesis/Disertasi

**Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Ilmu Komputer pada Program Studi Sistem Informasi, Fakultas Fakultas Ilmu Komputer, Universitas Indonesia.**

## DEWAN PENGUJI

Pembimbing : Prof. XXXX ( )

Penguji : Prof. XXX ( )

Penguji : Prof. XXXX ( )

Penguji : Prof. XXXXXX ( )

**@todo**

Jangan lupa mengisi nama para penguji.

Ditetapkan di : Depok

Tanggal : XX Januari 2010

## KATA PENGANTAR

Template ini disediakan untuk orang-orang yang berencana menggunakan  $\text{\LaTeX}$  untuk membuat dokumen tugas akhirnya. Mengapa  $\text{\LaTeX}$ ? Ada banyak hal mengapa menggunakan  $\text{\LaTeX}$ , diantaranya:

1.  $\text{\LaTeX}$  membuat kita jadi lebih fokus terhadap isi dokumen, bukan tampilan atau halaman.
2.  $\text{\LaTeX}$  memudahkan dalam penulisan persamaan matematis.
3. Adanya otomatis dalam penomoran caption, bab, subbab, subsubbab, referensi, dan rumus.
4. Adanya otomatisasi dalam pembuatan daftar isi, daftar gambar, dan daftar tabel.
5. Adanya kemudahan dalam memberikan referensi dalam tulisan dengan menggunakan label. Cara ini dapat meminimalkan kesalahan pemberian referensi.

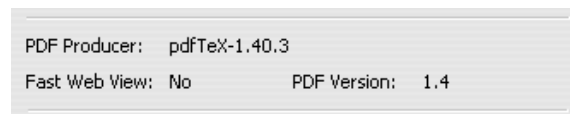
Template ini bebas digunakan dan didistribusikan sesuai dengan aturan *Creative Common License 1.0 Generic*, yang secara sederhana berisi:



**Gambar 1:** *Creative Common License 1.0 Generic*

Gambar 1 diambil dari [http://creativecommons.org/licenses/by-nc-sa/1.0/deed.en\\_CA](http://creativecommons.org/licenses/by-nc-sa/1.0/deed.en_CA). Jika ingin mengetahui lebih lengkap mengenai *Creative Common License 1.0 Generic*, silahkan buka <http://creativecommons.org/licenses/by-nc-sa/1.0/legalcode>. Seluruh dokumen yang dibuat dengan menggunakan template ini sepenuhnya menjadi hak milik pembuat dokumen dan bebas didistribusikan sesuai dengan keperluan masing-masing. Lisensi hanya berlaku jika ada orang yang membuat template baru dengan menggunakan template ini sebagai dasarnya.

Dokumen ini dibuat dengan  $\text{\LaTeX}$  juga. Untuk meyakinkan Anda, coba lihat properti dari dokumen ini dan Anda akan menemukan bagian seperti Gambar 2. Dokumen ini dimaksudkan untuk memberikan gambaran kepada Anda seperti apa mudahnya menggunakan  $\text{\LaTeX}$  dan juga memperlihatkan betapa bagus dokumen yang dihasilkan. Seluruh url yang Anda temukan dapat Anda klik. Seluruh referensi yang ada juga dapat diklik. Untuk mengerti template yang disediakan, Anda tetap harus membuka kode  $\text{\LaTeX}$  dan bermain-main dengannya. Penjelasan dalam PDF ini masih bersifat gambaran dan tidak begitu mendetail, dapat dianggap sebagai pengantar singkat. Jika Anda merasa kesulitan dengan template ini, mungkin ada baiknya Anda belajar sedikit dasar-dasar  $\text{\LaTeX}$ .



**Gambar 2:** Dokumen Dibuat dengan PDFLatex

Semoga template ini dapat membantu orang-orang yang ingin mencoba menggunakan  $\text{\LaTeX}$ . Semoga template ini juga tidak berhenti disini dengan ada kontribusi dari para penggunanya. Kami juga ingin berterima kasih kepada Andreas Febrian, Lia Sadita, Fahrurrozi Rahman, Andre Tampubolon, dan Erik Dominikus atas kontribusinya dalam template ini.

Depok, 30 Desember 2009

Luqman Sungkar

## HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

**Nama** : Luqman Sungkar  
**NPM** : 1106088303  
**Program Studi** : Sistem Informasi  
**Fakultas** : Fakultas Ilmu Komputer  
**Jenis Karya** : Tugas Akhir

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty Free Right)** atas karya ilmiah saya yang berjudul:

Judul Skripsi/Thesis/Disertasi

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok  
Pada tanggal : XX Januari 2010  
Yang menyatakan

(Luqman Sungkar)



## ABSTRAK

Nama : Luqman Sungkar  
Program Studi : Sistem Informasi  
Judul : Judul Skripsi/Thesis/Disertasi

**@todo**

Tuliskan abstrak laporan disini.

Kata Kunci:

**@todo**

Tuliskan kata kunci yang berhubungan dengan laporan disini

## ABSTRACT

Name : Luqman Sungkar  
Program : Sistem Informasi  
Title : Unknown Title for Final Report/Thesis/Disertation

**@todo**

Write your abstract here.

Keywords:

**@todo**

Write up keywords about your report here.

## DAFTAR ISI

<b>HALAMAN JUDUL</b>	<b>i</b>
<b>LEMBAR PERSETUJUAN</b>	<b>ii</b>
<b>LEMBAR PERNYATAAN ORISINALITAS</b>	<b>iii</b>
<b>LEMBAR PENGESAHAN</b>	<b>iv</b>
<b>KATA PENGANTAR</b>	<b>v</b>
<b>LEMBAR PERSETUJUAN PUBLIKASI ILMIAH</b>	<b>vii</b>
<b>ABSTRAK</b>	<b>viii</b>
<b>Daftar Isi</b>	<b>x</b>
<b>Daftar Gambar</b>	<b>xii</b>
<b>Daftar Tabel</b>	<b>xiii</b>
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	2
1.3 Ruang Lingkup Pengerjaan . . . . .	3
1.4 Tujuan . . . . .	3
1.5 Tahapan Pengerjaan . . . . .	3
1.6 Sistematika Penulisan Laporan . . . . .	4
<b>2 LANDASAN TEORI</b>	<b>6</b>
2.1 <i>Internet of Things</i> . . . . .	6
2.2 <i>Social internet of things</i> . . . . .	7
2.3 ZigBee . . . . .	8
2.4 Gateway . . . . .	9
2.4.1 ZigBee Gateway . . . . .	10
2.5 MQTT . . . . .	11
2.5.1 Sistem <i>Publish-Subscribe</i> . . . . .	11

	xi
2.5.2 Topik dalam MQTT . . . . .	12
<b>3 RANCANGAN IMPLEMENTASI</b>	<b>14</b>
3.1 Satu Persamaan . . . . .	14
3.2 Lebih dari Satu Persamaan . . . . .	14
<b>4 IMPLEMENTASI</b>	<b>16</b>
4.1 thesis.tex . . . . .	16
4.2 laporan_setting.tex . . . . .	16
4.3 istilah.tex . . . . .	16
4.4 hype.indonesia.tex . . . . .	16
4.5 pustaka.tex . . . . .	17
4.6 bab[1 - 6].tex . . . . .	17
<b>5 PENGUJIAN</b>	<b>18</b>
5.1 Mengubah Tampilan Teks . . . . .	18
5.2 Memberikan Catatan . . . . .	18
5.3 Menambah Isi Daftar Isi . . . . .	19
5.4 Memasukan PDF . . . . .	19
5.5 Membuat Perintah Baru . . . . .	23
<b>6 ??</b>	<b>24</b>
<b>7 KESIMPULAN DAN SARAN</b>	<b>25</b>
7.1 Kesimpulan . . . . .	25
7.2 Saran . . . . .	25
<b>LAMPIRAN</b>	<b>1</b>
<b>Lampiran 1</b>	<b>2</b>

## DAFTAR GAMBAR

1	<i>Creative Common License 1.0 Generic</i> . . . . .	v
2	Dokumen Dibuat dengan PDFLatex . . . . .	vi
2.1	Arsitektur <i>Gateway</i> ZigBee[2] . . . . .	10
2.2	Contoh topik dalam MQTT [8] . . . . .	12
2.3	Contoh <i>single level wildcard</i> [8] . . . . .	13
2.4	Contoh <i>multi level wildcard</i> [8] . . . . .	13

## **DAFTAR TABEL**

# BAB 1

## PENDAHULUAN

Pada bagian ini akan dijelaskan latar belakang dari pengerjaan tugas akhir ini, tujuan penulisan, rumusan masalah, ruang lingkup dan batasan pengerjaan, tahapan pengerjaan, dan sistematika penulisan laporan.

### 1.1 Latar Belakang

Penemuan internet beberapa dekade lalu telah merubah cara manusia dalam hal bertukar informasi. Popularitas dari internet terus meningkat hingga pada akhirnya, menurut data dari cisco[5], jumlah perangkat yang terhubung ke internet pada tahun 2010 telah melewati jumlah dari populasi manusia yang ada di Bumi. Lebih jauh lagi, diperkirakan pada tahun 2015 ini jumlah perangkat yang terhubung ke internet akan menjadi dua kali lipat dari jumlah populasi manusia. Hal ini disebabkan oleh teknologi komputer terbenam (komputer yang tidak terlihat keberadaannya) yang akan memicu lahirnya era baru dalam sejarah internet, era *internet of things*.

Saat ini, berbagai perangkat yang menerapkan konsep *internet of things* telah bermunculan di pasaran. Beberapa *platform* yang bertujuan menghubungkan berbagai perangkat *internet of things*-pun mulai bermunculan di pasaran. Berbagai *platform* tersebut memiliki berbagai pendekatan, mulai dari yang menargetkan *end-user* dengan model otomasi, sampai yang menargetkan konsumen *enterprise*. Salah satu pendekatan yang belum banyak terlihat keberadaannya di pasaran adalah pendekatan dengan model sosial media dan menargetkan pada *end-user*. Oleh karena itu, Penulis dan teman-teman Penulis tertarik untuk mengembangkan sebuah *platform internet of things* yang menggunakan model sosial media dan ditargetkan untuk *end-user*.

Untuk membuat sebuah *platform* yang menargetkan pada *end-user*, diperlukan cara untuk menghubungkan berbagai perangkat yang dimiliki konsumen dengan cara yang mudah. Mekanisme pendaftaran ini harus bisa menangani baik perangkat yang sudah beredar dipasaran maupun yang nantinya akan beredar di pasaran. Konsumen juga mungkin akan menginginkan agar perangkat yang mereka miliki walaupun berbeda merek, namun bisa dikendalikan secara bersamaan. Untuk itu, dibutuhkan sebuah *gateway* yang dapat menghubungkan berbagai perangkat berbeda ke *platform* yang diinginkan melalui internet.

Nisrina Luthfiyati dalam tugas akhirnya yang berjudul ”Implementasi ZigBee Coordinator dengan REST Interface”[12] telah berhasil membuat sebuah *coordinator* untuk perangkat berbasis ZigBee yang dapat diakses melalui perintah REST. Fauziah Rahmawati kemudian dalam tugas akhirnya yang berjudul ”Implementasi Home Automation Gateway untuk IOT Cloud Service berbasis ZigBee Network”[14] berhasil melanjutkan tugas akhir Nisrina Luthfiyati dan membuat sebuah *gateway* yang dapat menghubungkan perangkat berbasis ZigBee dengan *Home Automation Profile* ke internet. *Gateway* yang digunakan oleh Fauziah Rahmawati menerapkan konsep *publish-subscribe* dalam melakukan pengiriman data ke internet.

Namun, penulis merasa masih ada yang bisa ditingkatkan dari implementasi *gateway* yang telah dibuat oleh Fauziah, terutama jika akan digunakan pada *platform internet of things* yang akan dibuat oleh Penulis dan teman-teman Penulis yang menargetkan pada *end-user*. Implementasi *gateway* yang telah dibuat oleh Fauziah masih terlalu rumit untuk digunakan dan masih membutuhkan suatu komputer terpisah yang terus menyala untuk bisa digunakan. Sedangkan jika ingin bisa digunakan oleh *end-user*, *gateway* harus bisa mendaftarkan perangkat yang dimiliki konsumen dengan cara yang mudah dan bisa terhubung dengan pengelola perangkat yang ada pada *platform* terkait. *Gateway* harus menyertakan identitas ketika terhubung ke *platform* sehingga dapat dibedakan antar *gateway* yang dimiliki oleh *user* berbeda. *Gateway* juga harus memiliki bentuk yang ringkas dan tidak memerlukan komputer tambahan sehingga tidak perlu ada komputer konsumen yang menyala terus-menerus. Oleh karena itu, Penulis terpikir untuk membuat sebuah implementasi ZigBee *gateway* dengan menggunakan Raspberry Pi untuk menyelesaikan masalah yang disebutkan sebelumnya.

## 1.2 Rumusan Masalah

Masalah yang ingin diselesaikan dalam tugas akhir ini adalah bagaimana membuat sebuah *gateway* yang dapat menghubungkan perangkat yang dimiliki konsumen dengan *platform internet of things* yang menggunakan pendekatan sosial media. *Gateway* harus bisa digunakan oleh *end-user* dengan mudah dan dapat berkomunikasi dengan *platform* yang diinginkan. Secara spesifik, rumusan masalah dalam tugas akhir ini dapat dituangkan ke dalam pertanyaan-pertanyaan berikut:

1. Bagaimana membuat ZigBee *gateway* dengan menggunakan Raspberry Pi
2. Bagaimana membuat *gateway* yang bisa digunakan oleh *end-user* dengan mudah?



3. Bagaimana model komunikasi yang baik agar *gateway* dapat terhubung dengan *platform* berbasis sosial media yang menargetkan pengguna *end-user*?

### 1.3 Ruang Lingkup Pengerjaan

Ruang lingkup implementasi *generic gateway* untuk *platform internet of things* berbasis sosial media dengan target konsumen *end-user* adalah sebagai berikut:

1. *Device* yang akan bisa dihubungkan dibatasi pada perangkat yang menggunakan implementasi ZigBee dengan *profile Light Link*.
2. Implementasi ZigBee *coordinator* yang digunakan adalah implementasi yang telah disediakan oleh pihak dresden, yaitu deConz.
3. Implementasi *gateway* akan berdasarkan pada implementasi *gateway* yang telah dibuat oleh Fauziah Rahmawati[14].
4. *Platform* yang akan digunakan sebagai acuan adalah *platform* yang sedang dikembangkan oleh Penulis dan teman-teman Penulis.
5. Perangkat yang digunakan adalah Raspberry Pi Model B dengan sistem operasi Raspbian.

### 1.4 Tujuan

Tujuan dari tugas akhir ini adalah menghasilkan sebuah *gateway* yang dapat mengirimkan informasi dari perangkat ZigBee ke internet. *Gateway* yang dihasilkan diharapkan dapat mudah digunakan oleh *end-user*. *Gateway* juga dikembangkan dengan berpatokan pada *platform internet of things* yang berbasis sosial media.

### 1.5 Tahapan Pengerjaan

Pengerjaan dari tugas akhir ini akan dilakukan dengan tahapan sebagai berikut:

1. Studi Literatur

Sebelum memulai merancang dan melakukan implementasi, penulis harus memahami beberapa konsep yang berkaitan dengan apa yang ingin penulis buat. Beberapa konsep tersebut diantaranya, *internet of things*, ZigBee,

*MQTT*, *social internet of things*, dan *gateway*. Penulis juga akan mempelajari implementasi *coordinator* yang telah dikerjakan oleh Nisrina Luthfiyati[12] dan implementasi *gateway* yang telah dikerjakan oleh Fauziah Rahmawati[14].

## 2. Analisis dan Perancangan

Setelah melakukan studi literatur, penulis akan menganalisis kebutuhan dari *gateway* yang akan dibuat. Dari hasil analisis tersebut, penulis akan merancang implementasi dari *gateway* tersebut. Hal yang akan ditentukan dalam perancangan ini mencakup skema komunikasi antara *gateway* dengan *platform* yang akan digunakan, skema komunikasi di dalam *gateway*, serta desain pesan yang akan dikirim antara *gateway* dengan *platform* yang akan digunakan.

## 3. Implementasi

Setelah berhasil melakukan perancangan, penulis akan melakukan implementasi sesuai rancangan yang dibuat. Hasil rancangan akan berupa perangkat *gateway* dan implementasi *software*. Pada tahap ini, penulis akan menggunakan *tools* dan SDK yang sudah ada untuk memudahkan proses implementasi

## 4. Uji Coba dan Analisis Hasil

Setelah melakukan implementasi, penulis harus menguji perangkat yang sudah dibuat untuk memastikan apakah perangkat berjalan sesuai dengan kebutuhan yang telah ditentukan. Setelah dilakukan pengujian, penulis akan menganalisis hasil pengujian tersebut

## 5. Penarikan Kesimpulan

Setelah melakukan pengujian terhadap hasil implementasi, penulis dapat menarik kesimpulan dari seluruh kegiatan pengerjaan tugas akhir ini.

# 1.6 Sistematika Penulisan Laporan

Sistematika penulisan laporan tugas akhir ini adalah sebagai berikut:

- Bab 1 PENDAHULUAN

Bab ini berisi latar belakang pengerjaan tugas akhir, rumusan-rumusan

masalah, ruang lingkup pengerjaan, tujuan dari tugas akhir, tahapan pengerjaan yang akan dijalani oleh penulis, dan sistematika dari penulisan laporan ini.

- Bab 2 LANDASAN TEORI

Bab ini akan menjelaskan beberapa konsep yang diperlukan untuk mengerjakan tugas akhir ini. Konsep-konsep tersebut, diantaranya adalah *internet of things*, ZigBee yang meliputi definisi dan jenis perangkat, MQTT beserta mekanisme *publish-subscribe*, dan *gateway*.

- Bab 3 RANCANGAN IMPLEMENTASI

Bab ini akan menjelaskan tentang rancangan dari implementasi *gateway* yang akan dibuat. Pada bab ini juga akan dijelaskan skema komunikasi antara *gateway* dengan *platform* yang akan digunakan.

- Bab 4 IMPLEMENTASI

Bab ini akan menjelaskan tentang implementasi dari *gateway* meliputi konfigurasi *gateway*, dan bagaimana cara menggunakan *gateway* yang dibuat. Pada bab ini juga akan dijelaskan mengenai kode sumber yang diimplementasikan oleh penulis.

- Bab 5 PENGUJIAN

Bab ini akan menjelaskan tentang mekanisme pengujian yang dilakukan penulis terhadap *gateway* yang dibuat. Pada bab ini juga hasil dari pengujian akan ditampilkan dan dilakukan analisis dari hasil pengujian tersebut.

- Bab 6 KESIMPULAN DAN SARAN

Bab ini akan memberikan kesimpulan dari hasil implementasi *gateway* yang dilakukan oleh penulis. Penulis juga akan memberikan saran dan ide yang dapat digunakan untuk pengembangan selanjutnya.

## BAB 2

### LANDASAN TEORI

Bab ini akan menjelaskan beberapa teori dan konsep yang perlu dipahami untuk mengerjakan tugas akhir ini. Beberapa diantaranya adalah *internet of things*, Zig-Bee, gateway, MQTT, dan *social media internet of things*.

#### 2.1 *Internet of Things*

*Internet of Things* adalah suatu konsep dimana berbagai macam benda bisa terhubung ke internet. Benda-benda ini bisa bervariasi, mulai dari benda yang memang pada dasarnya sudah bisa terhubung ke internet seperti *smartphone*, hingga benda seperti lampu, kenop pintu, gembok, sensor-sensor, dan berbagai benda lain. Pada konsep ini, selain dapat terhubung ke internet, benda-benda tersebut juga dapat saling berkomunikasi tanpa bantuan manusia, sehingga membentuk suatu jaringan komunikasi antar benda. Implementasi konsep *internet of things* saat ini sudah sangat beragam, seperti pada bidang *home automation*, bidang kesehatan, bidang retail, dan bidang industri. Beberapa contoh produk populer yang sudah tersedia di pasaran saat ini sebagian besar fokus pada bidang *home automation*, seperti Philips Hue, Nest, WeMo, dan SmartThings.

Istilah *internet of things* pertama kali disebutkan pada tahun 1999 oleh Kevin Ashton dalam konteks manajemen rantai suplai dengan idenya untuk menggunakan RFID pada barang milik P&G [3]. Ia mengatakan bahwa komputer saat ini hampir seluruhnya bergantung pada manusia untuk mendapatkan informasi seperti melalui data yang diketikkan, mengambil gambar dari pemindai, menekan suatu tombol, dan lainnya. Masalahnya adalah, manusia memiliki waktu, perhatian, dan ketepatan yang terbatas. Bayangkan jika komputer bisa mengetahui segalanya tentang suatu hal tanpa bantuan sama sekali dari manusia. Kita bisa menghitung dan melacak segala hal, dan bisa mengurangi pengeluaran, dan kerugian secara besar-besaran. Kita bisa mencapai hal ini dengan membuat agar sebuah komputer dapat melihat, mendengar, dan membaui lingkungan di sekitarnya dengan sendirinya, tanpa batasan data yang harus dimasukkan oleh manusia [3]. Saat ini, istilah *internet of things* tidak hanya terbatas pada konteks manajemen rantai suplai dan mempunyai kemungkinan implementasi yang lebih luas seperti pada bidang kesehatan, transportasi, pendidikan, manajemen energi, manajemen limbah, layanan publik, dan

lainnya.

Saat ini juga sudah ada beberapa definisi mengenai *internet of things*. RFID Group mendefinisikan *internet of things* sebagai jaringan berskala dunia yang terdiri dari objek yang saling terhubung dan memiliki alamat yang unik berdasarkan protokol komunikasi yang standar[10]. Sebuah paper oleh Jayavardhana Gubbi et al.[10] juga mendefinisikan *internet of things* sebagai perangkat yang dapat bergerak dan merasakan keadaan di sekitarnya yang saling terhubung dan dapat berbagi informasi antar berbagai *platform* yang berbeda melalui sebuah framework yang tunggal, yang memberikan gambaran menyeluruh yang dapat membuat munculnya aplikasi inovatif. Hal ini bisa dicapai dengan menggunakan sensor *ubiquitous* yang terus bekerja, analisa data, dan representasi informasi menggunakan *cloud computing*.

Singkatnya, walaupun definisi '*things*' (benda) saat ini sudah berubah, tapi tujuan dari konsep ini menurut Kevin Ashton masih tetap sama, yaitu membuat sebuah mesin agar dapat merasakan apa yang terjadi di sekitar mereka, mengambil informasi dari apa yang terjadi di sekitar mereka, mengolah informasi tersebut, mengirimkannya ke mesin lain, dan kemudian memberikan respons yang sesuai. Semua hal tersebut dilakukan tanpa ada bantuan dari manusia.

## 2.2 *Social internet of things*

*Social internet of things* adalah sebuah konsep yang mencoba menghubungkan konsep *internet of things* dengan konsep sosial. Terdapat beberapa pendekatan mengenai konsep *social internet of things* ini. Luigi Atzori et al dalam papernya [11] menawarkan sebuah konsep yang mencoba memberikan hubungan diantara benda-benda, bukan diantara pemiliknya. Hubungan antara benda dengan benda ini meniru hubungan antara manusia dengan manusia. Sebagai contoh, salah satu hubungan yang paling dasar antara manusia adalah hubungan orang tua dengan anak. Dalam papernya, Luigi Atzori et al menamakan hubungan ini sebagai *parental object relationship*, yaitu hubungan antara benda dengan benda lain yang dibuat dalam periode waktu yang sama dan dibuat oleh perusahaan yang sama. Contoh hubungan lain yang ditawarkan dalam paper ini adalah *co-location object relationship* dan *co-work object relationship* seperti yang dilakukan manusia dalam kehidupannya di suatu tempat yang sama atau ketika sedang dalam pekerjaan. Hubungan tersebut ditentukan ketika beberapa benda berada dalam lokasi yang dekat dalam waktu cukup lama atau secara periodik melakukan hubungan untuk melakukan suatu tugas tertentu. Hubungan lain yang cukup menarik yang disebutkan oleh Luigi Atzori et

al dalam papernya adalah *social object relationship*. Sama seperti hubungan yang dialami oleh manusia ketika berkenalan dengan manusia lain, pada hubungan ini suatu benda juga dapat bertukar profil sosial dengan benda lain. Melalui hal ini, sebuah benda dapat saling bertukar *best practice* untuk menyelesaikan suatu masalah yang sudah dialami oleh "teman"nya.

Contoh pendekatan lain adalah seperti yang dibuat oleh Dominique Guinard et al [7]. Dalam papernya, Dominique menawarkan sebuah konsep untuk memanfaatkan hubungan manusia yang terdapat pada layanan sosial media yang sudah ada di pasaran seperti facebook, twitter, dan lainnya. Melalui konsep ini, seorang pengguna dapat membagikan *resource* dari suatu benda yang dimilikinya ke temannya yang terdapat pada suatu media sosial yang sama. Pembagian *resource* ini bisa hanya membagikan status dari *resource* terkait saja (*read only*) atau juga bisa memberikan orang lain akses untuk melakukan kontrol atau merubah status dari benda terkait.

### 2.3 ZigBee

Zigbee adalah suatu standar yang mendefinisikan sekumpulan protokol komunikasi untuk jaringan nirkabel dengan *data rate* rendah dan jarak yang dekat [6]. ZigBee beroperasi di pita frekuensi 868 MHz, 915 MHz, dan 2.4GHz. *Data rate* maksimal dari ZigBee adalah 250 kilo bit per detik.

Walaupun memiliki *data rate* yang rendah, ZigBee memiliki kelebihan dibandingkan dengan protokol komunikasi lain, yaitu penggunaan daya yang rendah. Hal ini karena memang ZigBee ditargetkan untuk digunakan pada benda yang menggunakan baterai, di mana *data rate* yang dibutuhkan tidak terlalu tinggi, dan biaya yang murah dan daya tahan baterai yang lama adalah kebutuhan utama. Baterai yang digunakan oleh perangkat ZigBee dapat bertahan beberapa tahun sampai perlu diganti. Hal ini dapat dicapai karena pada kebanyakan situasi, total waktu yang digunakan oleh suatu perangkat nirkabel dalam melakukan aktivitas sangat terbatas, sehingga perangkat terkait akan menghabiskan sebagian besar waktunya dalam kondisi hemat daya atau yang dikenal dengan *sleep mode*[6].

Kelebihan lain dari ZigBee adalah topologi jaringannya yang berupa *mesh network*. Ini berarti pengiriman data dari satu perangkat ke perangkat lain bisa tidak dilakukan secara langsung, namun melalui perangkat-perangkat lain sebagai perantara. ZigBee juga memiliki karakteristik *Self-Forming* dan *Self-Healing*. *Self-Forming* berarti sebuah jaringan ZigBee dapat langsung terbentuk ketika perangkat telah aktif tanpa diperlukan supervisi tambahan, sedangkan *Self-Healing* berarti jika

salah satu perangkat ZigBee yang berada dalam rute pengiriman data mati karena suatu hal, jaringan ZigBee akan secara otomatis membuat rute pengiriman data yang baru.

Dalam implementasinya, ZigBee memiliki beberapa profil. Profil-profil ini mendefinisikan bagaimana suatu perangkat harus bekerja. Dalam profil ini disebutkan apa saja fungsi-fungsi yang harus ada ketika akan dilakukan implementasi, dan apa saja atribut-atribut yang harus dimiliki oleh suatu perangkat. Dengan mengikuti profil ini, setiap perangkat yang dibuat oleh perusahaan berbeda akan tetap memiliki perilaku yang sama. Beberapa contoh profil yang telah ditentukan adalah:

- ZigBee Home Automation 1.2
- Light Link 1.0
- Building Automation 1.0
- Retail Services
- Remote Control 2.0
- Health Care 1.0

Namun pada saat tulisan ini dibuat, ZigBee Alliance sebagai pengembang ZigBee telah mengumumkan versi terbaru dari ZigBee, yaitu ZigBee 3.0. Pada ZigBee versi terbaru ini, dilakukan unifikasi dari berbagai profil ZigBee yang telah disebutkan sebelumnya, sehingga hanya ada satu standar ZigBee. Dengan dilakukannya unifikasi ini, diharapkan dapat memudahkan pengembang perangkat untuk mengembangkan perangkat berbasis ZigBee yang mendukung komunikasi antar perangkat dengan jenis berbeda.

## 2.4 Gateway

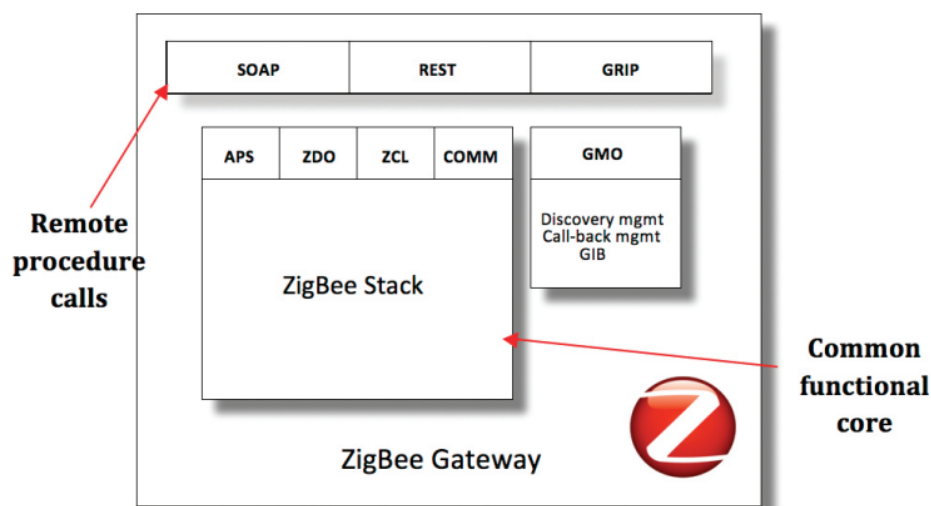
*Gateway* adalah sebuah entitas, baik berupa *hardware*, *software*, maupun gabungan keduanya, yang menyalurkan suatu data dari satu tempat ke tempat lainnya. *Gateway* bisa ada dalam banyak bentuk, seperti di antara beberapa LAN, diantara beberapa jenis jaringan yang berbeda, dan juga diantara beberapa aplikasi. *Gateway* biasanya beroperasi di atas *network layer* dan bisa melibatkan semua *layer*. Inti dari sebuah *gateway* adalah konversi atau perubahan. Sebuah *gateway* akan mengubah pesan yang melewatinya hingga ke *layer* dimana *gateway* tersebut dioperasikan. Tipe *gateway* yang biasa digunakan adalah *application gateway* yang beroperasi di

*application layer*. Sebagai contoh, sebuah *gateway* pesan elektronik dapat mengubah pesan dari suatu format pesan elektronik ke format lainnya. [4].

### 2.4.1 ZigBee Gateway

Seperti dijelaskan sebelumnya, *gateway* berfungsi sebagai penyalur dan pengubah data yang dikirim dari satu tempat ke tempat lainnya. Berbagai macam *gateway* telah dikembangkan sesuai dengan kebutuhan. ZigBee Alliance sebagai pengembang dari protokol ZigBee juga telah mengeluarkan sebuah standar untuk membuat *gateway*. *Gateway* ini dapat digunakan untuk menghubungkan perangkat berbasis ZigBee dengan jaringan lain. Dalam dokumen yang dikeluarkan oleh ZigBee Alliance, dikatakan bahwa perangkat *gateway* ZigBee menyediakan jalur komunikasi menuju sebuah jaringan ZigBee menggunakan *IP based host application* (IPHA) dan sebaliknya dengan menyediakan suatu mekanisme dimana aplikasi eksternal bisa berinteraksi dengan suatu perangkat ZigBee untuk melakukan suatu kontrol atau mengambil suatu data [1].

ZigBee Alliance juga menyertakan desain arsitektur *gateway* dalam dokumen yang dikeluarkannya. Dalam desain arsitektur tersebut, sebuah *gateway* ZigBee dapat mengimplementasikan satu atau lebih protokol untuk melakukan *Remote Procedure Calls* (RPC). Protokol yang disarankan adalah SOAP, REST, dan GRIP. Pemilihan protokol tersebut dapat dilakukan sesuai kebutuhan pengembang [2]. Gambar 2.1 menunjukkan gambaran arsitektur *gateway* ZigBee yang disediakan oleh ZigBee Alliance.



Gambar 2.1: Arsitektur Gateway ZigBee[2]



Seperti terlihat pada gambar 2.1, implementasi *gateway* ZigBee dibagi menjadi dua lapis. Pada lapisan pertama, terdapat sekumpulan fungsi abstraksi yang tidak tergantung pada protokol dan digunakan untuk melakukan fungsi-fungsi terkait perangkat ZigBee dan pengaturan dari *gateway* itu sendiri. Pada lapisan kedua, terdapat sekumpulan protokol RPC yang dapat digunakan untuk berkomunikasi dengan pihak lain. Untuk pihak lain bisa berkomunikasi dengan *gateway*, akan tergantung pada protokol apa yang digunakan pada lapisan kedua ini. *Gateway* kemudian akan mengubah pesan yang diterima di lapisan kedua dan disampaikan kepada lapisan pertama untuk kemudian diproses.

## 2.5 MQTT

MQTT merupakan singkatan dari *Message Queueing Telemetry Transport*. MQTT adalah sebuah protokol pengiriman pesan yang sangat simpel dan ringan, dan di desain untuk perangkat yang memiliki keterbatasan dan *bandwidth* yang rendah. Hal ini membuat MQTT menjadi cocok untuk kebutuhan komunikasi antar mesin yang tidak memerlukan *bandwidth* yang tinggi namun perlu melakukan komunikasi yang cukup sering dan bisa diandalkan [13].

MQTT di desain agar segala kompleksitas mekanisme pengiriman pesan terdapat di sisi *broker*, sehingga dalam menggunakan *client* MQTT dapat dilakukan dengan mudah. MQTT menggunakan konsep *publish-subscribe* dalam melakukan kerjanya. Model *publish-subscribe* yang digunakan adalah berdasarkan topik. Dengan menggunakan topik yang disusun menjadi sebuah hirarki, dapat dimungkinkan untuk suatu perangkat melakukan langganan lebih dari satu topik sekaligus [9].

### 2.5.1 Sistem *Publish-Subscribe*

Konsep sistem *publish-subscribe* (pub-sub) adalah suatu model komunikasi dimana pihak yang tertarik untuk mendapatkan informasi mengenai suatu hal, dapat mendaftarkan ketertarikannya. Proses pendaftaran ketertarikan ini disebut proses *subscription*, dan oleh karena itu pihak yang melakukan proses ini disebut *subscriber*. Pihak yang akan memberikan informasi kemudian melakukan suatu proses yang disebut *publishing*, sehingga pihak yang melakukan proses ini disebut sebagai *publisher*. Selain kedua pihak tersebut, perlu ada suatu pihak yang berperan menjadi penengah, dan bertanggung jawab untuk mengirimkan dan memastikan informasi yang dikirim oleh *publisher* dapat sampai ke *subscriber*. Pihak yang melakukan pekerjaan ini disebut sebagai *broker* [9].

Ada tiga tipe dasar dalam konsep sistem pub-sub ini, yaitu *topic-based* (berdasarkan topik), *type-based* (berdasarkan tipe), dan *content-based* (berdasarkan isi pesan). Pada tipe yang berdasarkan topik, proses *subscription* dan *publishing* hanya dapat dilakukan untuk suatu topik tertentu yang biasanya sudah diketahui terlebih dahulu ketika pengembangan sistem atau aplikasi. Untuk tipe *type-based*, pihak *subscriber* akan memberikan informasi tentang tipe data yang ingin ia *subscribe*, seperti misalnya data penggunaan listrik. Sedangkan untuk tipe yang terakhir yaitu *content-based*, subscriber akan melakukan proses *subscriber* dengan menjelaskan isi pesan seperti apakah yang ingin diterimanya [9].

### 2.5.2 Topik dalam MQTT

Topik dalam MQTT adalah sebuah *string* yang digunakan oleh *broker* MQTT untuk melakukan penyaringan pesan yang akan dikirim kepada *subscriber*. Topik dalam MQTT disusun menjadi suatu hirarki yang terbagi menjadi beberapa level. Setiap level dipisahkan dengan tanda *forward slash* (/). Setiap topik harus memuat minimal satu karakter dan bersifat *case sensitive* [8]. Gambar 2.2 menunjukkan contoh topik dalam MQTT.

Pemisah  
antar level  
dalam topik  
↓  
siot/userA/kamarTidur/lampuMeja

**Gambar 2.2:** Contoh topik dalam MQTT [8]

Dalam membuat topik MQTT, kita juga dapat menggunakan *wildcard*. Terdapat dua macam *wildcard* dalam MQTT, yaitu *single level wildcard* yang ditandai dengan karakter tanda tambah (+) dan *multi level wildcard* yang ditandai dengan karakter tanda pagar (#). *Single level wildcard* dapat digunakan untuk merepresentasikan suatu topik dalam satu level. Gambar 2.3 menunjukkan contoh penggunaan *single level wildcard*. Sedangkan *multi level wildcard* dapat merepresentasikan lebih dari satu level topik, dan oleh karena itu karakter *multi level wildcard* hanya dapat digunakan sebagai karakter terakhir dalam suatu *string* topik dan harus didahului oleh karakter *forward slash* (/). Namun demikian, kita dapat menggunakan *multi level wildcard* sebagai satu-satunya karakter dalam *string* topik yang berarti *subscriber* akan menerima semua pesan yang dikirimkan ke *broker* [8]. Gambar 2.4 menunjukkan contoh penggunaan *multi level wildcard*.

Contoh topik  
yang didaftarkan:

Single level  
wildcard

↓  
siot/userA/+/lampu

Contoh topik  
yang akan diterima pesannya :

- ✓ siot/userA/kamarTidur/lampu
- ✓ siot/userA/Dapur/lampu
- ✗ siot/userA/Dapur/kulkas
- ✗ siot/userB/Dapur/lampu

**Gambar 2.3:** Contoh *single level wildcard* [8]

Contoh topik  
yang didaftarkan:

Multi level  
wildcard

↓  
siot/userA/#

Contoh topik  
yang akan diterima pesannya :

- ✓ siot/userA/kamarTidur/lampu
- ✓ siot/userA/Rumah/Sensor/Temperatur
- ✓ siot/userA/sensor
- ✗ siot/userB/Dapur/lampu

**Gambar 2.4:** Contoh *multi level wildcard* [8]

## BAB 3

### RANCANGAN IMPLEMENTASI

**@todo**

tambahkan kata-kata pengantar bab 1 disini

### 3.1 Satu Persamaan

$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1} \quad (3.1)$$

Persamaan 3.1 diatas adalah persamaan garis. Persamaan 3.1 dan 3.2 sama-sama dibuat dengan perintah `\align`. Perintah ini juga dapat digunakan untuk menulis lebih dari satu persamaan.

$$\underbrace{|\overline{ab}|}_{\text{pada bola } |\overline{ab}| = r} = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + ||(z_b - z_a)^2} \quad (3.2)$$

### 3.2 Lebih dari Satu Persamaan

$$|\overline{a} * \overline{b}| = |\overline{a}| |\overline{b}| \sin \theta \quad (3.3)$$

$$\begin{aligned} \overline{a} * \overline{b} &= \begin{vmatrix} \hat{i} & x_1 & x_2 \\ \hat{j} & y_1 & y_2 \\ \hat{k} & z_1 & z_2 \end{vmatrix} \\ &= \hat{i} \begin{vmatrix} y_1 & y_2 \\ z_1 & z_2 \end{vmatrix} + \hat{j} \begin{vmatrix} z_1 & z_2 \\ x_1 & x_2 \end{vmatrix} + \hat{k} \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} \end{aligned}$$

Pada Persamaan 3.3 dapat dilihat beberapa baris menjadi satu bagian dari Persamaan 3.3. Sedangkan dibawah ini dapat dilihat bahwa dengan cara yang sama, Persamaan 3.4, 3.5, dan 3.6 memiliki nomor persamaannya masing-masing.

$$\int_a^b f(x) dx + \int_b^c f(x) dx = \int_a^c f(x) dx \quad (3.4)$$

$$\lim_{x \rightarrow \infty} \frac{f(x)}{g(x)} = 0 \quad \text{jika pangkat } f(x) < \text{pangkat } g(x) \quad (3.5)$$

$$a^{m^{a^n \log b}} = b^{\frac{m}{n}} \quad (3.6)$$

## BAB 4

### IMPLEMENTASI

**@todo**

tambahkan kata-kata pengantar bab 1 disini

#### 4.1 `thesis.tex`

Berkas ini berisi seluruh berkas Latex yang dibaca, jadi bisa dikatakan sebagai berkas utama. Dari berkas ini kita dapat mengatur bab apa saja yang ingin kita tampilkan dalam dokumen.

#### 4.2 `laporan_setting.tex`

Berkas ini berguna untuk mempermudah pembuatan beberapa template standar. Anda diminta untuk menuliskan judul laporan, nama, npm, dan hal-hal lain yang dibutuhkan untuk pembuatan template.

#### 4.3 `istilah.tex`

Berkas istilah digunakan untuk mencatat istilah-istilah yang digunakan. Fungsinya hanya untuk memudahkan penulisan. Pada beberapa kasus, ada kata-kata yang harus selalu muncul dengan tercetak miring atau tercetak tebal. Dengan menjadikan kata-kata tersebut sebagai sebuah perintah  $\text{\LaTeX}$  tentu akan mempercepat dan mempermudah pengerjaan laporan.

#### 4.4 `hype.indonesia.tex`

Berkas ini berisi cara pemenggalan beberapa kata dalam bahasa Indonesia.  $\text{\LaTeX}$  memiliki algoritma untuk memenggal kata-kata sendiri, namun untuk beberapa kasus algoritma ini memenggal dengan cara yang salah. Untuk memperbaiki pemenggalan yang salah inilah cara pemenggalan yang benar ditulis dalam berkas `hype.indonesia.tex`.

## 4.5 `pustaka.tex`

Berkas `pustaka.tex` berisi seluruh daftar referensi yang digunakan dalam laporan. Anda bisa membuat model daftar referensi lain dengan menggunakan `bibtex`. Untuk mempelajari `bibtex` lebih lanjut, silahkan buka <http://www.bibtex.org/Format>. Untuk merujuk pada salah satu referensi yang ada, gunakan perintah `\cite`, e.g. `\cite{latex.intro}` yang akan akan memunculkan **[latex.intro]**

## 4.6 `bab[1 - 6].tex`

Berkas ini berisi isi laporan yang Anda tulis. Setiap nama berkas e.g. `bab1.tex` merepresentasikan bab dimana tulisan tersebut akan muncul. Sebagai contoh, kode dimana tulisan ini dibuat berada dalam berkas dengan nama `bab4.tex`. Ada enam buah berkas yang telah disiapkan untuk mengakomodir enam bab dari laporan Anda, diluar bab kesimpulan dan saran. Jika Anda tidak membutuhkan sebanyak itu, silahkan hapus kode dalam berkas `thesis.tex` yang memasukan berkas  $\text{\LaTeX}$  yang tidak dibutuhkan; contohnya perintah `\include{bab6.tex}` merupakan kode untuk memasukan berkas `bab6.tex` kedalam laporan.

## BAB 5

### PENGUJIAN

#### @todo

Tambahkan kata-kata pengantar bab 5 disini.

### 5.1 Mengubah Tampilan Teks

Beberapa perintah yang dapat digunakan untuk mengubah tampilan adalah:

- `\f`  
Merupakan alias untuk perintah `\textit`, contoh *contoh hasil tulisan*.
- `\bi`  
**Contoh hasil tulisan.**
- `\bo`  
**Contoh hasil tulisan.**
- `\m`  
*Contohhasiltulisan.*
- `\mc`

*Contohhasiltulisan*

- `\code`  
Contoh hasil tulisan.

### 5.2 Memberikan Catatan

Ada dua perintah untuk memberikan catatan penulisan dalam dokumen yang Anda kerjakan, yaitu:



- `\todo`

Contoh:

**@todo**

Contoh bentuk todo.

- `\todoCite`

Contoh:

**@todo**  
Referensi

### 5.3 Menambah Isi Daftar Isi

Terkadang ada kebutuhan untuk memasukan kata-kata tertentu kedalam Daftar Isi. Perintah `\addChapter` dapat digunakan untuk judul bab dalam Daftar isi. Contohnya dapat dilihat pada berkas `thesis.tex`.

### 5.4 Memasukan PDF

Untuk memasukan PDF dapat menggunakan perintah `\inpdf` yang menerima satu buah argumen. Argumen ini berisi nama berkas yang akan digabungkan dalam laporan. PDF yang dimasukan dengan cara ini akan memiliki header dan footer seperti pada halaman lainnya.

Untitled

Ini adalah berkas pdf yang dimasukan dalam dokumen laporan.

Cara lain untuk memasukan PDF adalah dengan menggunakan perintah `\putpdf` dengan satu argumen yang berisi nama berkas pdf. Berbeda dengan perintah sebelumnya, PDF yang dimasukan dengan cara ini tidak akan memiliki footer atau header seperti pada halaman lainnya.

Untitled

Ini adalah berkas pdf yang dimasukan dalam dokumen laporan.

## 5.5 Membuat Perintah Baru

Ada dua perintah yang dapat digunakan untuk membuat perintah baru, yaitu:

- `\Var`  
Digunakan untuk membuat perintah baru, namun setiap kata yang diberikan akan diproses dahulu menjadi huruf kapital. Contoh jika perintahnya adalah `\Var{adalah}` maka ketika perintah `\Var` dipanggil, yang akan muncul adalah ADALAH.
- `\var`  
Digunakan untuk membuat perintah atau baru.

## BAB 6

??

**@todo**

tambahkan kata-kata pengantar bab 6 disini

## **BAB 7**

### **KESIMPULAN DAN SARAN**

**@todo**

Tambahkan kesimpulan dan saran terkait dengan pekerjaan yang dilakukan.

#### **7.1 Kesimpulan**

#### **7.2 Saran**

## BIBLIOGRAPHY

- [1] ZigBee Alliance. *Network Device Gateway Specification*. ZigBee Alliance. 2011.
- [2] ZigBee Alliance. *Understanding ZigBee Gateway*. ZigBee Alliance. 2010.
- [3] Kevin Ashton. “That ‘internet of things’ thing”. In: *RFiD Journal* 22.7 (2009), pp. 97–114.
- [4] Steve Smith Burnie Blakeley Deborah J. Boyd. *Introduction to Networking Technologies*. IBM Redbooks, 1994.
- [5] Cisco. *The Internet of Things*. URL: <http://share.cisco.com/internet-of-things.html>.
- [6] Shahin Farahani. *ZigBee wireless networks and transceivers*. Newnes, 2011.
- [7] Dominique Guinard, Mathias Fischer, and Vlad Trifa. “Sharing using social networks in a composable web of things”. In: *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on*. IEEE. 2010, pp. 702–707.
- [8] HiveMQ. *MQTT Essentials Part 5: MQTT Topics and Best Practice*. Accessed: 2015-05-20. URL: <http://www.hivemq.com/mqtt-essentials-part-5-mqtt-topics-best-practices>.
- [9] Urs Hunkeler, Hong Linh Truong, and Andy Stanford-Clark. “MQTT-SA publish/subscribe protocol for Wireless Sensor Networks”. In: *Communication systems software and middleware and workshops, 2008. comsware 2008. 3rd international conference on*. IEEE. 2008, pp. 791–798.
- [10] S.Marusic dan M. Palaniswami J. Gubby R.Buyya. “Internet of Things (IoT): A vision, architectural elements, and future directions”. In: *Elsevier* (2013).
- [11] Giacomo Morabito Luigi Atzori Antonio Iera. “SIoT: Giving a Social Structure to the Internet of Things”. In: *IEEE Communications Letters* ().
- [12] Nisrina Luthfiyati. *Implementasi zigbee coordinator dengan REST*.
- [13] MQTT. *Frequently Asked Question*. Accessed: 2015-05-19. URL: [mqtt.org/faq](http://mqtt.org/faq).
- [14] Fauziah Rahmawati. *Implementasi Home Automation Gateway untuk IOT Cloud Service berbasis ZigBee Network*.



# LAMPIRAN

## **LAMPIRAN 1**