

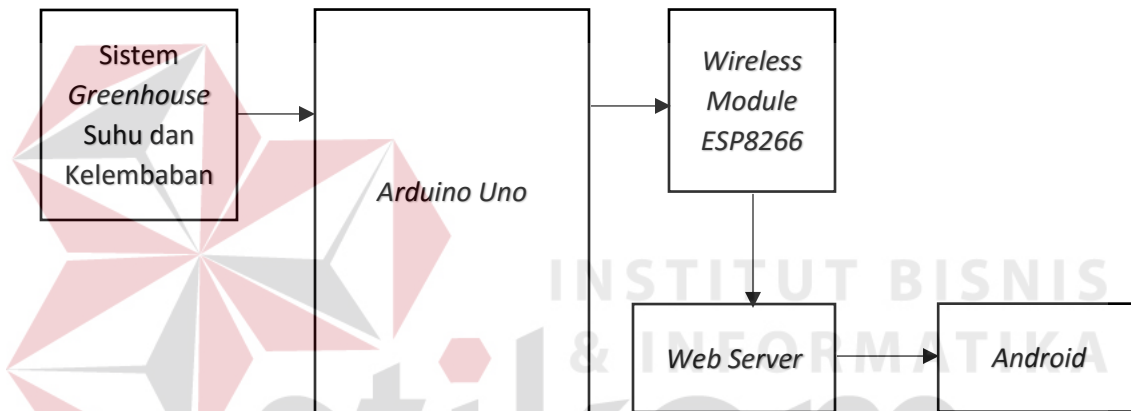
## BAB III

### METODE PENELITIAN

#### 3.1 Rancangan penelitian

Berikut merupakan Blok Diagram *Monitoring* dan Blok Diagram *Setting* pada sistem yang akan dibuat :

##### 3.1.1 Blok Diagram *Monitoring*



Gambar 3.1 Blok Diagram *Monitoring*

Dari gambar Blok Diagram *Monitoring* tersebut terdapat beberapa input dan output yang digunakan antara lain :

a. Input

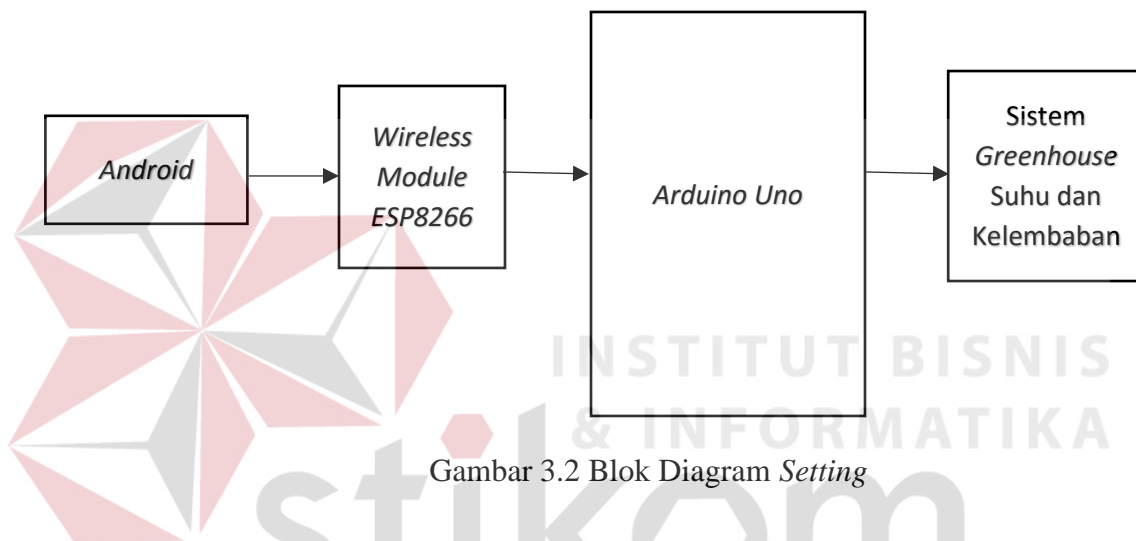
1. Sistem *Greenhouse* Suhu dan kelembaban : Digunakan untuk mendeteksi suhu dan kelembaban pada ruangan *greenhouse*.

b. Output

1. *Wireless Module ESP8266* : Digunakan untuk mengirimkan data suhu dan kelembaban dari Arduino Uno.

2. *Web Server* : Digunakan sebagai perantara untuk *monitoring* data dari *Arduino Uno* ke aplikasi *Android*.
3. *Android* : Digunakan untuk menampilkan data suhu dan kelembaban yang dideteksi oleh sensor suhu dan kelembaban.

### 3.1.2 Blok Diagram *Setting*



Gambar 3.2 Blok Diagram *Setting*

Dari gambar Blok Diagram *Setting* tersebut terdapat beberapa input dan output yang digunakan antara lain :

#### a. Input

1. *Android* : Sebagai pengolah sistem operasi aplikasi yang dibuat bertujuan untuk mengirimkan data suhu dan kelembaban melalui fungsi tombol yang terdapat pada aplikasi kemudian dikirimkan melalui koneksi internet.
2. *Wireless Module ESP8266* : Berfungsi untuk menerima data suhu dan kelembaban dari aplikasi *Android*.

b. Output

1. Arduino Uno : Menerima data yang disalurkan oleh *Wireless Module ESP8266* kemudian mengubah *set point* program yang sudah ditentukan untuk suhu dan kelembaban, *set point* lama akan digantikan set point baru sesuai dengan data yang dikirimkan oleh *Android*.
2. Sistem *Greenhouse* Suhu dan kelembaban : Sebagai pengaturan suhu dan kelembaban pada *greenhouse* dan untuk pengondisian sesuai dengan lingkungan.

### 3.2 Prosedur Penelitian

Adapun langkah-langkah yang dilakukan dalam melakukan penelitian ini dibagi menjadi beberapa bagian yaitu :

1. Studi Literatur

Merupakan langkah yang bertujuan untuk mencari teori sehingga membantu dalam pembuatan sistem. Langkah ini dilakukan dengan metode wawancara pada dosen dan membaca literatur yang berasal dari internet maupun buku-buku yang ada.

2. Perancangan dan Pembuatan Perangkat Lunak

Untuk pembuatan aplikasi berbasis *Android* menggunakan App Inventor.

3. Pengujian Sistem

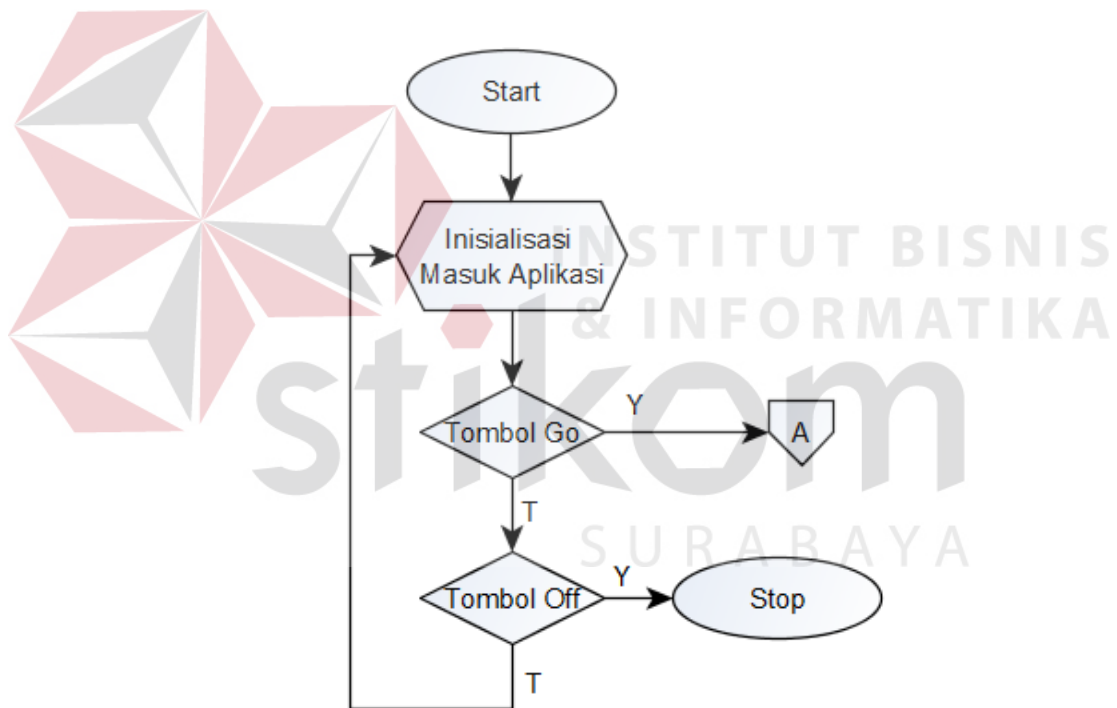
Pengujian ini dilakukan dengan tujuan sistem berjalan dengan sempurna sesuai dengan keinginan. Pengujian yang dilakukan adalah pengujian *hardware* dan

*software*. Ketika mengalami kesalahan pada pengujian, maka sistem akan diperbaiki sampai berjalan sesuai dengan keinginan.

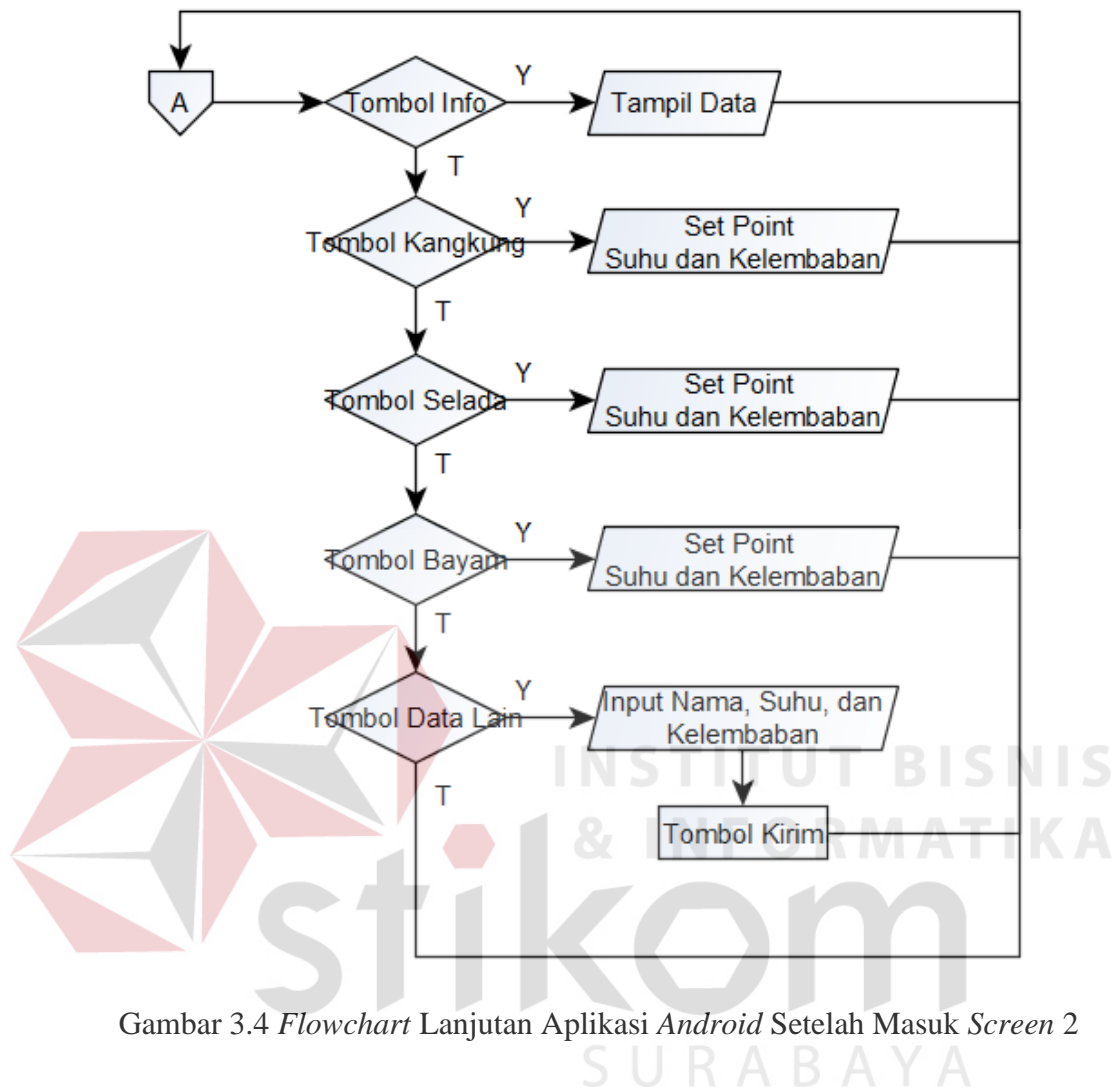
#### 4. Penyusunan Laporan

Penyusunan laporan ini dilakukan setelah semua prosedur penelitian selesai dilakukan. Pelaporan ini dilakukan secara mendetail agar dapat dijadikan literatur bagi yang ingin mengembangkannya.

### 3.3 Flowchart Aplikasi Android



Gambar 3.3 *Flowchart* Pertama Masuk Aplikasi *Android Screen 1*



Gambar 3.4 Flowchart Lanjutan Aplikasi Android Setelah Masuk Screen 2

Dari gambar 3.3 dan 3.4 Flowchart aplikasi Android di atas dapat dijelaskan cara kerja saat aplikasi dijalankan yaitu sebagai berikut :

- a. Start : Menjalankan program/aplikasi pertama kali
- b. Inisialisasi masuk aplikasi : Menginisialisasi tampilan awal pada saat masuk aplikasi, mendeteksi terdapat tombol apa saja, diantaranya yaitu Tombol Go, dan Tombol Off.

- c. Tombol Go : Tombol ini berfungsi untuk masuk kedalam *screen* berbeda yang terdapat beberapa tombol yaitu Tombol Info Suhu Dan Kelembaban, Tombol Input Data Kangkung, Tombol Input Data Selada, Tombol Input Data Bayam, Tombol Input Data Tanaman Lain.
- d. Tombol Off : Tombol ini berfungsi untuk menghentikan proses aplikasi dan akan keluar dari aplikasi.
- e. Tombol Info Suhu Dan Kelembaban : Tombol ini berfungsi untuk menampilkan data suhu dan kelembaban secara real time sesuai dengan sensor suhu dan kelembaban yang terdapat pada Arduino.
- f. Tombol Input Data Kangkung : Saat tombol ini ditekan, data suhu dan kelembaban akan dikirimkan pada Arduino sesuai dengan data suhu dan kelembaban yang sudah diatur pada Android yaitu suhu sekitar 24-26<sup>0</sup> celcius dan kelembaban lebih dari 60%.
- g. Tombol Input Data Selada : Tombol ini berfungsi untuk mengirimkan data suhu dan kelembaban tanaman selada dengan suhu 15-20<sup>0</sup> celcius dan kelembaban sekitar 58,17-75,5%.
- h. Tombol Input Data Bayam : Tombol ini berfungsi untuk mengirimkan data suhu dan kelembaban tanaman bayam dengan suhu 16-20<sup>0</sup> celcius dan kelembaban sekitar 40-60%.
- i. Tombol Input Data Tanaman Baru : Tombol ini saat ditekan akan menampilkan *screen* baru yang berfungsi untuk memasukkan data tanaman lain yang berbeda sesuai yang diinginkan oleh pengguna/user. Pengguna/user dapat memasukkan

nama tanaman, suhu tanaman, dan kelembaban tanaman sesuai keinginan dan kemudian data akan disimpan serta data tersebut akan dikirimkan ke Arduino.

### 3.4 Perancangan Aplikasi *Android Greenhouse*

#### 3.4.1 Pembuatan Desain Aplikasi

Pengerjaan tugas akhir ini diawali dengan pembuatan desain aplikasi *Android* yang akan digunakan untuk Monitoring dan Setting greenhouse. Pembuatan aplikasi *Android* ini menggunakan software berbasis *web* yaitu App Inventor yang diharuskan menggunakan koneksi internet untuk mengaksesnya. Desain aplikasi ini terdiri atas 7 *screen* yang memiliki desain dan fungsi masing-masing.

##### 3.4.1.1 Screen 1

Saat pertama kali memulai aplikasi tampilan desain yang dibuat yaitu *Screen 1*, berfungsi untuk menjadi tampilan pembuka. Dapat dilihat pada gambar 3.5 yaitu desain *Screen 1* dibawah ini.



Gambar 3.5 Desain *Screen 1* Aplikasi

Dari gambar 3.4 diatas dapat dibuat dengan memakai *components* yang terdapat pada App Inventor. Berikut ini adalah list *components* yang digunakan untuk membuat *Screen 1* dapat dilihat pada tabel 3.1 dibawah ini :

Tabel 3.1 *List Components Screen 1*

Components	Properties	Keterangan
Screen 1	AlignHorizontal	Center
	AlignVertical	Top
	AppName	greenhouse
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S - GREENHOUSE
Horizontal Arangement 1	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
	FontBold	√
	FontItalic	√
Label (judul)	FontSize	20
	Text	MONITORING DAN SETTING GREENHOUSE BERBASIS ANDROID
	TextAlignment	Left
	TextColor	Magenta
	Image	logo_stikom_warna.png
Horizontal Arangement 3	AlignHorizontal	Left
	AlignVertical	Top
	Height	25 Percent
	Width	75 Percent
	Image	logo_stikom_warna.png
Horizontal Arangement 4	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 5	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 6	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Horizontal Arangement 9	AlignHorizontal	Left
	AlignVertical	Top
	Height	40 Pixels
	BackgroundColor	None
	AlignHorizontal	Left



Horizontal Arangement 7	AlignVertical	Center
	Width	Fill Parent
Button (keluar)	FontBold	√
	FontItalic	√
	FontSize	14.0
	Height	45 Pixels
	Width	45 Pixels
	Image	turn_off_button-icon.gif
	TextAlignment	Center
Horizontal Arangement 8	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	None
	Height	20 Pixels
	Width	Fill Parent
Button (masuk)	BackgroundColor	Green
	FontBold	√
	FontItalic	√
	FontSize	14.0
	Height	45 Pixels
	Width	45 Pixels
	Image	GOIcon.jpg
	Shape	Rounded
Label (Notif)	TextAlignment	Center
	BackgroundColor	None
	FontBold	√
	FontSize	14.0
	FontTypeface	Monospace
	TextAlignment	Left
WebView1	(Default)	(Default)
TinyDB1	(Default)	(Default)
Web1	(Default)	(Default)

### 3.4.1.2 Screen 2

Pada *screen 2* akan digunakan banyak tombol yang berfungsi untuk memberikan akses pada *screen-screen* yang lainnya. Dalam *screen* ini juga terdapat tombol untuk mengirimkan data suhu dan kelembaban yang dikirimkan ke *wireless module ESP8266* melalui *web server* serta koneksi internet yaitu tombol Input Data

Kangkung, Input Data Selada, dan Input Data Bayam, serta juga Input Tanaman Lain untuk menginputkan data secara manual sesuai keinginan *user* atau pengguna aplikasi, tetapi untuk menggunakan input data manual akan masuk pada *screen 7*. Tombol Info Suhu dan Kelembaban Ruangan yaitu tombol yang ketika ditekan akan masuk pada *screen 3* sebagai monitoring suhu dan kelembaban yang mendapatkan data dari sensor DHT11 yang dikirimkan oleh *wireless module ESP8266* melalui koneksi internet ke *web server* dan kemudian akan dikirimkan ke aplikasi *Android*. Desain *screen 2* dapat dilihat dari gambar 3.5 dibawah ini.



Gambar 3.6 Desain *Screen 2* Aplikasi

Dari gambar desain *screen 2* tersebut dapat dilihat bahwa pada *screen* ini terdapat banyak fungsi-fungsi tombol. *Components* yang digunakan dalam App

Inventor juga akan menggunakan banyak tombol. Berikut list *components* yang digunakan untuk membuat *screen 2* pada tabel 3.2 dibawah ini.

Tabel 3.2 *List Components Screen 2*

Components	Properties	Keterangan
Screen 1	AlignHorizontal	Center
	AlignVertical	Top
	AppName	greenhouse
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S - GREENHOUSE
Horizontal Arrangement 1	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (judul)	FontBold	√
	FontItalic	√
	FontSize	20
	Text	MONITORING DAN SETTING GREENHOUSE BERBASIS ANDROID
	TextAlignment	Left
	TextColor	Magenta
	TextColor	Magenta
Horizontal Arrangement 2	AlignHorizontal	Center
	AlignVertical	Bottom
	Width	Fill Parent
Horizontal Arrangement 3	AlignHorizontal	Left
	AlignVertical	Top
	Height	25 Percent
	Width	75 Percent
	Image	logo_stikom_warna.png
Horizontal Arrangement 4	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arrangement 5	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Label (IPadd)	FontBold	√
	FontSize	14
	FontTypeface	Sans Serif
	Height	Fill Parent
	Text	IP Address :
	TextAlignment	Left
	TextColor	Dark Gray

Text Box 1	FontSize	20
	Height	Fill Parent
	Image	Left
	TextAlignment	Black
Button (ok)	FontSize	14.0
	Height	45 Pixels
	Width	45 Pixels
	Image	ad.jpg
	TextAlignment	Center
Button (hapus)	FontSize	14.0
	Height	45 Pixels
	Width	45 Pixels
	Image	BckGnd-ClearImage.png
	TextAlignment	Center
Horizontal Arangement 6	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (IPtersimpan)	FontBold	√
	FontSize	14.0
	BackgroundColor	None
	FontTypeface	Sans Serif
	Text	IP Tersimpan :
Label (IP)	TextColor	Dark Gray
	BackgroundColor	None
	FontSize	20
	TextAlignment	Left
Horizontal Arangement 9	TextAlignment	Magenta
	AlignHorizontal	Left
	AlignVertical	Top
	Height	40 Pixels
Horizontal Arangement 7	BackgroundColor	None
	AlignHorizontal	Left
	AlignVertical	Center
	Width	Fill Parent
Button (keluar)	FontBold	√
	FontItalic	√
	FontSize	14.0
	Height	45 Pixels
	Width	45 Pixels
	Image	turn_off_button-icon.gif
	TextAlignment	Center
Horizontal Arangement 8	AlignHorizontal	Left
	AlignVertical	Top

	BackgroundColor	None
	Height	20 Pixels
	Width	Fill Parent
Button (masuk)	BackgroundColor	Green
	FontBold	√
	FontItalic	√
	FontSize	14.0
	Height	45 Pixels
	Width	45 Pixels
	Image	GOIcon.jpg
	Shape	Rounded
	TextAlignment	Center
Label (Notif)	BackgroundColor	None
	FontBold	√
	FontSize	14.0
	FontTypeface	Monospace
	TextAlignment	Left
	TextColor	Magenta
WebView1	(Default)	(Default)
TinyDB1	(Default)	(Default)
Web1	(Default)	(Default)

### 3.4.1.3 Screen 3

Desain pada *screen 3* digunakan untuk *monitoring* data suhu dan kelembaban yang terbaca oleh sensor suhu dan kelembaban yang terhubung pada Arduino, kemudian data tersebut dipancarkan oleh *wireless module ESP8266* menuju *web server* yang akan ditangkap oleh *Android* dan akan menampilkan data tersebut pada *screen 3*. Berikut tampilan desain *screen 3* pada gambar 3.6.



Gambar 3.7 Desain *Screen 3* Aplikasi

Desain *screen 3* diatas tampak kosong karena untuk menampilkan data suhu dan kelembaban setelah menerima data dari sensor. Data yang akan muncul secara *Real-Time* sesuai waktu *delay* yang telah ditentukan, misalkan data akan diberi *delay* 1 detik maka data secara *Real-Time* muncul setiap 1 detik sekali. Dan untuk membuat desain tersebut membutuhkan *component* sebagai berikut :

Tabel 3.3 *List Components Screen 3*

Components	Properties	Keterangan
Screen 3	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	White
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S - GREENHOUSE
Label (judul)	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	16

	Text	Tampilan Informasi Suhu dan Kelembaban :
	TextAlignment	Left
	TextColor	Black
Label (tampilan)	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	14.0
	Height	Fill Parent
	Width	Fill Parent
	TextAlignment	Left
	TextColor	Magenta
Clock1	Time Interval	1000

#### 3.4.1.4 Screen 4, Screen 5, dan Screen 6

Pada *Screen 4*, *Screen 5*, dan *Screen 6* ini memiliki fungsi yang sama yaitu hanya berfungsi sebagai informasi tentang nama latin tanaman kangkung, tanaman selada, tanaman bayam, suhu tanaman kangkung, tanaman selada, tanaman bayam dan kelembaban tanaman kangkung, tanaman selada, tanaman bayam. Ditampilkan dengan menggunakan label. Berikut adalah tampilan desain *Screen 4*, *Screen 5*, dan *Screen 6*.



Gambar 3.8 Desain *Screen 4* Aplikasi

Gambar desain diatas adalah *screen 4* yang menunjukkan informasi tentang tanaman kangkung. Daftar dari *components* untuk membuatnya yaitu sabagai berikut.

Tabel 3.4 *List Components Screen 4*

Components	Properties	Keterangan
Screen 4	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	White
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S - GREENHOUSE
Horizontal Arangement 1	AlignHorizontal	Center
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 2	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	None
Label1	Width	Fill Parent
	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Nama Latin :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 3	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label2	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Ipomoea aquatic Forsk.
	TextAlignment	Left
	TextColor	Magenta
Horizontal Arangement 4	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 5	AlignHorizontal	Left
	AlignVertical	Top



	Width	Fill Parent
Label3	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Nama Latin :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arrangement 6	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label4	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	24 – 26 Derajat Celcius
	TextAlignment	Left
	TextColor	Magenta
Horizontal Arrangement 7	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arrangement 8	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label5	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Kelembaban :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arrangement 9	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label6	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	+60%
	TextAlignment	Left
	TextColor	Magenta



Gambar 3.9 Desain *Screen 5* Aplikasi

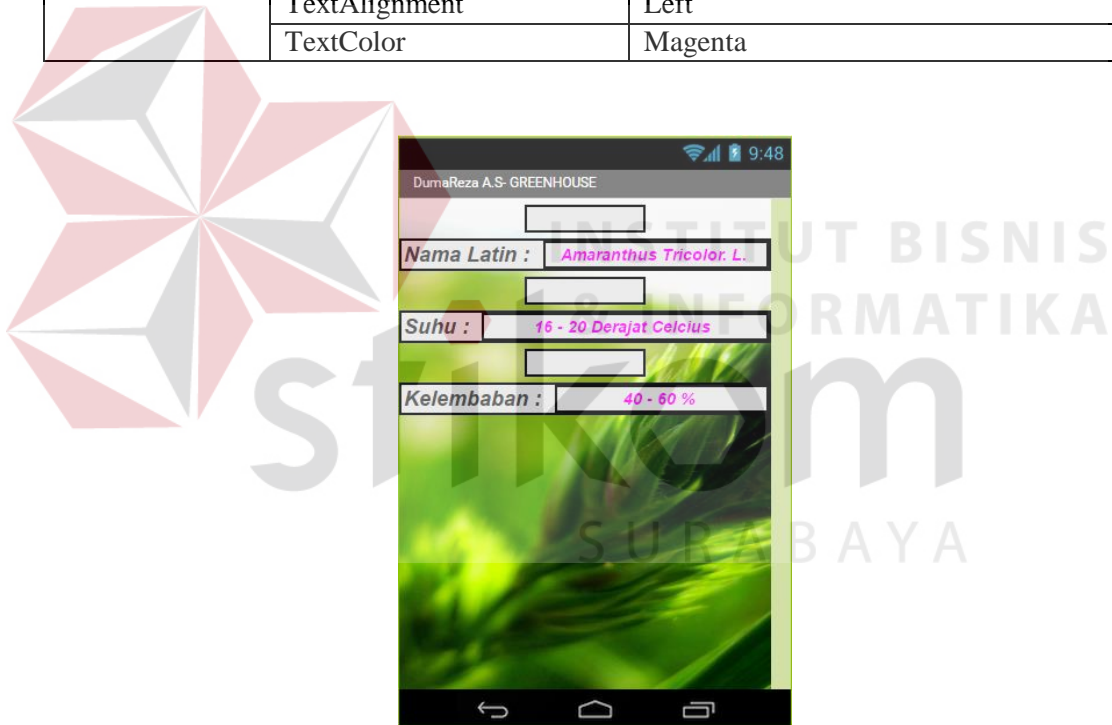
*Screen 5* ini untuk memberikan informasi kepada *user* agar mengetahui nama latin, suhu, dan kelembaban dari tanaman selada. Berikut daftar *component* yang dibutuhkan untuk membuat *screen 5* :

Tabel 3.5 *List Components Screen 5*

Components	Properties	Keterangan
Screen 5	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	White
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S - GREENHOUSE
Horizontal Arrangement 1	AlignHorizontal	Center
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arrangement 2	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	None
	Width	Fill Parent
Label1	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20

	Text	Nama Latin :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 3	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label2	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Lactuca sativa L.
	TextAlignment	Left
Horizontal Arangement 4	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 5	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label3	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Nama Latin :
	TextAlignment	Left
Horizontal Arangement 6	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label4	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	15 – 20 Derajat Celcius
	TextAlignment	Left
Horizontal Arangement 7	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 8	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
	BackgroundColor	None

Label5	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Kelembaban :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arrangement 9	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label6	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	58.17% - 75.5%
	TextAlignment	Left
	TextColor	Magenta



Gambar 3.10 Desain *Screen 6* Aplikasi

*Screen 6* memberikan informasi nama latin, suhu, dan kelembaban dari tanaman bayam seperti pada gambar 3.10. Pada tabel 3.6 adalah daftar *component* yang dibutuhkan untuk membuat *screen 6*.

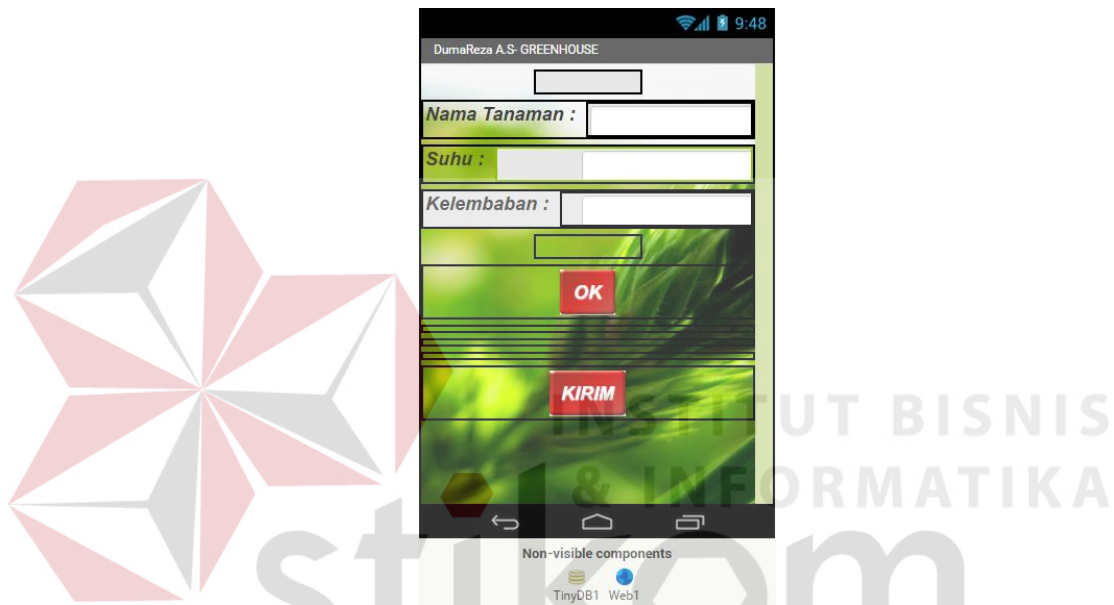
Tabel 3.6 List Components Screen 6

Components	Properties	Keterangan
Screen 6	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	White
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S – GREENHOUSE
Horizontal Arangement 1	AlignHorizontal	Center
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 2	AlignHorizontal	Left
	AlignVertical	Top
	BackgroundColor	None
	Width	Fill Parent
Label1	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Nama Latin :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 7	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label4	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Amaranthus Tricolor L.
	TextAlignment	Left
	TextColor	Magenta
Horizontal Arangement 3	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 4	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label2	BackgroundColor	None
	FontBold	√
	FontItalic	√
	Text	Nama Latin :

	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arrangement 8	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label5	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	16 – 20 Derajat Celcius
	TextAlignment	Left
	TextColor	Magenta
Horizontal Arrangement 5	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arrangement 6	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label3	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Kelembaban :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arrangement 9	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Label6	BackgroundColor	None
	FontBold	√
	FontItalic	√
	FontSize	20
	Text	40% - 60%
	TextAlignment	Left
	TextColor	Magenta

### 3.4.1.5 Screen 7

Screen 7 ini berfungsi untuk memasukkan data suhu dan kelembaban sesuai yang diinginkan user selain 3 tanaman yang telah ditentukan diatas. Untuk memasukkan data, user memasukkan data secara manual dengan mengetikkan nama tanaman, suhu tanaman, dan kelembaban tanaman.



Gambar 3.11 Desain *Screen 7* Aplikasi

Dari gambar 3.11 diatas terdapat textbox untuk memasukkan nama tanaman sesuai keinginan user. Kemudian terdapat tombol OK yang berfungsi untuk menyimpan data yang telah dimasukkan kedalam database sedangkan tombol KIRIM berfungsi setelah data tersimpan didalam database kemudian tombol KIRIM akan mengirimkan data tersebut. Untuk membuat desain screen 7 dapat menggunakan *component* pada tabel 3.7 berikut.

Tabel 3.7 List Components Screen 7

Components	Properties	Keterangan
Screen 7	AlignHorizontal	Center
	AlignVertical	Top
	BackgroundColor	White
	BackgroundImage	hijau.jpg
	Title	Duma Reza A.S – GREENHOUSE
Horizontal Arangement 1	AlignHorizontal	Center
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 2	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (namatanaman)	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Nama Tanaman :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 5	AlignHorizontal	Right
	AlignVertical	Top
	Width	Fill Parent
TextBoxnama	FontBold	√
	FontItalic	√
	FontSize	20
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 3	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (suhu)	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Suhu :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 4	AlignHorizontal	Right
	AlignVertical	Top
	Width	Fill Parent
TextBoxsuhu	FontBold	√
	FontItalic	√
	FontSize	20



	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 6	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (kelembaban)	FontBold	√
	FontItalic	√
	FontSize	20
	Text	Kelembaban :
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 7	AlignHorizontal	Right
	AlignVertical	Top
	Width	Fill Parent
TextBoxlembab	FontBold	√
	FontItalic	√
	FontSize	20
	TextAlignment	Left
	TextColor	Dark Gray
Horizontal Arangement 8	AlignHorizontal	Left
	AlignVertical	Top
	Height	20 Pixels
Horizontal Arangement 9	AlignHorizontal	Center
	AlignVertical	Top
	Width	Fill Parent
Button (ok)	FontBold	√
	FontItalic	√
	FontSize	20
	Height	45 Pixels
	Width	55 Pixels
	Image	k9387944.jpg
	Shape	Rounded
	Text	OK
	TextAlignment	Center
TextColor	White	
Horizontal Arangement 10	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (simpannama)	FontBold	√
	FontItalic	√
	FontSize	16
	TextAlignment	Left
	TextColor	Magenta

Horizontal Arrangement 11	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (simpansuhu)	FontBold	√
	FontItalic	√
	FontSize	16
	TextAlignment	Left
	TextColor	Magenta
Horizontal Arrangement 12	AlignHorizontal	Left
	AlignVertical	Top
	Width	Fill Parent
Label (simpanlembab)	FontBold	√
	FontItalic	√
	FontSize	16
	TextAlignment	Left
	TextColor	Magenta
Horizontal Arrangement 13	AlignHorizontal	Center
	AlignVertical	Center
	Width	Fill Parent
Button ( kirim )	FontBold	√
	FontItalic	√
	FontSize	20
	Height	45 Pixels
	Width	55 Pixels
	Image	k9387944.jpg
	Shape	Rounded
	Text	KIRIM
	TextAlignment	Center
	TextColor	White
WebView1	(Default)	(Default)
TinyDB1	(Default)	(Default)
Web1	(Default)	(Default)

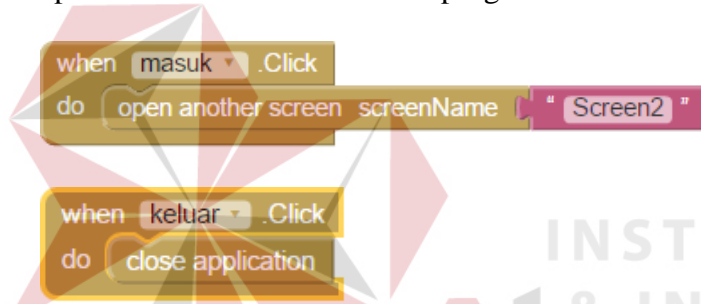
### 3.4.2 Pembuatan *Code Blocks / Program*

Setelah pembuatan desain pada setiap *screen* sudah selesai selanjutnya akan dilakukan pembuatan *Blocks* yaitu istilah pembuatan program yang terdapat pada App Inventor. Bentuk *code program* pada App Inventor berbentuk *puzzle* yaitu hanya

dengan cara *drag* untuk membuat program didalamnya. Program tersebut akan dibuat pada beberapa *screen* diantaranya yaitu *screen 1*, *screen 2*, *screen 3*, dan *screen 7*.

#### 3.4.2.1 *Blocks Screen 1*

*Block* pada *screen 1* digunakan sebagai tampilan awal pada saat pertama kali membuka aplikasi. Pada *screen 1* ini terdapat 2 fungsi tombol yaitu tombol “Go” dan tombol “Off”. Fungsi kedua tombol tersebut sangat sederhana, tombol “Go” berfungsi untuk masuk pada *screen 2* sedangkan tombol “Off” berfungsi untuk keluar dari aplikasi. Berikut adalah *blocks* program kedua tombol tersebut.

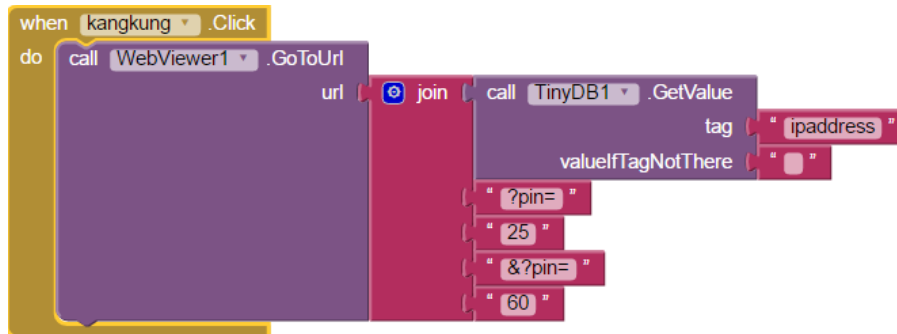


*Blocks masuk.Click* yaitu sebagai tombol “Go” yang berfungsi ketika tombol ditekan maka akan masuk pada “Screen2”. *Blocks keluar.Click* yaitu sebagai tombol “Off” yang berfungsi ketika tombol ditekan maka aplikasi akan keluar atau menutup aplikasi.

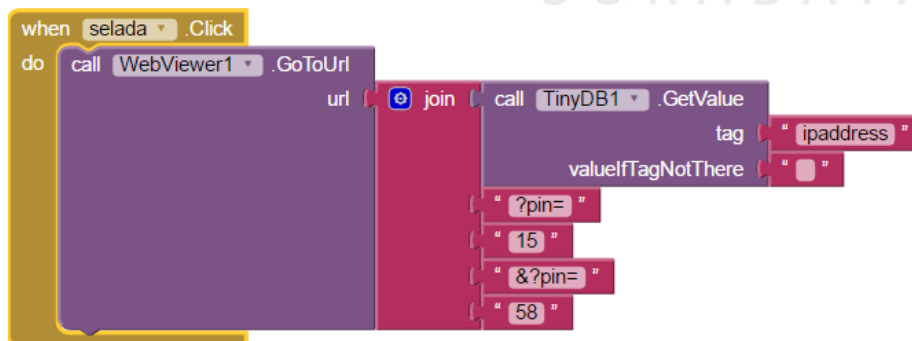
#### 3.4.2.2 *Blocks Screen 2*

Program *blocks* pada *screen 2* ini memiliki beberapa fungsi yaitu sebagai jembatan untuk masuk pada *screen-screen* yang lain dan untuk melakukan pengiriman data *setting* yang dikirimkan pada *web server* dan kemudian akan diolah lebih lanjut

oleh *Wireless Module ESP8266* dan *Arduino Uno*. Data yang dikirim yaitu data suhu dan kelembaban tanaman kangkung, selada, dan bayam.

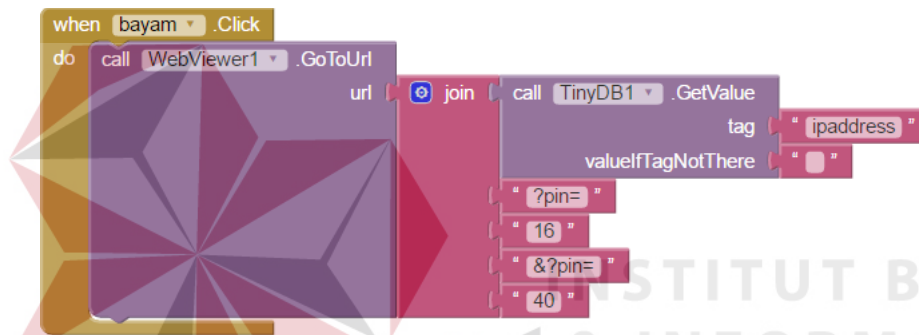


*Block* diatas berfungsi untuk mengirimkan data suhu dan kelembaban kangkung ketika *button kangkung.Click* ditekan oleh *user*, maka akan memanggil **WebView1.GoToUrl** yaitu untuk menampung url yang ingin dikirimkan pada koneksi *wireless*. Disini url yang akan dikirimkan yaitu *IP Address* dengan cara memanggil **TinyDB1.GetValue** karena *IP Address* sudah tersimpan pada *database* pada saat proses memasukkan *IP Address* di *screen 1* dengan *tag* "ipaddress". Kemudian data yang dikirimkan yaitu "?pin=" saat masuk pada *arduino*, maka akan mengirimkan data suhu sebesar 25 derajat dan data kelembaban sebesar 60%.



Sama dengan *button kangkung block* diatas berfungsi untuk mengirimkan data suhu dan kelembaban kangkung ketika *button selada.Click* ditekan

oleh *user*, maka akan memanggil **WebView1.GoToUrl** yaitu untuk menampung url yang ingin dikirimkan pada koneksi *wireless*. Disini url yang akan dikirimkan yaitu *IP Address* dengan cara memanggil **TinyDB1.GetValue** karena *IP Address* sudah tersimpan pada *database* pada saat proses memasukkan *IP Address* di *screen* 1 dengan tag “*ipaddress*”. Kemudian data yang dikirimkan yaitu “*?pin=*” saat masuk pada arduino, maka akan mengirimkan data suhu sebesar 15 derajat dan data kelembaban sebesar 58%.



Sama dengan *button* kangkung dan sedala sebelumnya, *block* diatas berfungsi untuk mengirimkan data suhu dan kelembaban kangkung ketika *button bayam.Click* ditekan oleh *user*, maka akan memanggil **WebView1.GoToUrl** yaitu untuk menampung url yang ingin dikirimkan pada koneksi *wireless*. Disini url yang akan dikirimkan yaitu *IP Address* dengan cara memanggil **TinyDB1.GetValue** karena *IP Address* sudah tersimpan pada *database* pada saat proses memasukkan *IP Address* di *screen* 1 dengan tag “*ipaddress*”. Kemudian data yang dikirimkan yaitu “*?pin=*” saat masuk pada arduino, maka akan mengirimkan data suhu sebesar 16 derajat dan data kelembaban sebesar 40%.

```

when info .Click
do open another screen screenName " Screen3 "

```

*Block* diatas berfungsi untuk membuka *screen 3* ketika *button info.Click* ditekan. *Screen 3* berfungsi untuk menampilkan data suhu dan kelembaban yang diterima dari Arduino Uno melalui koneksi *wireless module ESP8266*.

```

when infokangkung .Click
do open another screen screenName " Screen4 "

```

*Block* ini ketika **infokangkung.Click** ditekan maka akan berpindah ke *screen 4*. Dimana *screen 4* berfungsi untuk menampilkan informasi mengenai tanaman kangkung yaitu nama latin, suhu, dan kelembaban.

```

when infojelada .Click
do open another screen screenName " Screen5 "

```

*Block* ini ketika **infojelada.Click** ditekan maka akan berpindah ke *screen 5*. Dimana *screen 5* berfungsi untuk menampilkan informasi mengenai tanaman selada yaitu nama latin, suhu, dan kelembaban.

```

when infobayam .Click
do open another screen screenName " Screen6 "

```

*Block* ini ketika **infobayam.Click** ditekan maka akan berpindah ke *screen 6*. Dimana *screen 6* berfungsi untuk menampilkan informasi mengenai tanaman bayam yaitu nama latin, suhu, dan kelembaban.

```

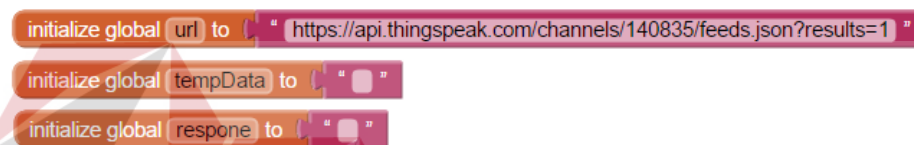
when Screen2 .BackPressed
do close screen

```

*Block* diatas berfungsi pada saat **Screen2.BackPressed** ditekan maka *screen 2* akan keluar dan kembali pada *screen 1*.

### 3.4.2.3 *Blocks Screen 3*

Pada *screen 3* akan dibuat *blocks program* yang berfungsi untuk *monitoring* data suhu dan kelembaban yang telah tersimpan pada *web server*. Data tersebut yang sebelumnya telah dikirim oleh *Wireless Module ESP8266* yang diambil dari *Arduino Uno* dan sensor *DHT11*. Berikut *block program* pada *screen 3* dan penjelasannya.

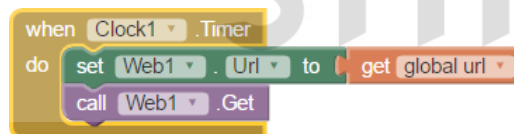


```

initialize global url to " https://api.thingspeak.com/channels/140835/feeds.json?results=1 "
initialize global tempData to ""
initialize global response to ""

```

*Blocks program* diatas ini adalah inisialisasi variabel global *url* agar dapat dengan mudah diakses pada *blocks program* lainnya. *Url* yang dimasukkan yaitu untuk mengambil data suhu dan kelembaban dari *web server*. Untuk variabel **tempData** dan **response** digunakan untuk melakukan parsing data.

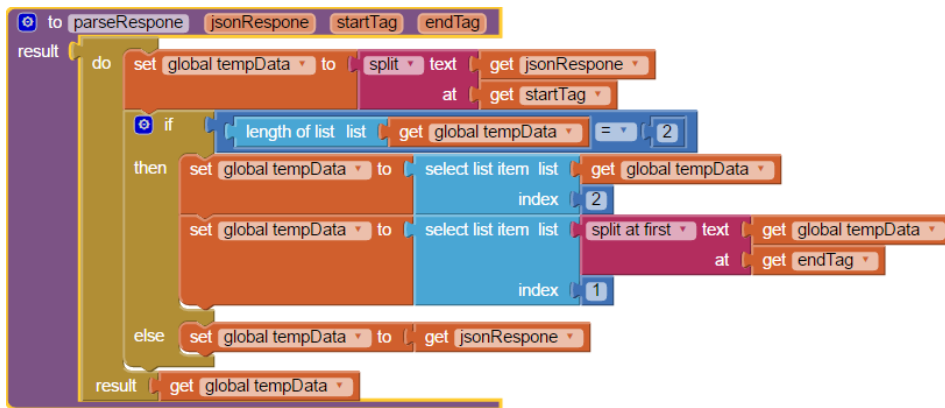


```

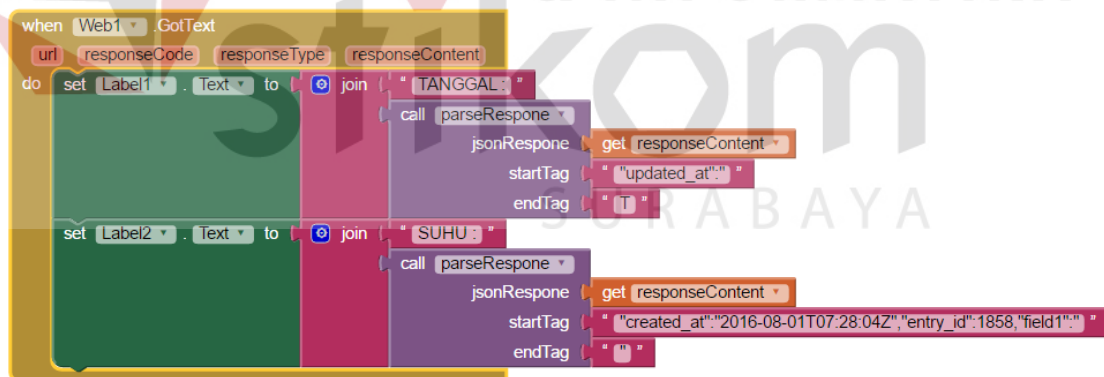
when Clock1.Timer
do
  set Web1.Url to get global url
  call Web1.Get

```

*Blocks program* *Clock.Timer* ini berfungsi ketika pada *screen 3* mendapatkan data yang masuk akan diproses secara waktu delay yang telah ditentukan. *Timer* ini akan mengambil data secara real time setiap 1 detik dari *web server*. Dalam hal ini yang berfungsi untuk mengambil data dari *web server* yaitu **Web1.Get**.



*Block program* diatas berfungsi untuk melakukan parsing data atau membagi data agar menjadi lebih sederhana setelah screen 3 mendapatkan data suhu dan kelembaban dari *web server* api.thingspeak.com. Data yang didapatkan pertama kali akan amat banyak content-content yang ikut disertakan. Pada *block* ini akan memisahkan data yang didapatkan menjadi 2 bagian yaitu **startTag** (data pertama yang disimpan) dan **endTag** (data terakhir yang disimpan).



*Block program* ini berfungsi menampilkan data parsing dengan *block* sebelumnya. Untuk menampilkan data harus memasukkan data yang diperoleh dari web server pada **startTag** dan **endTag** agar data yang ingin di tampilkan sesuai dengan yang diinginkan. Misalkan untuk dapat menampilkan data tanggal "updated\_at":"2016-08-09T02:32:57Z" pada data yang diperoleh dari



*web server* maka yang dimasukkan dalam **startTag** yaitu "updated\_at":  
kemudian yang dimasukkan pada **endTag** yaitu T dan hasil yang akan ditampilkan  
pada monitor aplikasi yaitu 2016-08-09.

```
when Screen3 .BackPressed
do close screen with value result " Screen2 "
```

*Block program* ini hanya berfungsi ketika tombol *back* pada *smartphone* di tekan. Setelah dilakukan penekanan maka *screen* akan berpindah pada *screen* sebelumnya yaitu *screen 2*.

#### 3.4.2.4 Blocks Screen 7

Pada *screen 7* ini berfungsi untuk menginputkan data suhu dan kelembaban secara manual dan sesuai keinginan *user*. Pada dasarnya *screen 7* ini berfungsi seperti pada *screen 2* tetapi yang membedakan adalah data yang dimasukkan harus secara manual dengan cara mengetik data pada aplikasi *Android*.

```
when ok .Click
do
  call TinyDB1 .StoreValue
  tag "isi2"
  valueToStore join TextBoxsuhu .Text

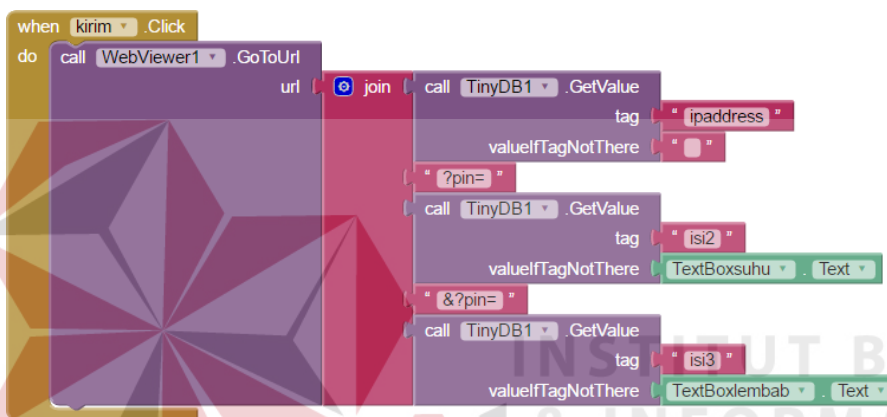
  call TinyDB1 .StoreValue
  tag "isi3"
  valueToStore join TextBoxlembab .Text

  set simpannama .Text to call TinyDB1 .GetValue
  tag "isi"
  valueIfTagNotThere TextBoxnama .Text

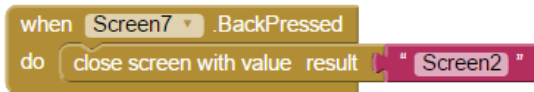
  set simpansuhu .Text to call TinyDB1 .GetValue
  tag "isi2"
  valueIfTagNotThere TextBoxsuhu .Text

  set simpanlembab .Text to call TinyDB1 .GetValue
  tag "isi3"
  valueIfTagNotThere TextBoxlembab .Text
```

*Block* diatas berfungsi untuk menyimpan data yang dimaukkan oleh user yang telah diketikkan secara manual pada **TextBoxnama**, **TextBoxsuhu**, dan **TextBoxlembab** kemudian data tersebut akan disimpan pada **TinyDB1.StoreValue**. Kemudian untuk menampilkan data yang telah disimpan tersebut dengan cara set **simpansuhu.Text** kemudian memanggil **TinyDB1.GetValue** dan memanggil **TextBoxsuhu.Text**.



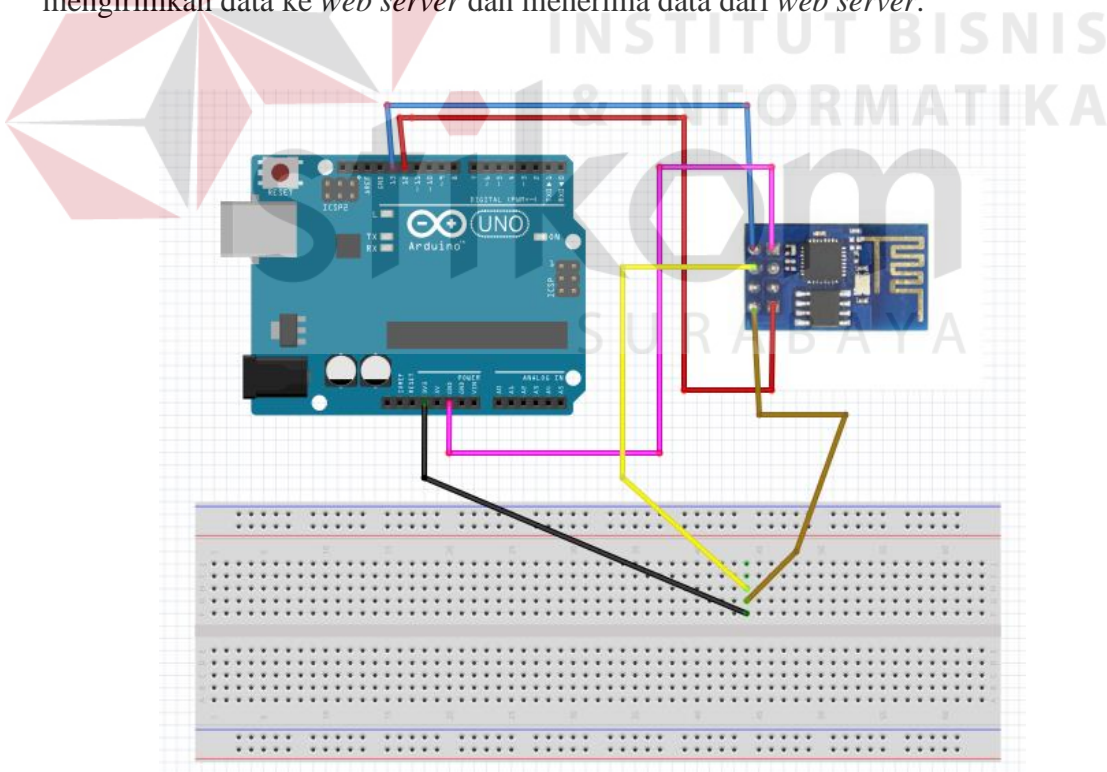
Sama dengan *button* pada *screen 2* sebelumnya, *block* diatas berfungsi untuk mengirimkan data suhu dan kelembaban kangkung ketika *button selada.Click* ditekan oleh *user*, maka akan memanggil **WebView1.GoToUrl** yaitu untuk menampung url yang ingin dikirimkan pada koneksi *wireless*. Disini url yang akan dikirimkan yaitu IP *Address* dengan cara memanggil **TinyDB1.GetValue** karena IP *Address* sudah tersimpan pada *database* pada saat proses memasukkan IP *Address* di *screen 1* dengan tag "ipaddress". Kemudian data yang dikirimkan yaitu "?pin=" saat masuk pada arduino, maka akan mengirimkan data suhu sesuai dengan yang diinputkan pada **TextBoxsuhu** dan "&?pin=" untuk data kelembaban sesuai dengan yang diinputkan pada **TextBoxlembab** oleh *user*.



*Block program* ini hanya berfungsi ketika tombol *back* pada *smartphone* di tekan. Setelah dilakukan penekanan maka *screen* akan berpindah pada *screen* sebelumnya yaitu *screen 2*.

### 3.5 Perancangan *Hardware*

Perancangan *hardware* pada tugas ini yaitu merangkai dan menghubungkan Arduino Uno dengan *Wireless Module ESP8266* untuk melakukan pengiriman data dan penerimaan data. *Hardware* ini akan berfungsi sebagai pengatur program agar dapat mengirimkan data ke *web server* dan menerima data dari *web server*.

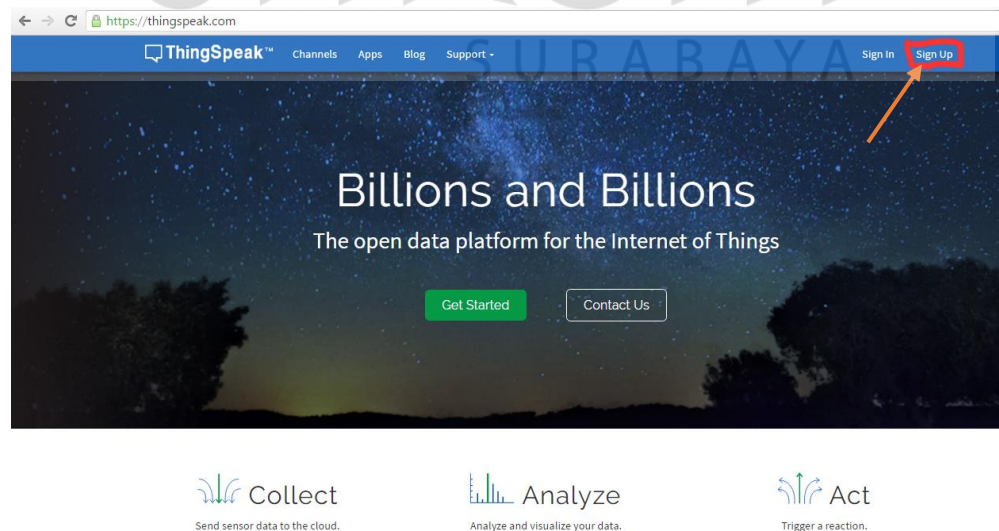


Gambar 3.12 Perancangan *Hardware*

Pada gambar 3.11 diatas adalah rangkaian dengan menggunakan kabel *jumper* dan dengan bantuan *breadboard* / *project board* agar dapat terhubung. Kabel jumper biru menghubungkan pin TX pada *Wireless Module ESP8266* dengan pin 13 pada *Arduino Uno*. Kabel jumper merah menghubungkan pin RX pada *Wireless Module ESP8266* dengan pin 12 pada *Arduino Uno*. Kabel jumper pink menghubungkan pin ground pada kedua perangkat. Kabel jumper hitam yaitu terhubung pada VCC 3.3v pada *Arduino Uno* dihubungkan dengan VCC 3.3v *Wireless Module ESP8266* pada *breadboard* / *project board*. Kabel jumper kuning pin CH-PD yang dihubungkan dengan VCC dengan bantuan *breadboard* / *project board*.

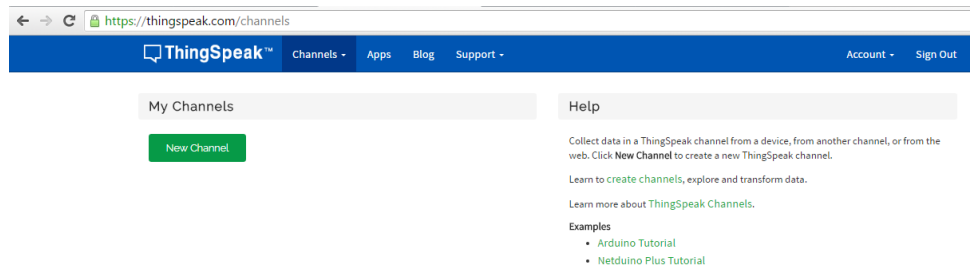
### 3.6 Pengaturan Web Server

Web server pada tugas akhir ini menggunakan <https://thingspeak.com/> yang berfungsi untuk menyimpan data dan sebagai perantara *monitoring* dan *setting*. Pertama yang dilakukan agar bisa mengakses *web server* tersebut yaitu mendaftarkan *account* baru.



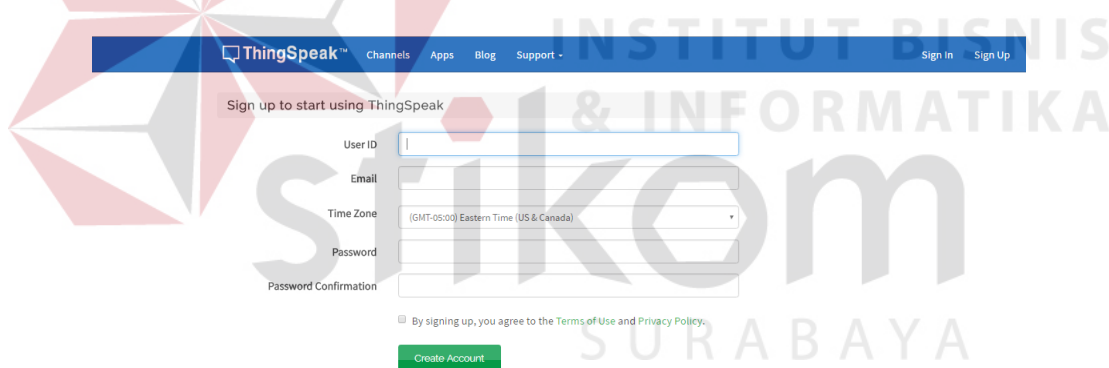
Gambar 3.13 *Sign Up* thingspeak.com

Pada gambar 3.13 setelah masuk pada link <https://thingspeak.com/> klik pada Sign Up untuk melakukan penftaran *account* baru.



Gambar 3.14 Mengisikan Data Baru Pendaftaran

Isikan data seperti User ID, email, Time Zone, dan Password seperti pada gambar 3.14 diatas. Setelah melengkapi data tersebut selanjutnya tekan Create Account. Setelah itu akan muncul halaman baru seperti pada gambar 3.14 dibawah ini.



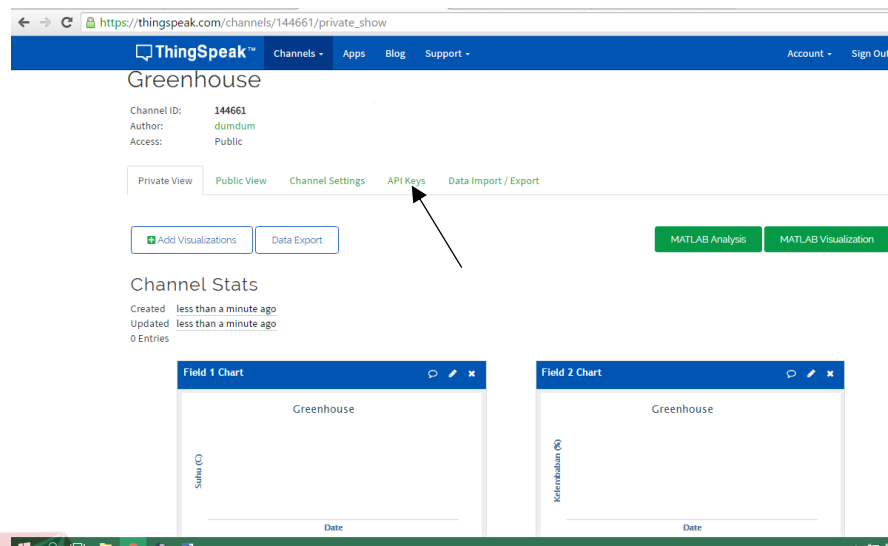
Gambar 3.15 Tampilan Awal Sebelum Membuat Channel Baru

Klik *New Channel* seperti pada gambar 3.15 diatas kemudian kehalaman baru seperti gambar 3.16 dibawah ini.

The screenshot shows the 'New Channel' form in the ThingSpeak interface. The 'Name' field contains 'Greenhouse'. The 'Description' field is empty. Under 'Fields', 'Field 1' is 'Suhu (C)' and 'Field 2' is 'Kelembaban (%)', both with checked checkboxes. Fields 3 through 8 are empty. Below the fields are 'Metadata' and 'Tags' input boxes. To the right, the 'Help' section explains 'ThingSpeak Channel' and 'Channel Settings' with a list of instructions for each field.

Gambar 3.16 Mengisikan Nama dan Field pada Channel Baru

Pada gambar 3.16 diatas yaitu mengisikan nama dengan nama Greenhouse kemudian gunakan **Field 1** dengan nama Suhu (C) dan **Field 2** dengan nama Kelembaban (%). Setelah itu klik *save channel* pada bagian bawah halaman untuk menyimpan *channel* baru tersebut dan *channel* yang telah dibuat sudah bisa digunakan untuk menyimpan data.



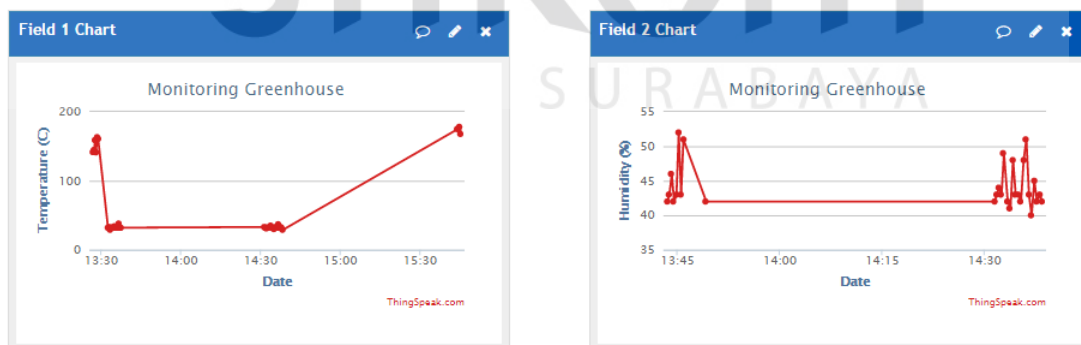
Gambar 3.17 Tampilan Awal Channel Baru

Setelah membuat channel baru seperti pada gambar 3.17 di atas maka melihat API key channel tersebut dengan cara klik API Keys. API Key tersebut sangat penting agar dapat melakukan pengiriman data dan penerimaan data serta kode API key akan digunakan dalam pemrograman Android dan Arduino Uno. Berikut adalah tampilan saat melihat API key pada gambar 3.18.

The screenshot shows the 'API Keys' tab for a channel named 'Greenhouse'. The channel ID is 144661, the author is 'dumdum', and the access is 'Public'. There are tabs for 'Private View', 'Public View', 'Channel Settings', 'API Keys', and 'Data Import / Export'. The 'Write API Key' section has a key field containing 'SjXhRouIZtSLRMWj' and a 'Generate New Write API Key' button. The 'Read API Keys' section has a key field containing 'YYK9ZSGNRTD9KCLY' and a 'Note' field. A 'Help' section explains that API keys enable writing data to a channel or reading data from a private channel. A 'Create a Channel' section shows a POST request to 'https://api.thingspeak.com/channel1.json' with headers for 'api\_key' and 'name'.

Gambar 3.18 Tampilan API Keys

Setelah mengetahui API Keys tersebut maka akan dilakukan percobaan untuk mengirimkan data ke channel tersebut. Setelah berhasil mengirimkan data, tampilan pada channel seperti gambar 3.18 dibawah ini.



Gambar 3.19 Tampilan Data Suhu dan Kelembaban pada Web Server

Pada gambar 3.19 diatas Field 1 sebagai penyimpan data suhu sedangkan Field 2 sebagai penyimpan data kelembaban. Data tersebut akan selalu bergerak secara real



time sesuai dengan delay yang diatur oleh Arduino maupun Android menggunakan koneksi internet. Tetapi terkadang delay tidak sesuai dikarenakan koneksi internet yang kurang baik dan kurang stabil.

