

## **BAB IV**

### **HASIL DAN ANALISIS PENELITIAN**

#### **4.1 Analisa Perancangan Sistem**

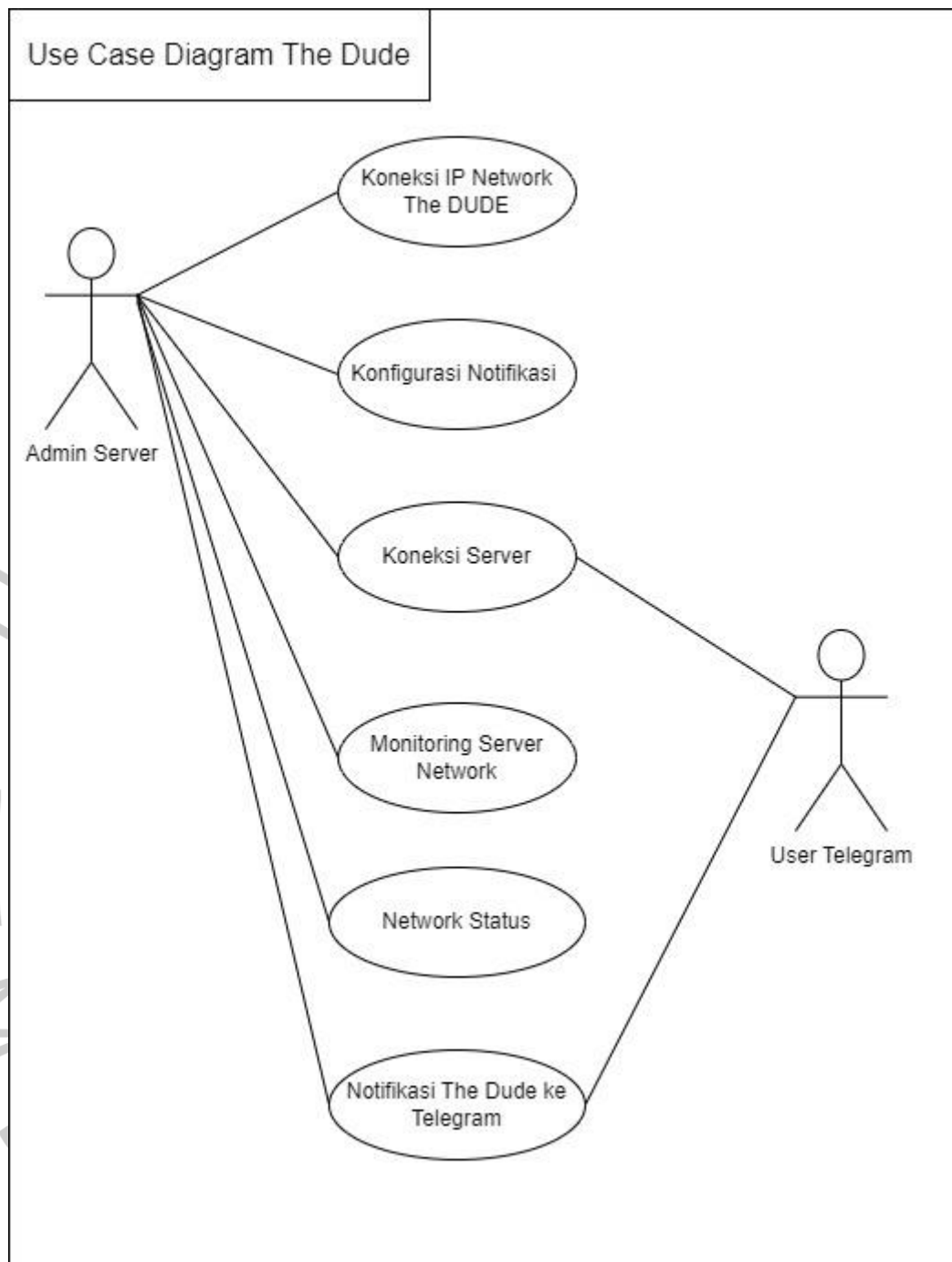
Analisa perancangan sistem merupakan rancangan yang dibuat untuk mempelajari tentang apa yang harus dilakukan oleh suatu sistem serta mengetahui bagaimana suatu elemen-elemen dalam sistem dapat bekerja dan saling berinteraksi. Analisa perancangan dibuat berdasarkan hasil dari permasalahan dan kebutuhan sistem yang dilakukan pada bab sebelumnya. Berdasarkan Analisa permasalahan yang dilakukan, dalam penelitian ini mengusulkan untuk mencari solusi agar menyelesaikan masalah yang ada. Maka pada penelitian ini dibuat perancangan sistem *monitoring* yang digunakan untuk mengirim notifikasi *server* jaringan ke dalam sebuah *BOT* pada *Telegram* dengan menggunakan Aplikasi *monitoring* jaringan yang Bernama *THE DUDE*. Agar bisa meneliti dan menganalisa sebuah jaringan yang dipantau dari aplikasi tersebut.

#### **4.2 Perancangan Diagram Sistem Usulan**

Perancangan diagram sistem dilakukan dalam penelitian tersebut menggunakan sistem metode *Object-Oriented Analysis Diagram* (OOAD) dengan jenis model *Unified Modelling Language* (UML) antara lain *Use Case Diagram*, *Activity Diagram*, *Flowchart Diagram*, dan *Entity Relationship Diagram* (ERD) untuk menentukan rancangan sistem jaringan pada aplikasi tersebut.

##### **4.2.1. Use Case Diagram**

*Use case diagram* merupakan salah satu jenis model UML yang menggambarkan hubungan interaksi antara sistem dan aktor. dalam *Use Case* menggambarkan sebuah interaksi sistem antara *admin server* yang bekerja sebagai *monitoring server* yang bertugas untuk memantau jaringan dan membuat konfigurasi notifikasi ke dalam *telegram*. Dan aktor lain yaitu *User Telegram* yang hanya bisa melihat notifikasi jaringan di *BOT Telegram*. Berikut adalah *Use case Diagram* yang digambarkan oleh peneliti:

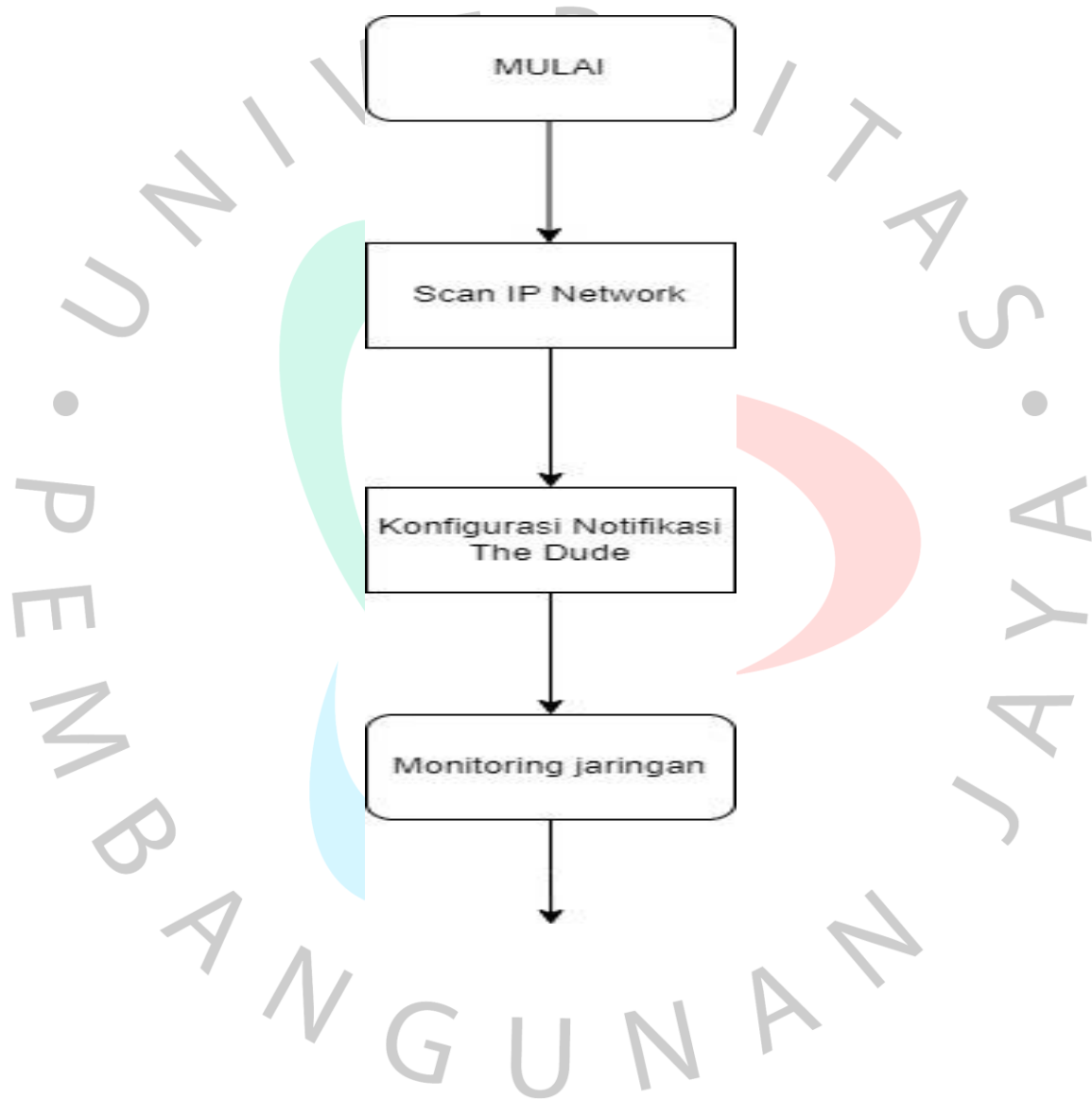


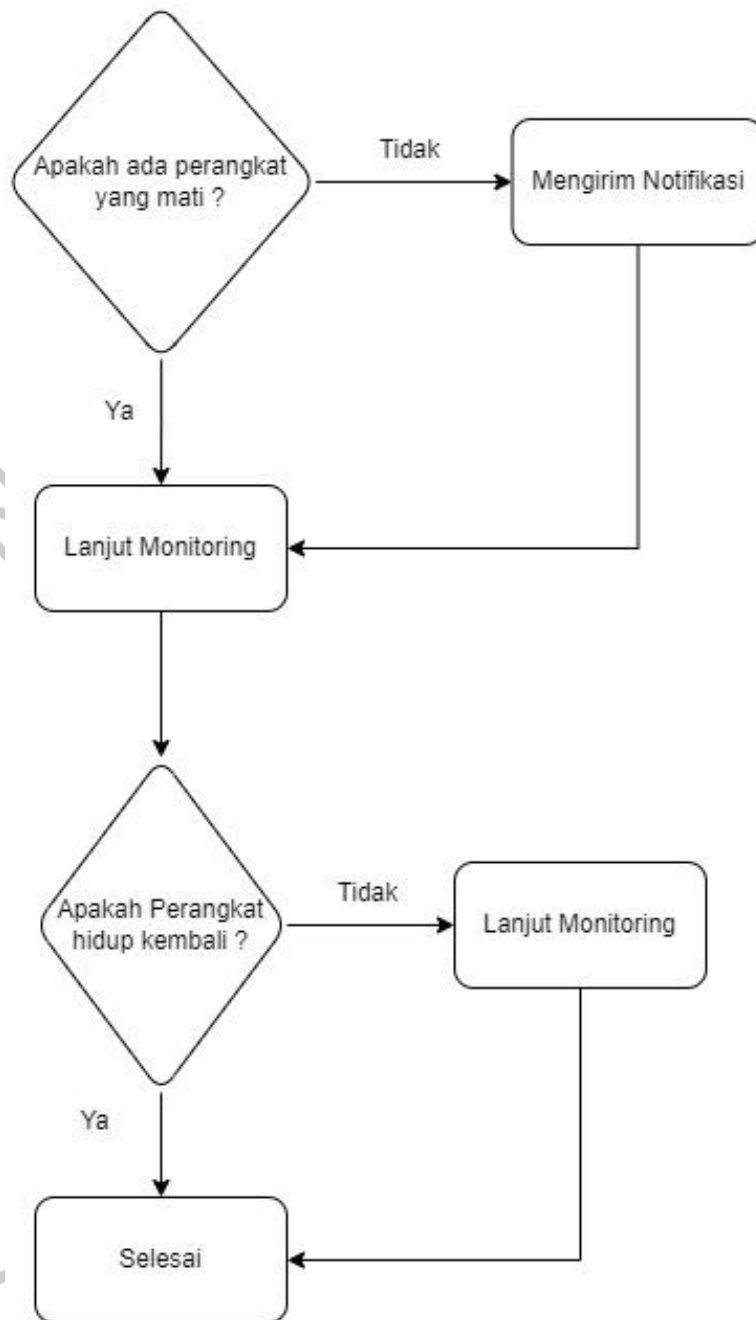
Gambar 4. 1 Use Case Diagram The Dude

#### 4.2.2. Flowchart Diagram

*Flowchart Diagram* adalah sebuah alur diagram yang menampilkan Langkah-langkah untuk melakukan sebuah proses dari suatu program. Setiap Langkah digambarkan dalam

bentuk diagram dan dihubungkan dengan garis atau panah. *Flowchart* berperan penting dalam memutuskan sebuah Langkah atau fungsional dari sebuah proyek pembuatan sistem program. Penggunaan *Flowchart* dalam dunia pemrograman juga merupakan cara yang tepat untuk menghubungkan antara kebutuhan teknis dan non-teknis. Berikut adalah *Flowchart* yang menggambarkan alur *monitoring* jaringan:





Gambar 4. 2 Flowchart Diagram

### 4.2.3. Spesifikasi Use Case

Berikut merupakan skenario Spesifikasi Use Case dalam menjelaskan rincian sistem operasi pada aplikasi *Monitoring Jaringan* yang dilakukan antara *admin* dan *user telegram*. Spesifikasi ini meliputi pada sistem proses dan kondisi yang dilakukan pada saat sistem tersebut berjalan ataupun gagal. Berikut adalah rangkuman spesifikasi use case yang telah dibuat:

Tabel 4. 1 Spesifikasi Use Case Login The Dude

<b>Nama Operasi</b>	Login	
<b>Aktor</b>	Admin	
<b>Deskripsi</b>	Dilakukan oleh Admin untuk melakukan login dengan cara mencari IP Address	
<b>Exception</b>	IP Address tidak ditemukan	
<b>Pre-Condition</b>	Jika menemukan IP Address, Maka akan bisa ke dalam Dashboard The Dude	
<b>Success Scenario</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Admin membuka Aplikasi The Dude.	2. Scan IP Network. 3. Ditemukan IP Network 4. Berhasil masuk ke Dashboard The Dude.
<b>Alternate Flow</b>	1. Admin membuka Aplikasi The Dude.	2. Scan IP Network. 3. Tidak menemukan IP Network.
	4. Refresh IP Network.	5. Ditemukan IP Network. 6. Berhasil masuk ke Dashboard The Dude

<b>Post Condition</b>	Admin akan melakukan login untuk masuk ke dalam Dashboard The Dude.	

Tabel 4. 2  
Spesifikasi  
Use Case  
Konfigurasi

Jaringan

<b>Nama Operasi</b>	Konfigurasi Jaringan	
<b>Aktor</b>	Admin	
<b>Deskripsi</b>	Dilakukan oleh Admin untuk mengkonfigurasi jaringan agar terhubung dalam notifikasi tersebut.	
<b>Exception</b>	-	
<b>Pre-Condition</b>	-	
<b>Success Scenario</b>	<b>Aktor</b>	<b>Sistem</b>
	<ol style="list-style-type: none"> <li>1. Membuka Fitur Notification.</li> <li>3. Klik Notifikasi Telegram.</li> <li>5. Memasukan Script API Token.</li> <li>6. Klik Apply</li> </ol>	<ol style="list-style-type: none"> <li>2. Menampilkan jenis-jenis Notifikasi.</li> <li>4. Menampilkan Nama, Tipe, dan Script.</li> <li>7. Data Tersimpan</li> </ol>
<b>Alternate Flow</b>	-	
<b>Post Condition</b>	Script API token Akan otomatis tersimpan dalam notifikasi agar bisa terhubung notifikasi jaringan tersebut	

Tabel 4. 3 Spesifikasi Use Case Notifikasi Telegram

<b>Nama Operasi</b>	Notifikasi Telegram	
<b>Aktor</b>	User Telegram	
<b>Deskripsi</b>	Membuat sebuah BOT Telegram oleh User Telegram untuk menjalani notifikasi Telegram Tersebut.	
<b>Exception</b>	-	
<b>Pre-Condition</b>	-	
<b>Success Scenario</b>	<b>Aktor</b>	<b>Sistem</b>
	<p>1. Membuat BOT melalui BOT Father</p> <p>3. Klik /newbot untuk membuat User BOT baru.</p> <p>5. Membuat User ID BOT Telegram.</p> <p>7. Masukkan API Token ke dalam Link URL</p> <p>9. Buka kembali Aplikasi The Dude untuk konfigurasi ulang</p> <p>10. Matikan salah satu IP dan Nyalakan Notifikasi tersebut.</p>	<p>2. Menampilkan Sistem BOT</p> <p>4. Menampilkan chat untuk memberikan ID Username BOT</p> <p>6. Menampilkan Chat bahwa User ID BOT sudah dibuat dan memberikan API Token.</p> <p>8. Memberikan perintah dalam bentuk Script Code.</p>

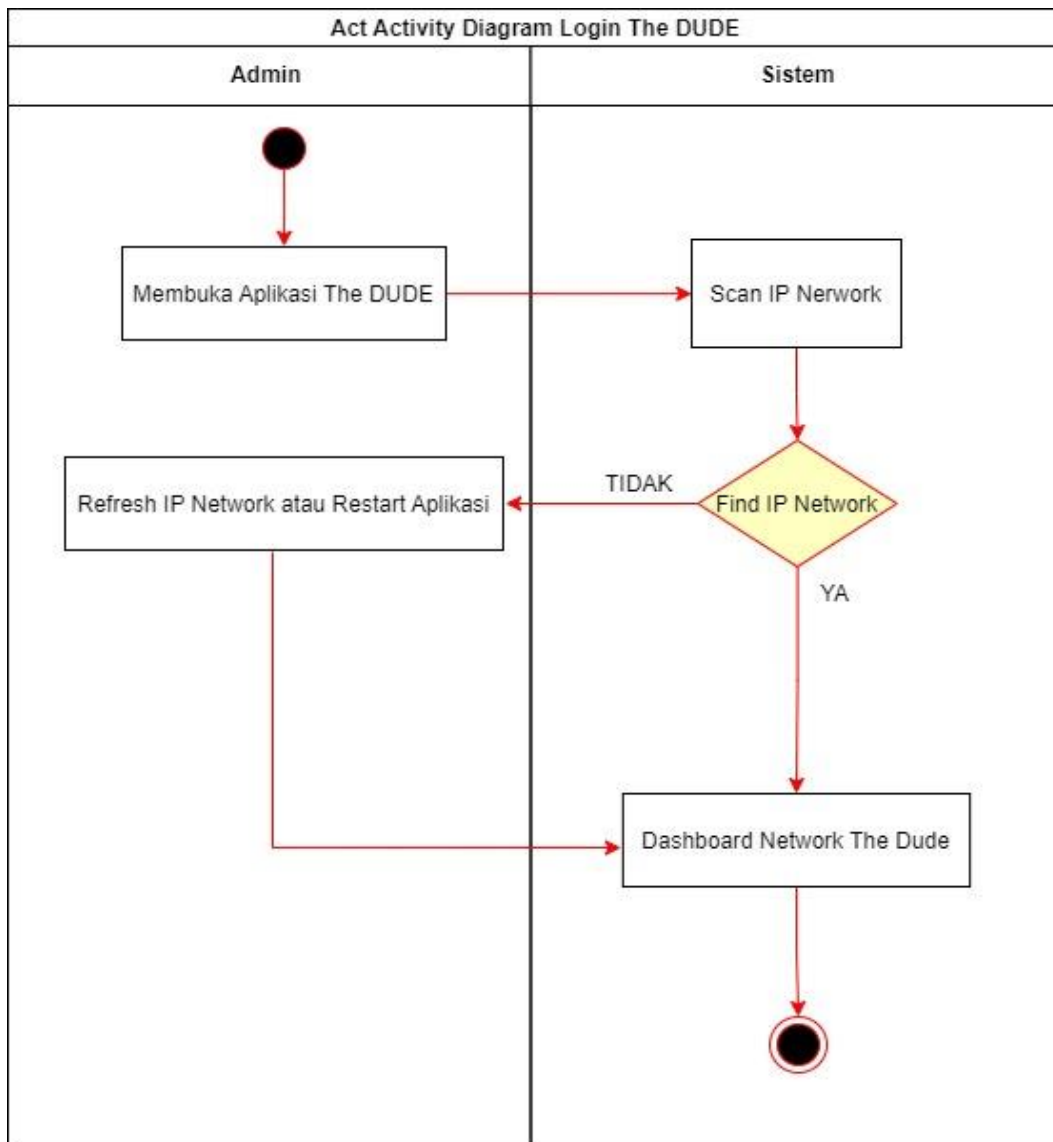
		11. Muncul Notifikasi Server pada aplikasi Telegram.
<b>Alternate Flow</b>	1. Memasukan API Token Ke dalam Link URL  4. Periksa kembali API Token  6. Buka kembali aplikasi the dude dan konfigurasi kembali  7. Matikan salah satu IP dan nyalakan notifikasi tersebut.	2. Memberikan Perintah dalam bentuk Script Code  3. Perinta dari Script Code Failed  5. Memberikan Perintah OK dalam bentuk Script Code  8. Muncul Notifikasi Server pada aplikasi telegram
<b>Post Condition</b>	Akan menampilkan pesan berupa notifikasi jaringan dalam aplikasi telegram	

#### 4.2.4. Activity Diagram

*Activity Diagram* merupakan diagram yang dirancang untuk melihat dan menganalisa sebuah aktifitas sistem yang dijalankan. *Activity Diagram* juga memiliki elemen-elemen dengan berbagai bentuk yang dihubungkan dengan tanda panah agar mengarah ke diagram aktivitas tersebut sebagai proses jalannya suatu sistem. Berikut adalah *Activity Diagram* yang dibuat oleh Peneliti:

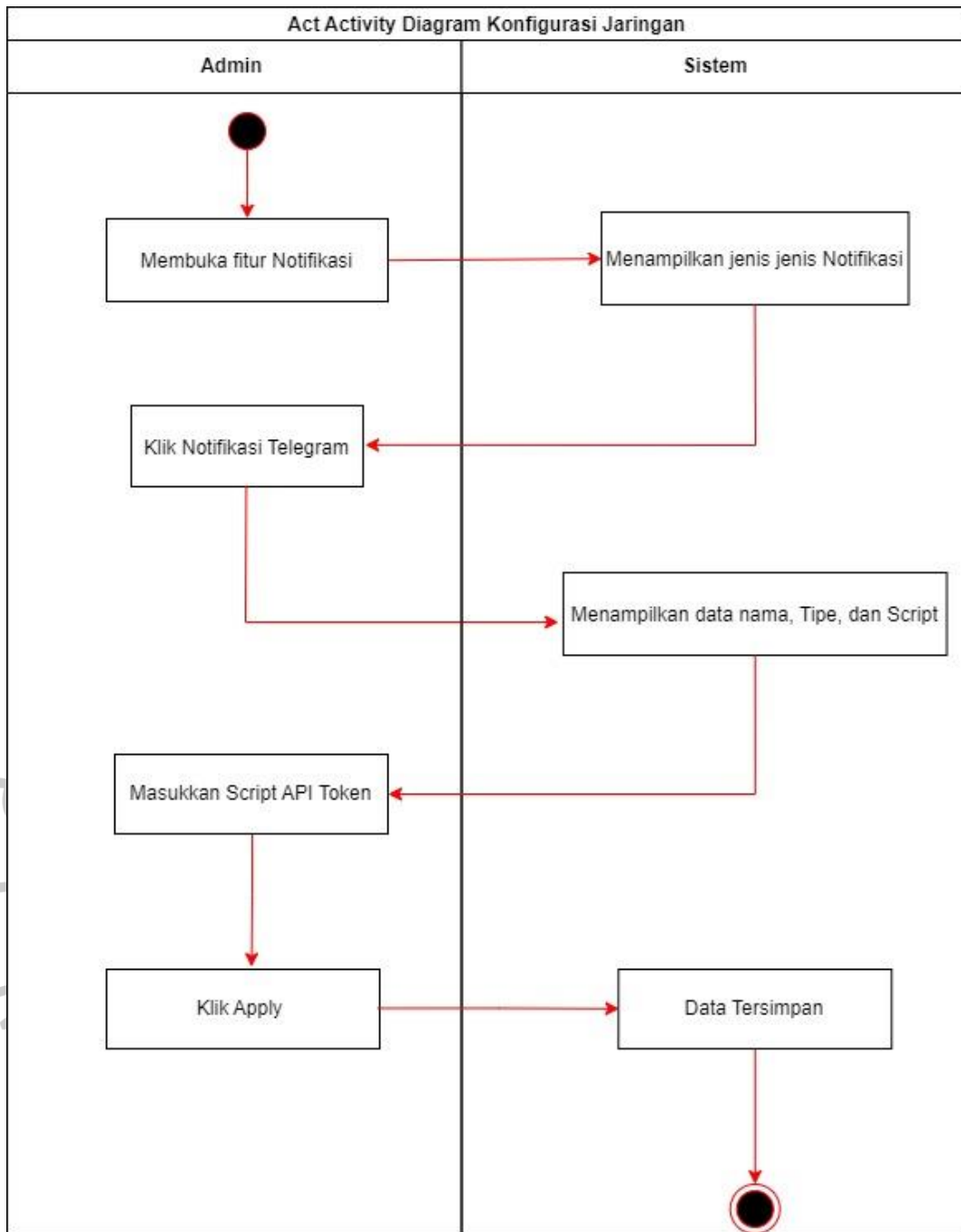
**Gambar 4.3** menjelaskan alur aktivitas diagram yang dilakukan oleh *Admin Server* dalam membuka aplikasi *The Dude*.





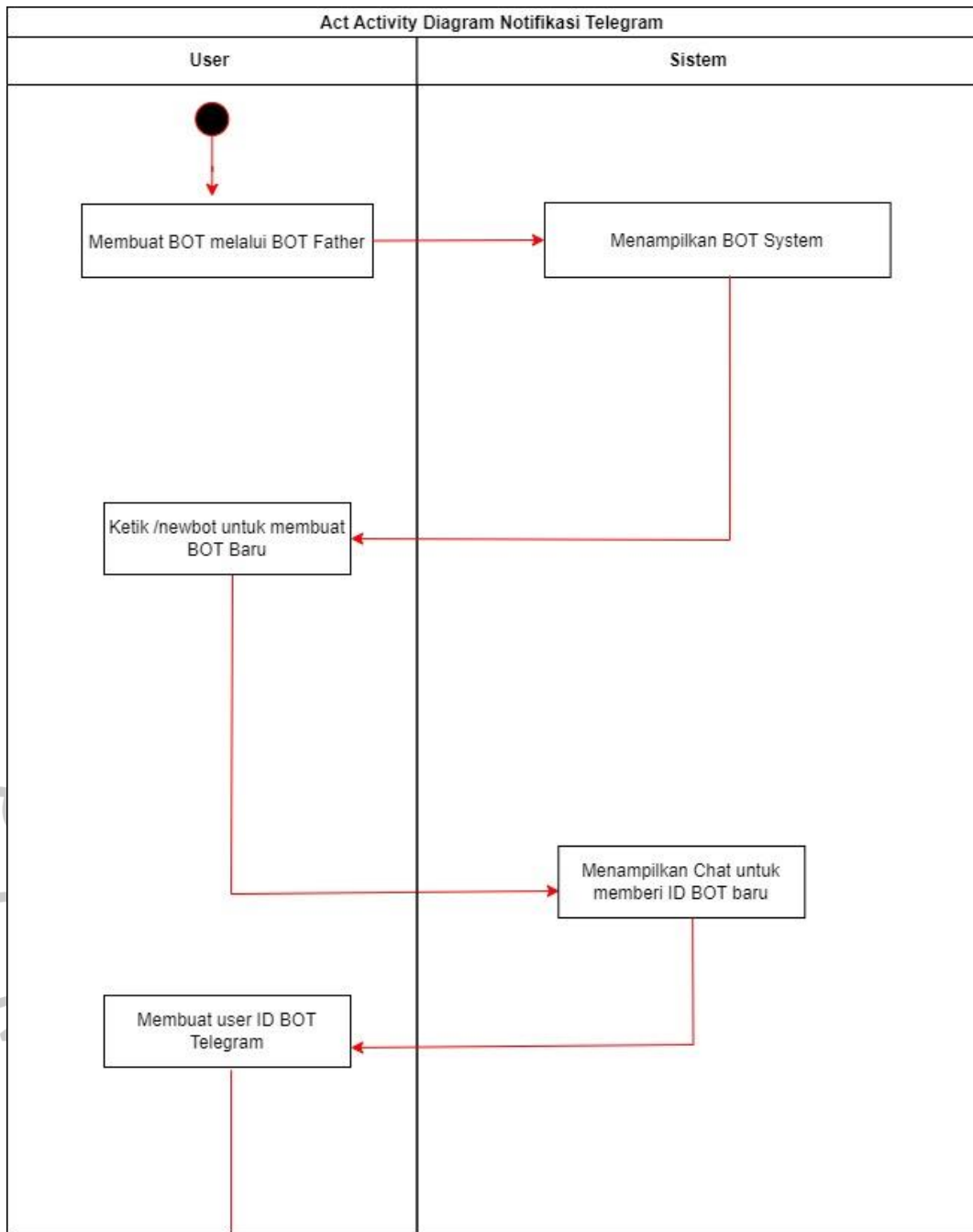
Gambar 4. 3 Activity Diagram Login The Dude

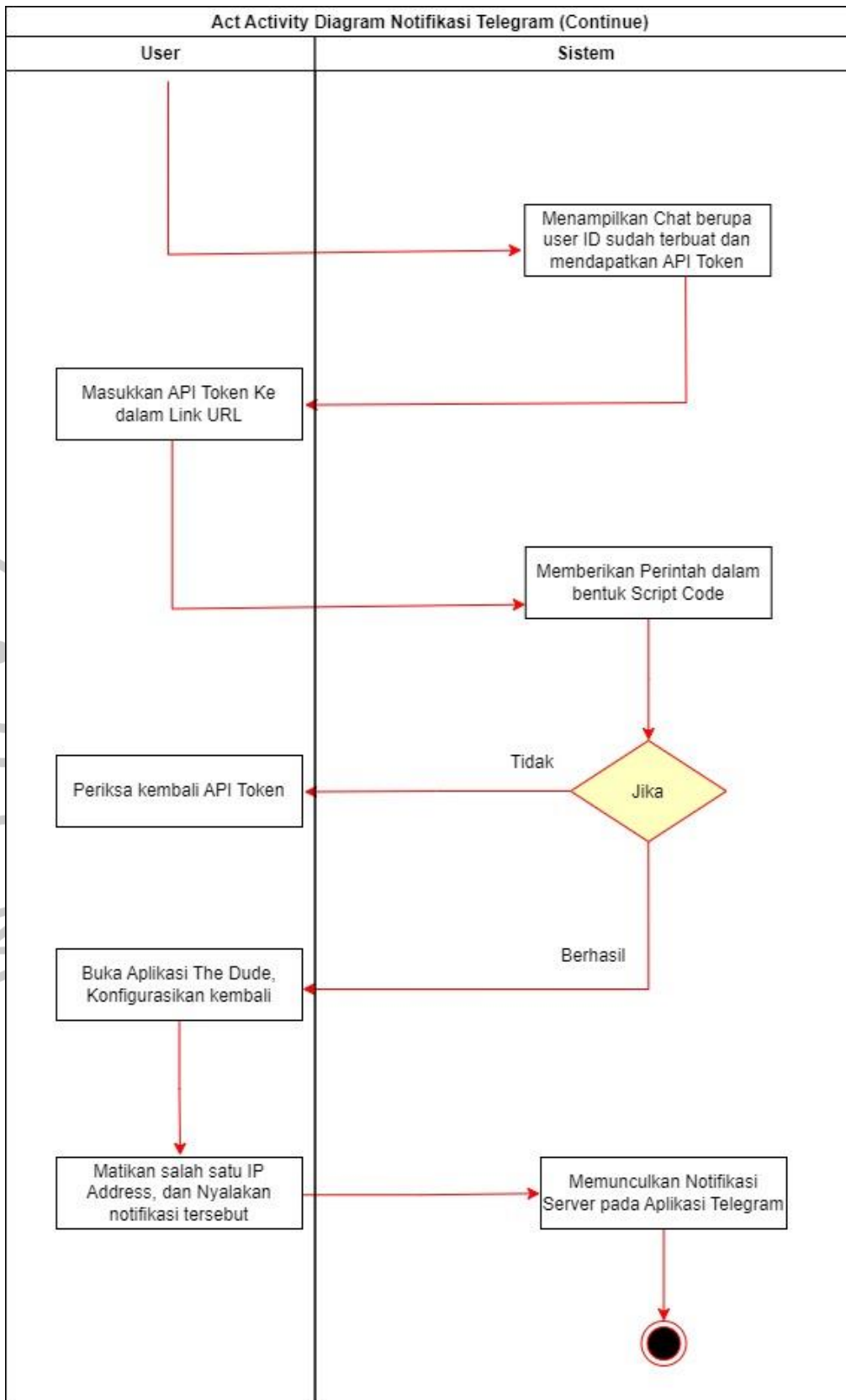
Gambar 4.4 Menjelaskan Activity Diagram cara mengkonfigurasi jaringan ke dalam fitur notifikasi dalam aplikasi *The Dude*.



Gambar 4. 4 Activity Diagram Konfigurasi Jaringan

Gambar 4.5 menjelaskan sebuah alur Activity Diagram yang menggambarkan sebuah alur koneksi pada notifikasi jaringan pada *The Dude* ke dalam aplikasi *BOT Telegram*.





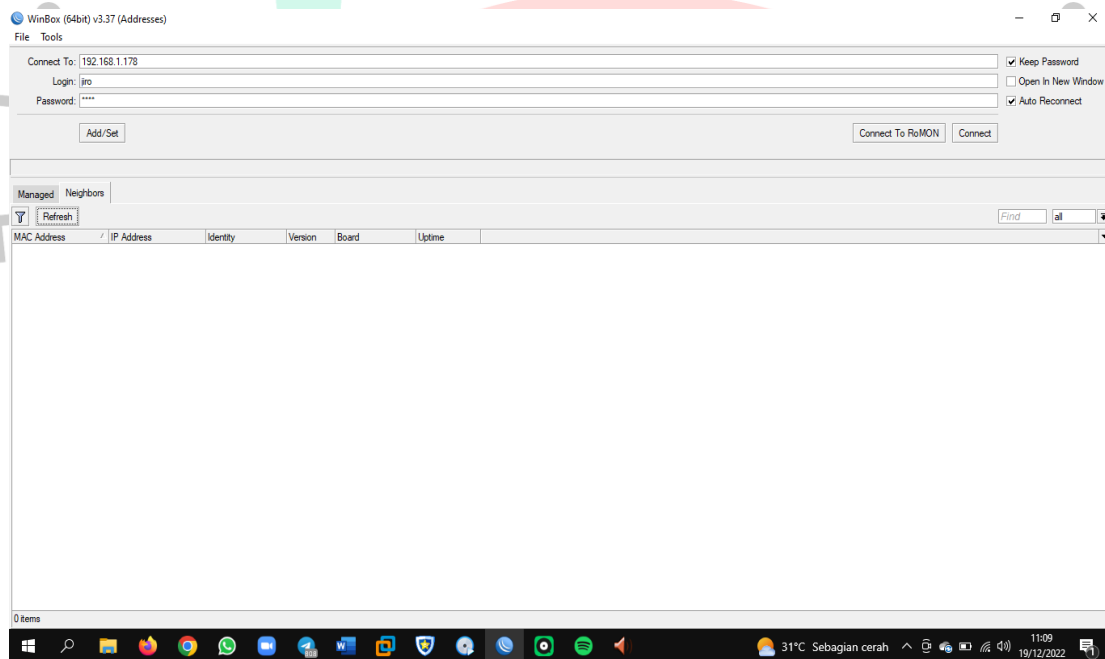
**Gambar 4. 5 Activity Diagram Notifikasi Telegram**

### 4.3 Perancangan Antar Muka Pengguna

Perancangan antar muka pengguna merupakan suatu perancangan yang menjelaskan bagaimana cara menampilkan dan menjalankan sebuah sistem pada aplikasi tersebut. Perancangan tersebut dilakukan untuk dapat melihat bagaimana cara kerja dari aplikasi tersebut dan juga menganalisa sebuah gambaran grafik dari aplikasi jaringan tersebut. Perancangan antar muka pada aplikasi *The Dude* dilakukan oleh *Admin Server* untuk memantau jaringan agar jaringan berjalan dengan baik. Berikut adalah tampilan-tampilan dalam aplikasi Winbox dan *The Dude* yang menampilkan sebuah *Dashboard Server* jaringan.

#### 4.3.1. Tampilan Aplikasi Winbox

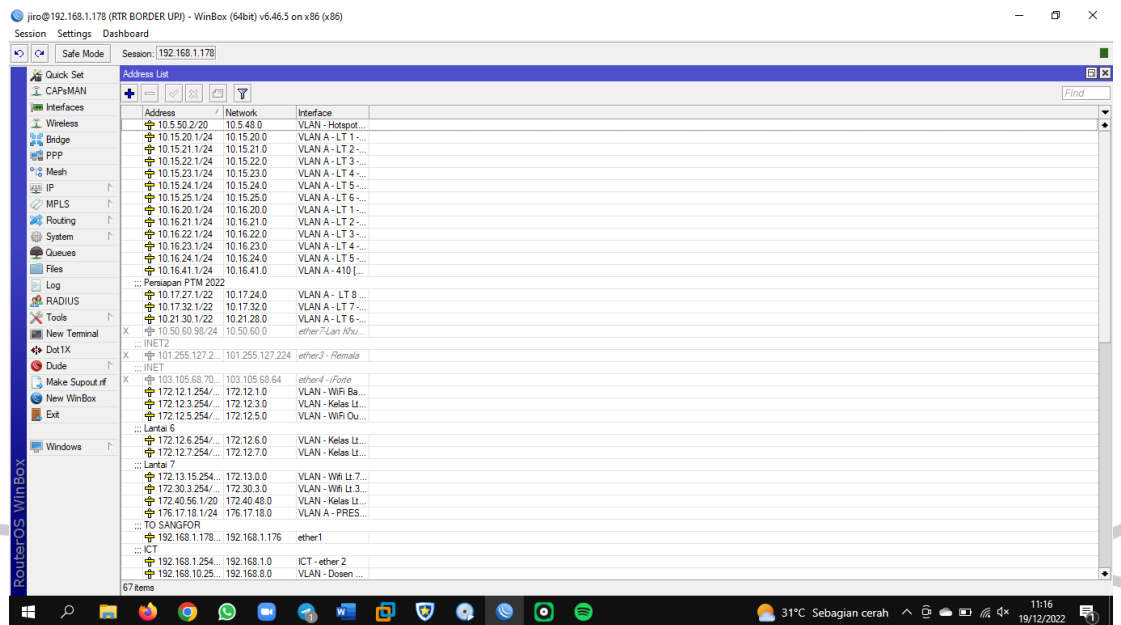
**Gambar 4.6** merupakan tampilan awal dari aplikasi *Winbox*, menjelaskan bahwa *admin* harus mencari *IP Router* yang akan tersambung ke dalam aplikasi tersebut. *IP Router* yang ditemukannya adalah *IP Router* milik ICT UPJ yang tersambung ke dalam *winbox* tersebut.



Gambar 4. 6 Tampilan Aplikasi Winbox64

### 4.3.2. Tampilan Server IP Address

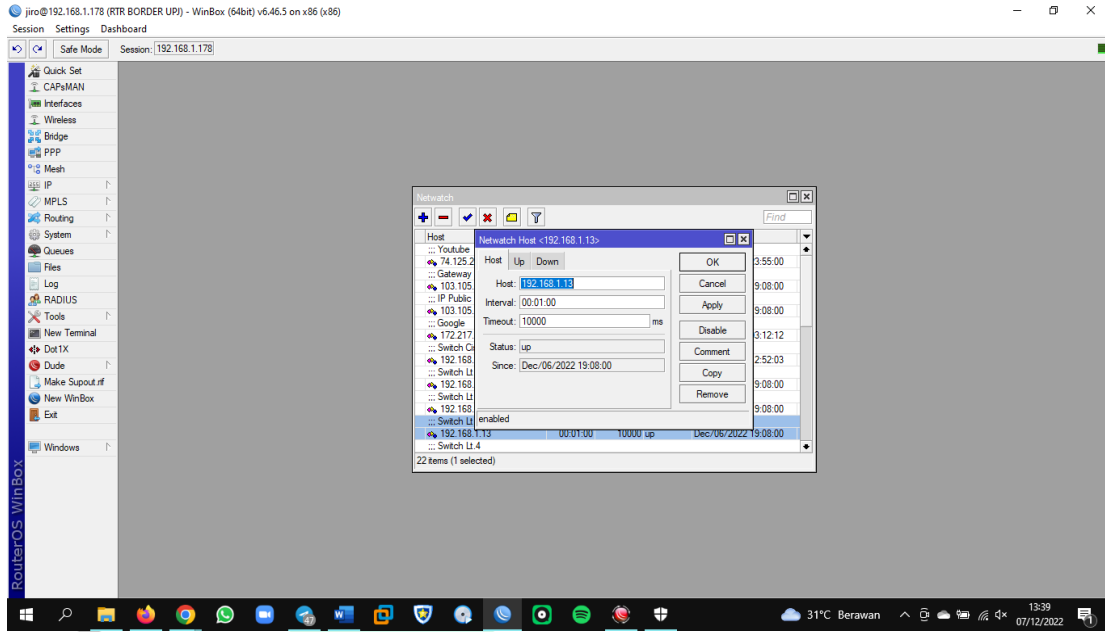
Gambar tersebut merupakan *IP Address* yang ditampilkan dalam daftar *Address*. Banyak *IP Address* yang terpasang dalam setiap lantai dalam Gedung kampus yang berbeda. *IP Address* yang digunakan untuk bahan penelitian adalah *IP Address* Lantai 3 Gedung B UPJ dengan *IP* 192.168.1.154. Berikut adalah tampilan *IP Address*:



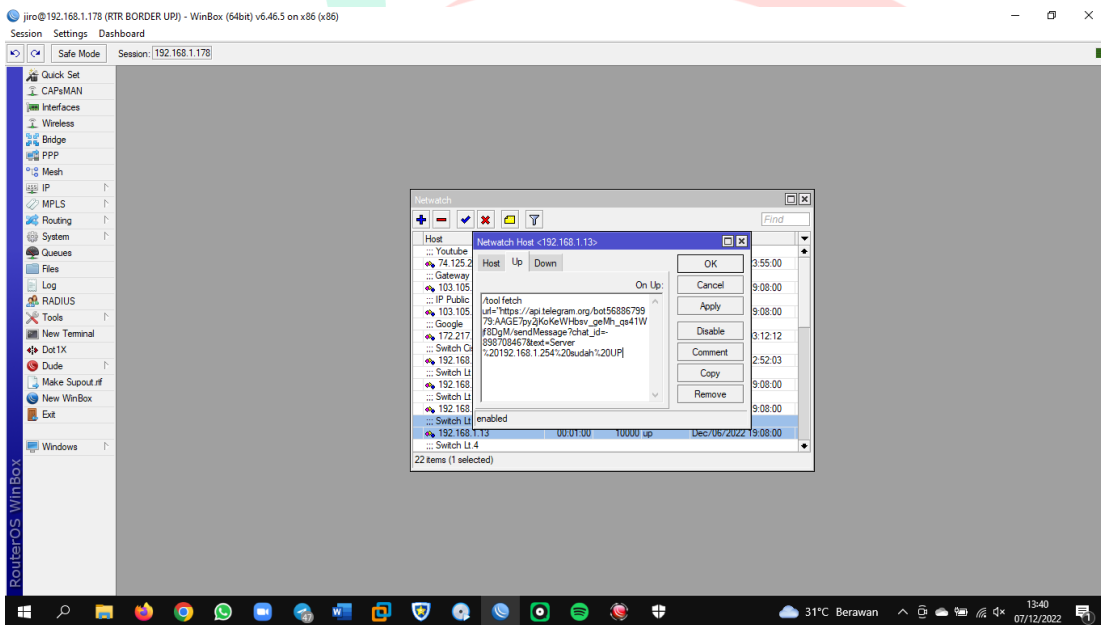
Gambar 4. 7 Tampilan IP Address

### 4.3.3. Tampilan Netwatch Host

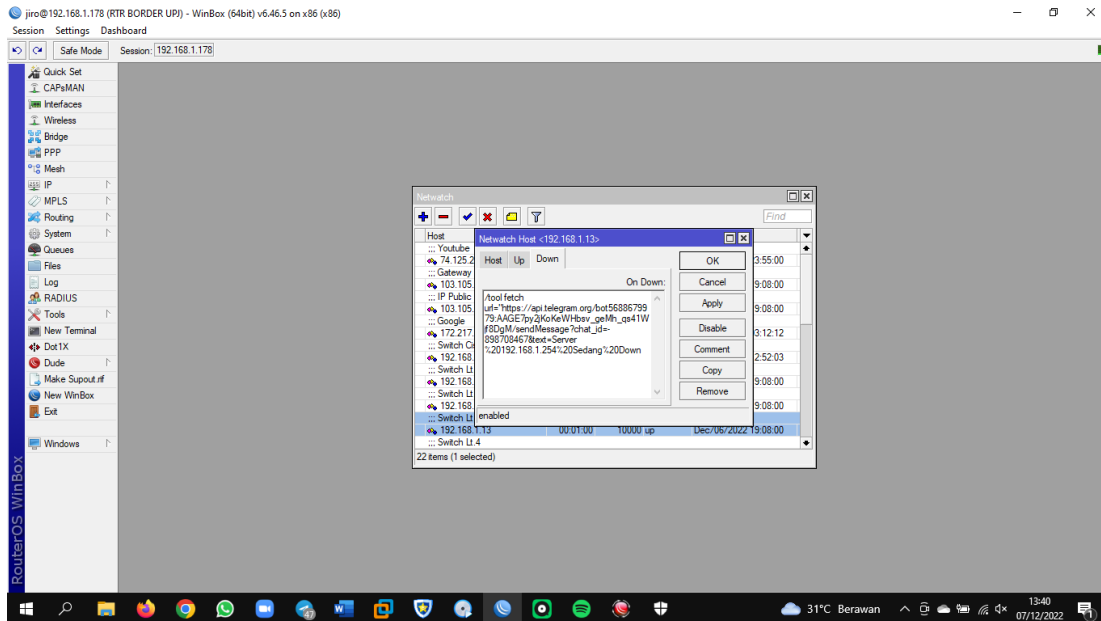
Dalam kedua gambar tersebut merupakan fitur yang digunakan untuk konfigurasi *IP* yang ingin terhubung ke dalam notifikasi *telegram*. Dalam tersebut terdapat 3 langkah untuk mengkonfigurasi jaringan antara lain *Host*, *Up*, dan *Down*. *Host* merupakan fitur dimana *Configuration Host* dengan *IP Address* yang harus dimasukkan ke dalam fitur tersebut agar jaringan mudah terkoneksi ke dalam aplikasi tersebut. Fitur *Up* merupakan sebuah fitur yang dilakukan untuk memasukkan *Script Code* dari *API Token Telegram* yang terdapat dari *BOT Father* agar proses sistem berjalan dan berguna untuk memastikan bahwa *Server* tersebut sedang *UP*. Fitur *Down* kurang lebih seperti fitur *Up* yang dimana untuk memastikan bahwa *Server* tersebut sedang *Down*. Berikut adalah tampilan *Netwatch Host* dalam aplikasi *Winbox* :



Gambar 4. 8 Tampilan Netwatch Host



Gambar 4. 9 Netwatch Host dengan Server Up



Gambar 4. 10 Netwatch Host dengan Server Down

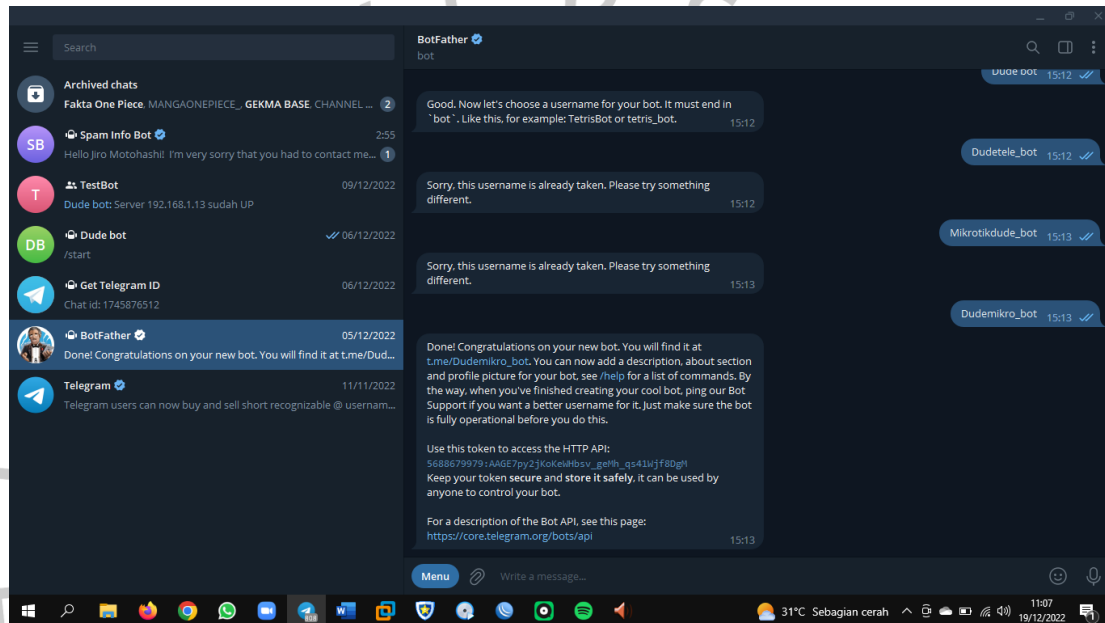
#### 4.4 Perancangan Implementasi

Perancangan implementasi merupakan suatu proses rancangan aplikasi yang sudah dibuat untuk dijadikan bahan objek penelitian Tugas Akhir. Proses tersebut akan dilakukan secara *Step by Step*. Perancangan tersebut telah dibuat berdasarkan Analisa yang dilakukan serta diagram sudah digambarkan sesuai dengan proses berjalannya aplikasi tersebut. Penelitian ini juga telah melakukan pengujian terhadap aplikasi yang telah dibuat. Namun, ditengah pengujian mengalami kesulitan pada aplikasi *The Dude* tersebut sehingga mencari solusi agar aplikasi tetap berjalan. Berikut adalah hasil percobaan dari aplikasi *The Dude* yang telah dibuat untuk koneksi jaringan dari Aplikasi *The Dude* agar memunculkan notifikasi jaringan yang terhubung dalam *BOT Telegram*.



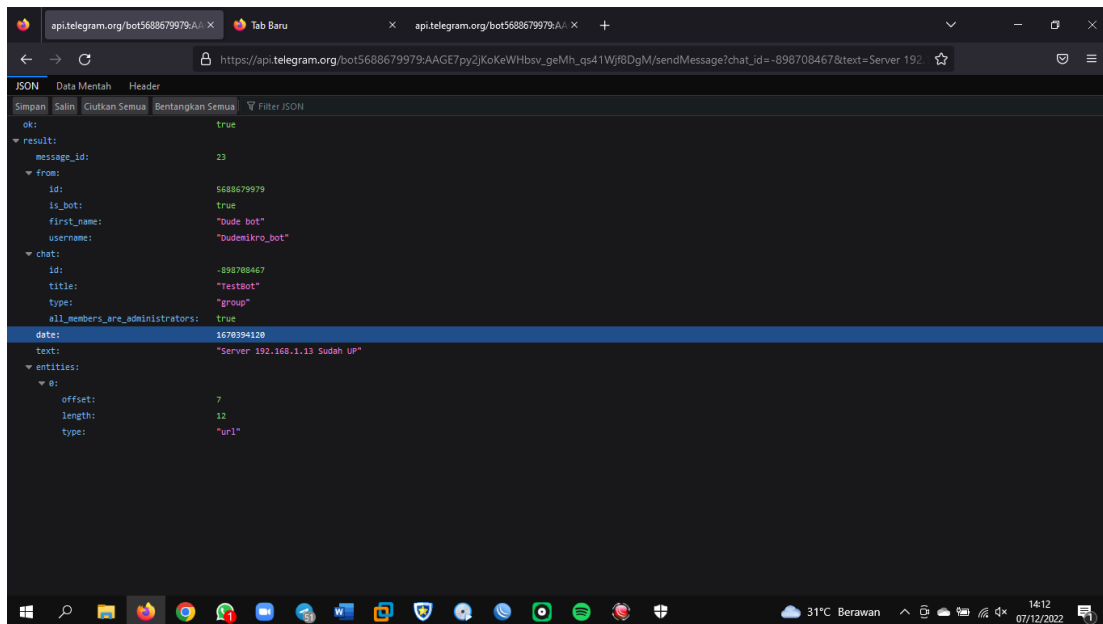
#### 4.4.1. Perancangan Aplikasi *Telegram*

Dalam perancangan ini, dilakukan tahap pertama yaitu membuat sebuah *BOT* melalui sistem *BOT Father*. *BOT Father* adalah *Channel* yang dibuat oleh *Telegram* sebagai sistem *robot coding* yang akan dijalankan secara otomatis. Dalam tahap ini, dibuat nama *BOT* sesuai yang dibutuhkan. Seperti yang ada pada **Gambar 4.12** yang dimana merupakan nama dari *BOT* tersebut.



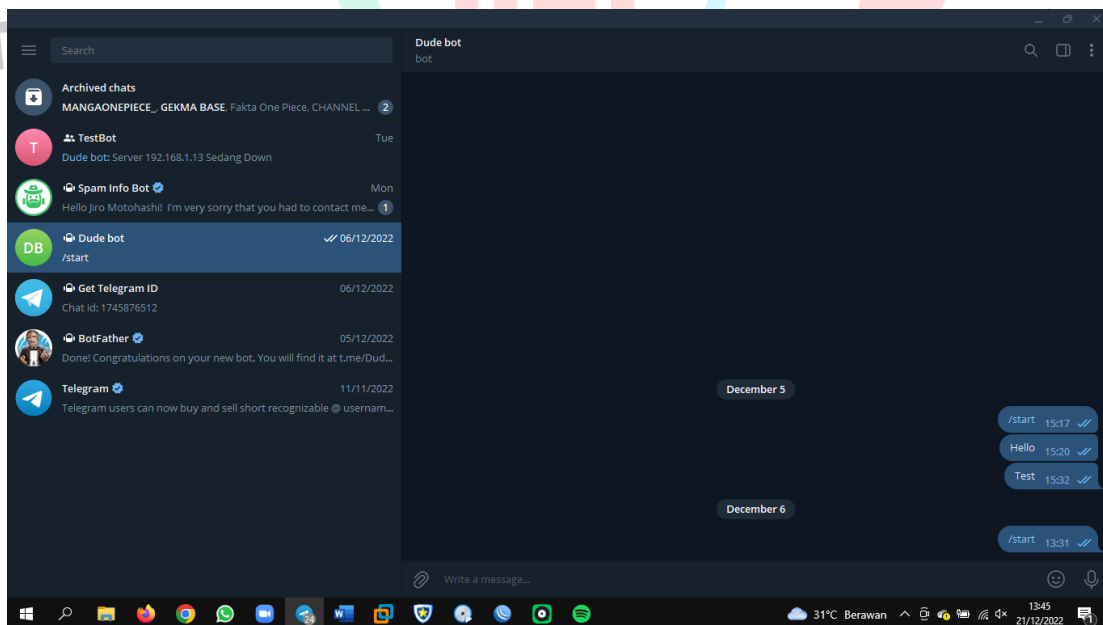
**Gambar 4. 11** Tampilan *Channel BOT Father*

Sesuai gambar yang diatas, setelah *User* memiliki nama *BOT* maka akan diberikan *API Token Telgram*. *API (Application Programming Interface)* adalah sebuah *Script Interface* yang dapat menghubungkan pada aplikasi tersebut ke aplikasi lainnya. *API Token* merupakan autentikasi *script* yang digunakan untuk memberi perintah dalam *Link URL* tersebut. Setelah mendapatkan *API Token* link tersebut disalin ke dalam link *web* untuk mendapatkan perintah seperti di dalam **Gambar 4.13** sebagai berikut:

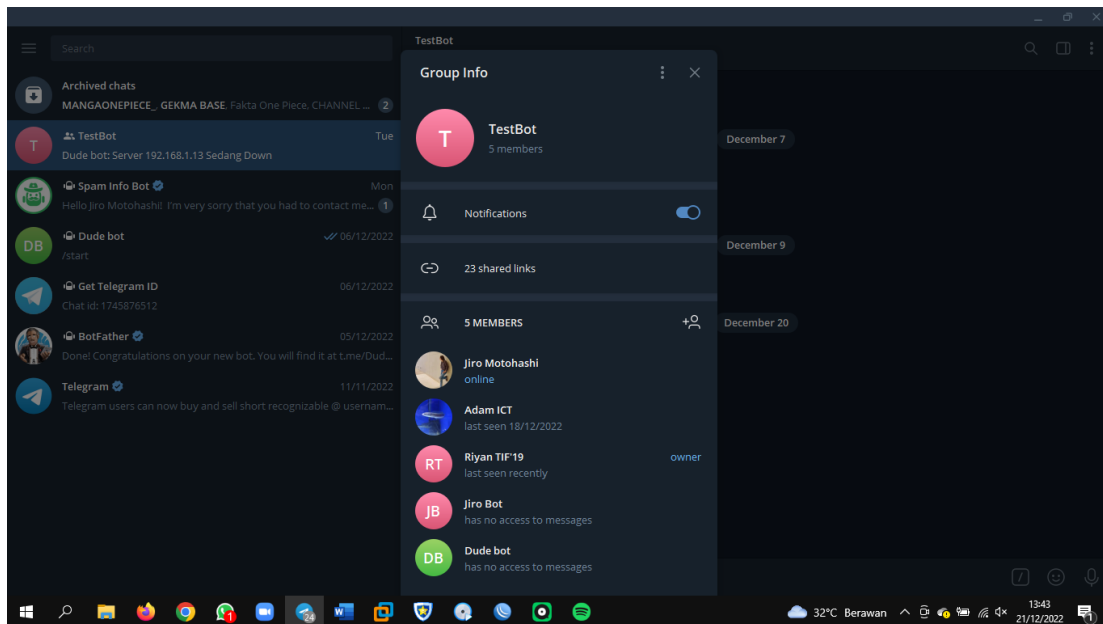


Gambar 4. 12 Sistem Perintah dari API Token

Setelah mendapatkan perintah dari *API Token*, *User BOT* akan mengaktifkan notifikasi dan memasukkan ke dalam grup khusus notifikasi jaringan. Berikut adalah nama *BOT* yang digunakan dan juga grup yang dimasukkan:



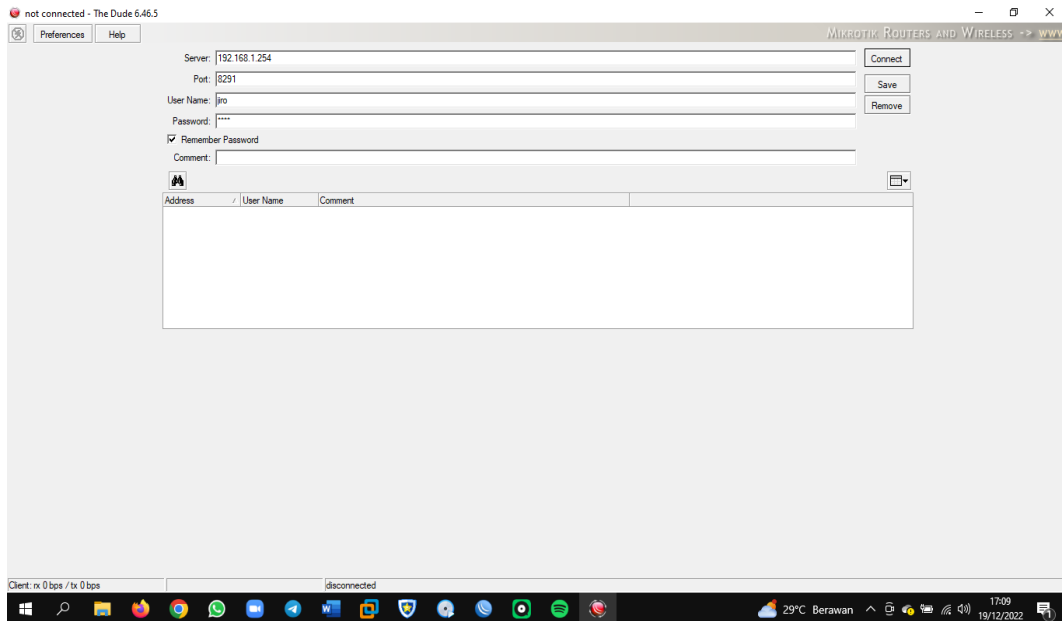
Gambar 4. 13 BOT User dengan nama "Dude\_Bot"



Gambar 4. 14 Grup yang digunakan untuk Notifikasi Jaringan

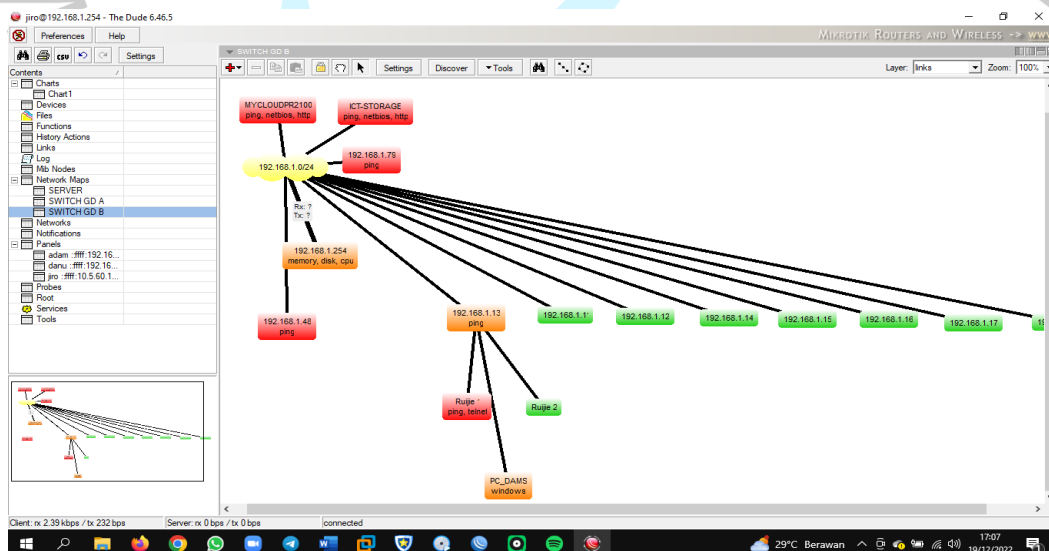
#### 4.4.2. Perancangan Aplikasi *The Dude*

*The Dude* merupakan aplikasi *Monitoring Network* yang dibuat oleh Mikrotik yang berguna untuk menganalisa, dan memantau jaringan *server*. Aplikasi tersebut akan digunakan untuk menjalani sistem proses notifikasi jaringan pada aplikasi *Telegram*. Dalam aplikasi tersebut ada beberapa tahapan yang harus dilakukan sehingga dapat berjalan notifikasi jaringan tersebut. Berikut adalah tampilan depan dari aplikasi *The Dude* yang ada pada **Gambar 4.16**. Dalam tampilan tersebut *Admin* akan mencari *IP Address* yang ditemukan, *IP* yang dipakai adalah *IP* 192.168.1.13. *IP* ini yang akan digunakan untuk mengkonfigurasi jaringan ke dalam aplikasi *Telegram* sehingga dapat menjalani sistem notifikasi tersebut.



Gambar 4. 15 Tampilan Aplikasi The Dude

Setelah melakukan *login*, *admin* akan masuk ke dalam *Dashboard The Dude* yang dimana tampilan tersebut banyak sekali fitur-fitur yang ada pada aplikasi tersebut. Dalam fitur *Network Maps* terdapat 3 jenis *Network* antara lain *Server*, *Switch GD A*, dan *Server GD B*. Namun, *Server* yang digunakan adalah *Server GD B* di dalam *Network Server GD B* terdapat beberapa *Device IP* yang terpasang. *IP* yang digunakan untuk notifikasi jaringan adalah *IP* 192.168.1.13 milik ICT yang berada di lantai 3 Gedung B UPJ sedangkan *Server* dengan *IP* 192.168.1.0/24 adalah *Router* utama yang berada pada ruang *server* ICT berikut adalah tampilan *Network Server* yang ditampilkan pada gambar berikut :



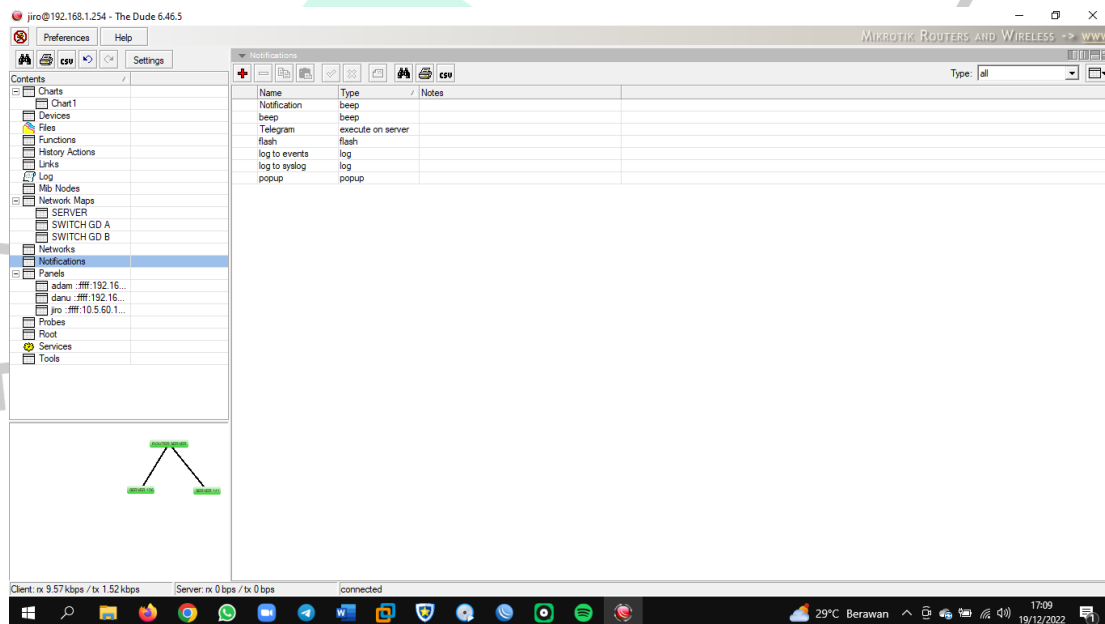
Gambar 4. 16 Server Gedung B UPJ

Sesuai gambar diatas tersebut, merupakan sebuah tampilan *Server* yang berada pada Gedung B UPJ, Namun dilihat dari warna tersebut memiliki fungsi masing-masing. Jikalau

server berwarna hijau itu menandakan bahwa jaringan tersebut aman digunakan, Jika berwarna jingga menandakan bahwa Server tersebut ada dua kemungkinan antara lain Server tersebut aman atau bisa saja sedang mengalami *Error*. Dan jika berwarna merah sudah dipastikan bahwa Server sedang mengalami *Down* atau *Error*. Maka dari itu jika mengalami warna merah. *Admin* akan mencoba bagaimana cara mengatasi jaringan *error* tersebut.

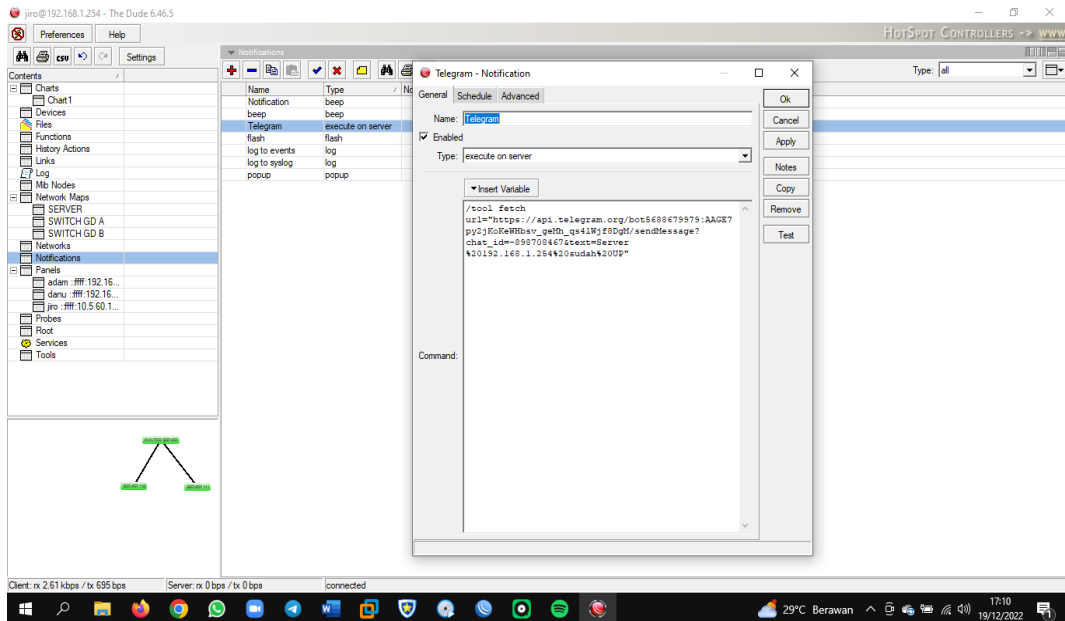
#### 4.4.3. Konfigurasi Jaringan

Tahap berikutnya, *Admin* melakukan notifikasi jaringan. *Admin* melakukan konfigurasi jaringan yang terdapat pada aplikasi *The Dude* dalam fitur *Notification*. Dalam fitur tersebut terdapat beberapa jenis notifikasi, selanjutnya *Admin* memasukkan *Script* notifikasi tersebut dengan nama *Telegram*. Seperti pada **Gambar 4.17** sebagai berikut:



Gambar 4. 17 Fitur *Notifications* dalam Aplikasi *The Dude*

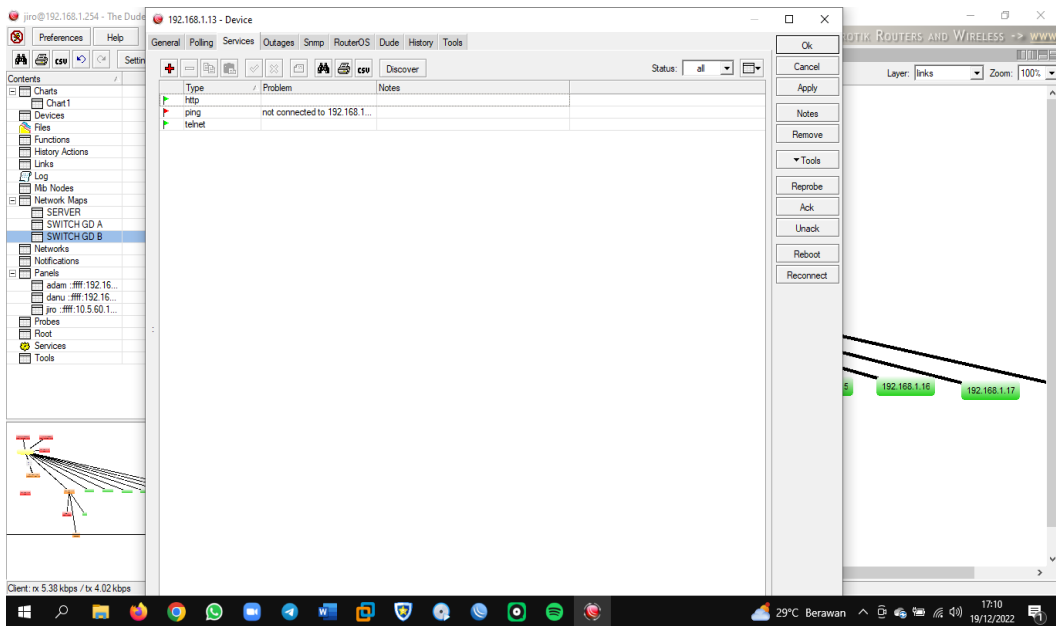
Kemudian, *Admin* masukkan *link API Token* yang sudah diberikan oleh *BOT Father* dari aplikasi *Telegram* ke dalam konfigurasi notifikasi jaringan seperti pada gambar yang ditampilkan sebagai berikut:



Gambar 4. 18 Script Code yang dimasukkan dalam Notifikasi

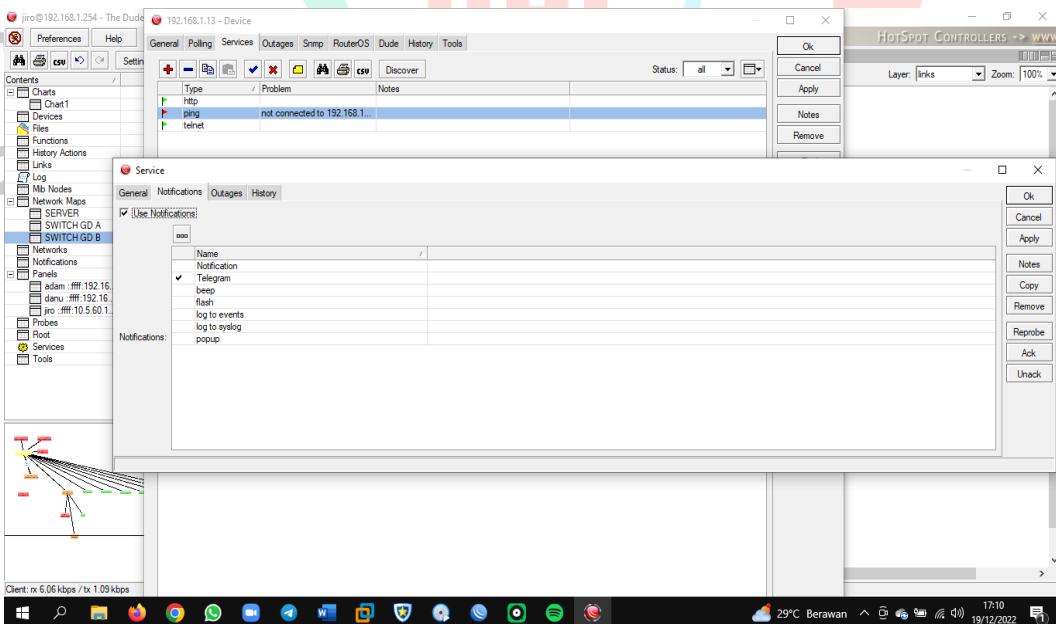
#### 4.4.4. Notifikasi Jaringan

Setelah konfigurasi jaringan selesai, berikutnya tahap terakhir untuk melakukan notifikasi jaringan. *Admin* memilih salah satu *IP Server* untuk melakukan notifikasi jaringan. *IP* yang digunakannya adalah 192.168.1.13, lalu klik dan muncul dengan tampilan *Device*. Dalam gambar tersebut terdapat beberapa jenis *Device Service* yang ada. Kemudian nyalakan notifikasi tersebut yang ada pada *Device* dengan *Type* “ping”. Jika *Type* “Ping” dinyalakan maka secara otomatis notifikasi jaringan tersebut terkoneksi ke dalam aplikasi *Telegram* tersebut. Seperti pada **Gambar 4.19** sebagai berikut:



Gambar 4. 19 Tampilan Device Notifications

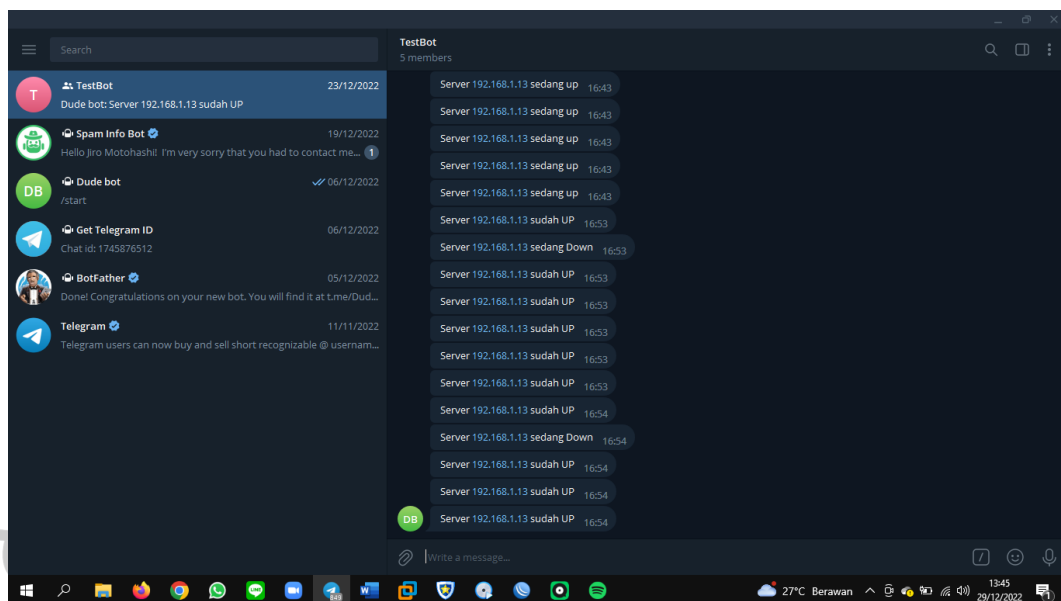
Kemudian nyalakan notifikasi *Telegram* pada *Device Service* seperti pada Gambar 4.20 sebagai berikut:



Gambar 4. 20 Jenis Notifikasi dalam Device Service

Jika sudah dinyalakan notifikasi tersebut, *Admin* mencoba untuk mematikan *IP Server* tersebut untuk melakukan tes uji coba pada notifikasi jaringan dengan cara mematikan *Device* “ping”. Setelah dimatikan akan muncul notifikasi jaringan yang sudah terkonfigurasi dari aplikasi *The Dude*. Berikut adalah hasil notifikasi jaringan yang sudah masuk ke dalam aplikasi

Telegram. Dalam hasil tersebut terdapat dua notifikasi yaitu “*Server 192.168.1.13 sudah UP*” yang dimana merupakan bahwa *Server* sudah bisa digunakan apabila *Device* “ping” dinyalakan. Namun, apabila *Device* “ping” dimatikan maka akan muncul notifikasi dengan perintah “*Server 192.168.1.13 sedang Down*”. Berikut adalah hasil yang sudah dibuat ada pada **Gambar 4.22** sebagai berikut:



**Gambar 4. 21 Hasil Notifikasi Jaringan**

Gambar diatas tersebut menjelaskan bahwa hasil notifikasi sudah terkoneksi ke dalam grup BOT Telegram tersebut yang mengatakan bahwa “*Server 192.168.1.13 sudah UP*” yang artinya bahwa *server* tersebut sudah membaik atau terkoneksi kembali. Dan notifikasi yang mengatakan “*Server 192.168.1.13 sedang Down*” mengatakan bahwa *server* sedang mengalami gangguan pada *IP Server* tersebut sehingga akses jaringan tidak bisa terkoneksi pada alat perangkat yang ada.