

## **BAB 4 ANALISIS KEBUTUHAN**

### **4.1 Deskripsi Umum**

Deskripsi umum ini akan menjelaskan beberapa poin yaitu tujuan, kegunaan, karakteristik pengguna, lingkungan pengguna, asumsi dan ketergantungan. Adapun pengertian dari masing-masing pengguna akan dijelaskan pada sub-bab berikut ini.

#### **4.1.1 Tujuan**

Pada bab ini akan menjelaskan mengenai sistem kendali pompa air secara *wireless* dimana dengan diterapkannya sistem *mark pump* ini diharapkan mampu untuk membantu pekerja pabrik pompa air PTPN X PG Pesantren Baru, Kediri dalam pekerjaannya. Penjelasan yang lebih terperinci akan diterangkan kegunaan dari sistem *mark pump*. Selain kegunaan, terdapat juga karakteristik pengguna, lingkungan serta asumsi ketergantungan sistem yang akan dijelaskan. Pada penelitian ini memiliki beberapa poin yang harus dicapai berdasarkan analisis kebutuhan, yakni kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional yang merupakan tujuan utama dari penelitian ini, dan kebutuhan non-fungsional merupakan kebutuhan untuk mendukung fungsi dari tujuan utama.

#### **4.1.2 Kegunaan**

*Mark pump* berguna di dalam sistem kendali jarak jauh. Pada penelitian ini sistem yang dibangun berfungsi untuk melakukan kendali pompa air yang terdapat pada PG Pesantren Baru, Kediri secara *wireless*. Penelitian ini juga nantinya diharapkan mampu untuk memangkas waktu yang terpakai untuk mengendalikan pompa. Penggunaan sistem *mark pump* nantinya dapat menjadi acuan untuk pengembangan lebih lanjut terkait pengendalian jarak jauh dalam satu lingkup industri. Penelitian ini juga bertujuan untuk membuktikan bahwa penerapan sistem cerdas dengan menggunakan *relay* dapat bekerja, sesuai dengan hasil dari penelitian yang sudah dilakukan sebelumnya

#### **4.1.3 Karakteristik Pengguna**

Pada penerapannya sistem dikendalikan oleh pekerja pabrik melalui *smartphone* dengan cara menjalankan aplikasi yang telah dibuat sebelumnya. Pekerja pabrik dapat mengendalikan pompa dengan cara menghidup dan mematikan pompa. Selain itu pekerja pabrik juga berperan dalam mendapatkan data yang terjadi selama proses berlangsung. Setelah data diterima oleh pekerja pabrik, data tersebut menjadi penilaian untuk menentukan kinerja sistem. pekerja pabrik juga dapat memeriksa secara langsung ke lapangan apakah pompa bekerja sesuai dengan perintah yang telah diberikan.

#### 4.1.4 Lingkungan Pengguna

Agar sistem dapat berjalan sesuai dengan tujuan penelitian, ada beberapa persyaratan yang harus terpenuhi beberapa diantaranya:

1. Jarak antara pekerja pabrik (smartphone) dengan *node* pusat tidak terlalu jauh. Artinya jarak keduanya tidak lebih dari jarak jangkauan router. Dalam hal ini router digunakan sebagai server.
2. Setiap *node* yang berfungsi sebagai perantara data tidak terhalang oleh tembok.
3. Pengujian dilakukan di PTPN X PG Pesantren Baru, Kediri. Dengan menggunakan 7 pompa yang terdapat di sekitar pabrik.

#### 4.1.5 Kebutuhan Fungsional

Dalam penelitian ini target yang harus terpenuhi adalah sistem mampu mengendalikan pompa air dari jarak jauh secara *wireless*. Untuk mengendalikan pompa *smartphone* pekerja pabrik harus mampu memberikan nilai ON atau OFF kepada masing-masing pompa air. Oleh karena itu pada penelitian ini tuntutan untuk:

1. Smartphone dapat memberikan nilai ke *node* pusat sesuai dengan format yang telah ditentukan sebelumnya
2. *Node* pusat meneruskan nilai yang diterima dari smartphone ke *node* pompa yang sudah tersebar di area pabrik
3. *Node* pompa mampu menerima dan berperilaku sesuai dengan data yang telah dikirim *node* pusat

#### 4.1.6 Kebutuhan Non Fungsional

Untuk mendukung performa dari sistem ini, maka terdapat kebutuhan non-fungsional. Berikut akan dijelaskan beberapa kebutuhan non-fungsional.

1. Pekerja pabrik hanya dapat menghidupkan pompa apabila tombol ON yang terdapat pada aplikasi *smartphone* kondisi *enable* adalah *true*
2. Pekerja pabrik hanya dapat mematikan pompa apabila tombol OFF yang terdapat pada aplikasi *smartphone* kondisi *enable* adalah *true*

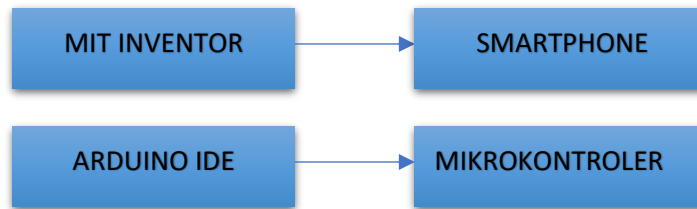
Pada poin 1 dan 2 dilakukan agar pekerja pabrik tidak bingung melakukan pengendalian pompa, karena sistem sudah menunjukkan pompa mana yang sedang aktif

#### 4.2 Kebutuhan Antarmuka Eksternal

Pada sub-bab ini akan menjelaskan tentang antarmuka eksternal yang dibutuhkan untuk membangun sistem *mark pump*. Kebutuhan antarmuka eksternal terbagi menjadi dua, yaitu perangkat keras dan perangkat lunak. Berikut akan dijelaskan kebutuhan tersebut secara bertahap.

### 4.2.1 Kebutuhan Perangkat Lunak

Perangkat lunak merupakan kebutuhan yang penting di dalam membangun sebuah sistem cerdas. Oleh karena itu pada saat mengerjakan penelitian ini dibutuhkan beberapa perangkat lunak untuk merancang sampai menganalisis sistem. Seperti yang terdapat pada Gambar 4.1 Tugas perangkat lunak masing-masing perangkat lunak memiliki tugas untuk memprogram masing-masing perangkat keras yang berbeda. Berikut akan dijelaskan perangkat lunak yang digunakan selama pengerjaan penelitian ini.

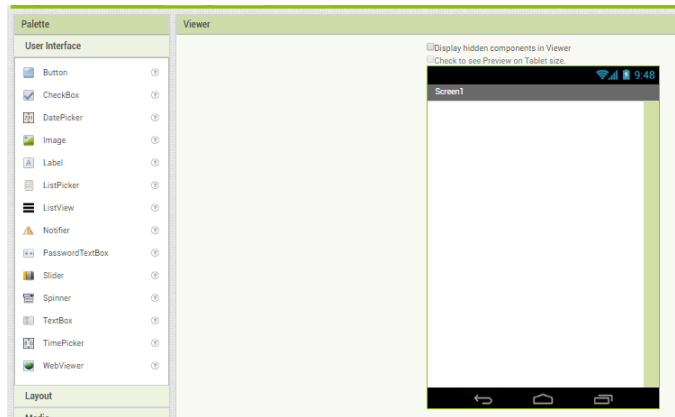


Gambar 4.1 Tugas perangkat lunak

#### 4.2.1.1 App Inventor

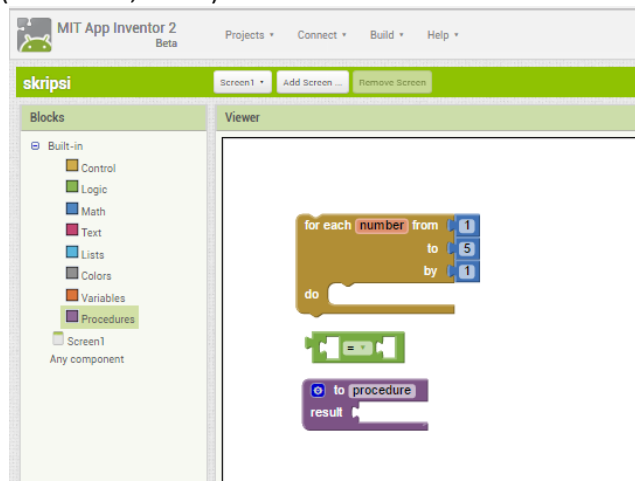
Pada penerapannya aplikasi yang digunakan harus bisa diaplikasikan di seluruh smartphone yang memiliki sistem operasi android. Berdasarkan kebutuhan untuk membuat aplikasi, App Inventor merupakan pemrograman yang sesuai untuk merancang sebuah aplikasi android. App Inventor dikembangkan oleh Massachusetts Institute of Technology (MIT). App Inventor merupakan pemrograman visual dengan antarmuka drag-and-drop, hal ini diharapkan mampu membantu penulis merancang aplikasi yang memiliki nilai fungsional yang maksimal. Sebagai perangkat yang terus berkembang, aplikasi mobile juga harus dikembangkan dengan cepat untuk memenuhi permintaan pasar. App Inventor memiliki dua elemen utama yaitu designer component dan blok editor. App Inventor memberikan kemudahan bagi pengguna yang sudah atau belum memiliki pengalaman membuat aplikasi android. Pengguna dapat merancang komponen antarmuka aplikasi dan komponen yang tidak terlihat dan menghubungkannya dengan designer component.

Fungsi dari blok editor agar pengguna dapat menentukan sendiri perilaku sebuah aplikasi di dalam kondisi tertentu. Gambar 4.2 merupakan tampilan dari *component designer App Inventor*. Pada tahap ini pengguna dapat merancang tampilan sesuai dengan kebutuhan pengguna. Dengan menggunakan *component designer* dapat dirancang komponen yang terlihat, seperti menu atau tombol. Kita juga dapat membuat komponen yang tak terlihat pada tahap ini seperti penambahan komponen *web connection* atau suara dan menghubungkannya dengan komponen terlihat. Pengguna dapat melakukan *drag and drop* atau memasukkan tombol atau menu ke antarmuka *View*. Untuk mengubah ukuran, warna dan lainnya pada tombol atau menu pengguna dapat mengubahnya pada jendela *properties*.



**Gambar 4.2 App Inventor component designer**

**Gambar 4.3** merupakan tampilan dari *bloks editor*. Fungsi utama dari *block editor* adalah untuk menentukan perilaku sebuah tombol atau menu dalam suatu kondisi tertentu dan bagaimana aplikasi menanggapi suatu perintah. Kelebihan dari *App Inventor* adalah menyediakan antarmuka pemrograman visual. Bahasa pemrograman dalam bentuk *block* sangat membantu pembuat aplikasi dalam proses pembangunan aplikasi. Dengan adanya bantuan *drag and drop* pengguna tidak perlu lagi mengetik kode program, hal ini dapat mencegah terjadinya selahan penulisan sintak. Bentuk dari blok-blok sendiri berbeda, hal ini juga bertujuan untuk membantu pengguna dalam penyesuaian perintah dengan fungsi dari sebuah komponen (Hwansoo, 2015).

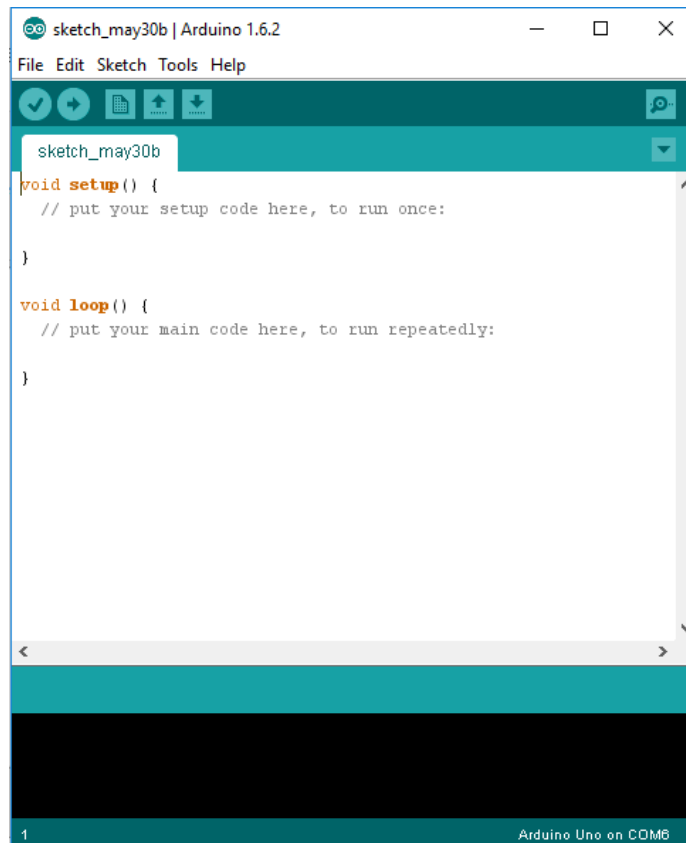


**Gambar 4.3 App Inventor blocks editor**

#### 4.2.1.2 Arduino Software (IDE)

*Integreted Development Enviroenment* (IDE) merupakan lingkungan terintegrasi yang dibangun untuk tujuan pengembangan. *Software* ini dikeluarkan oleh *arduino* untuk para pengembang sistem yang menggunakan *arduino* sebagai *microcontroller*. Dalam penggunaannya *arduino* menggunakan bahasa pemrograman sendiri yang mirip dengan bahasa c. Bahasa pemrograman yang digunakan memudahkan para pemula dalam mengembangkan idenya. *Sketch* merupakan sebutan untuk program yang ditulis dengan menggunakan *arduino*

*software* (IDE). Untuk ekstensi file yang disimpan adalah **.ino**. Untuk memudahkan para *programmer arduino software* memiliki fitur-fitur *cut*, *copy*, *paste*, dan yang lainnya. Pada bagian bawah *software arduino* IDE terdapat kotak berwarna hitam, kotak ini berfungsi untuk menampilkan status kode program, seperti pesan *error*, *compile*, dan *upload* program. Sedangkan pada bagian atas kanan terdapat gambar seperti kaca pembesar, tombol ini berfungsi sebagai penunjuk *biar* yang terkonfigurasi beserta *com ports* yang digunakan. Pada menu *file* terdapat *sketch example* yang tujuannya untuk memberikan arahan kepada pengguna *software* tentang cara konfigurasi *arduino* dengan perangkat lainnya.



**Gambar 4.4 Arduino software IDE**

Pada Gambar 4.4, terdapat 5 tombol yang masing-masing memiliki fungsi tersendiri. Tombol-tombol tersebut terdiri dari *verify*, *upload*, *new*, *open*, dan *save*.



Tombol *verify* tersebut digunakan untuk melakukan pemeriksaan terhadap kode program yang sudah dibuat terdapat *error* atau tidak, sehingga pengguna bisa menyimpulkan program sudah bisa diupload.



Tombol *upload* merupakan tombol yang berfungsi untuk menerjemahkan Bahasa program yang telah dibuat agar dapat dipahami oleh *board arduino*.



Tombol *new* merupakan tombol yang berfungsi untuk membuka lembar pekerjaan (*sketch*) baru



Tombol *open*, untuk membuka *file sketch* yang sudah pernah dibuat sebelumnya



Tombol *save*, untuk menyimpan *file sketch* yang sudah dikerjakan.



Tombol Serial Monitor, merupakan tombol yang sangat penting ketika sedang melakukan pengujian. Karena dengan menggunakan Serial Monitor kita bisa mengetahui data apa yang sedang dikerjakan oleh *board arduino*.

## 4.2.2 Kebutuhan Perangkat Keras

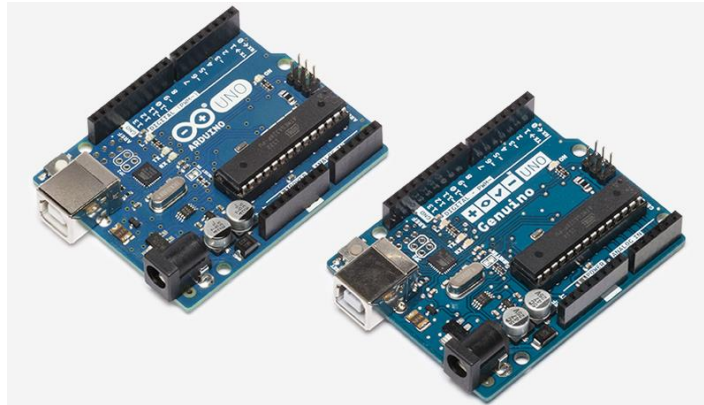
Kebutuhan perangkat keras merupakan kebutuhan yang nantinya akan berhubungan langsung dengan pompa air. Perangkat keras tersebut meliputi *router, smartphone, arduino uno* dan *nano*, serta *relay* yang akan berhubungan dengan power dari pompa air. Pada sub bab akan dijelaskan lebih *detail* mengenai perangkat-perangkat tersebut.

### 4.2.2.1 Microcontroller

Untuk menampung seluruh program yang akan dijalankan dibutuhkan sebuah komputer mini yang dikenal dengan nama *microcontroller*. *Microcontroller* merupakan sebuah sistem komputer yang didalamnya terdapat prosesor, memori sebagai penyimpan program dan berbagai perlengkapan masukan ataupun keluaran. Pada pengerjaan skripsi ini menggunakan dua buah *microcontroller* yaitu *arduino uno* dan *arduino uno*.

#### a. Arduino Uno

*Arduino uno* merupakan papan sirkuit yang berbasis *microcontroller* Atmega328. *Microcontroller* ini memiliki 14 *pin* digital masukan atau *keluaran* dimana 6 diantaranya dapat digunakan sebagai *keluaran* PWM (*Pulse With Modulation*), 6 analog masukan, pembangkit sinyal 16 Mhz, *port usb*, soket untuk adaptor, *pin* ICSP (*In Circuit Serial Programming*) dan tombol *reset*. Beberapa hal inilah yang dibutuhkan sebagai pembantu *microcontroller* dapat terhubung dengan kabel power *usb* atau kabel power *supply* adaptor AC ke DC atau juga baterai. Pada Gambar 4.5 menunjukkan bentuk dari *arduino uno*



**Gambar 4.5 Arduino uno**

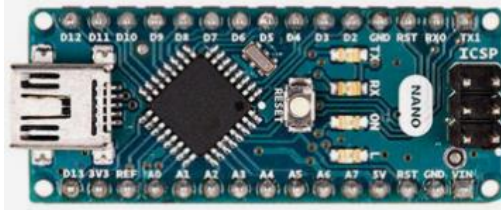
sumber : *Arduino.cc* (2017)

*Arduino uno* berbeda dengan beberapa versi yang dikeluarkan awal-awal dimana *microcontroller* versi sebelumnya tidak menggunakan *chip* khusus FTDI USB to serial. Sebagai penggantinya *chip* untuk FTDI diterapkan pada ATmega16U2 versi R2 (versi sebelumnya ATmega8U2). *Arduino uno* Rev. 2 dilengkapi resistor 8U2 ke *pin ground* yang lebih mudah diberikan ke mode DFU. Masing-masing dari 14 *pin* yang terdapat di *arduino uno* dapat digunakan sebagai masukan atau *keluaran*. Untuk menentukan fungsi *pin* dapat menggunakan perintah *pinMode()*, *digitalWrite*, dan *digitalRead*. Masing-masing perintah bekerja pada tegangan operasi 5V. *Arduino Uno* memiliki *pin* MOSI dan MISO dimana masing-masing *pin* terdapat pada *pin* 11 dan 12. *Pin* MOSI merupakan kependekan dari Master Out Slave In, sesuai dengan pengertiannya, *pin* ini berfungsi sebagai jalur pengirim data. *Pin* MISO kependekan dari Master In Slave Out, *pin* ini berfungsi sebagai penerima data. *Pin* SCK yang dalam ini berfungsi sebagai pengendali agar sewaktu digunakan secara bersamaan data yang melalui *pin* MOSI dan MISO tidak bertabrakan, *pin* SCK terletak pada *pin* 13. Sementara *pin* 2-10 dapat digunakan sebagai digital *input* dan *output*.

b. *Arduino Nano*

*Microcontroller* ini merupakan sebuah *board* yang dikeluarkan *arduino* dengan menggunakan *chip* ATmega328. Secara keseluruhan hampir sama dengan *Arduino uno*. *Arduino* ini menggunakan kabel *usb* mini-b, bukan kabel *usb standart*. Pada *board*-nya *arduino nano* memiliki 14 *pin* dimana *pin-pin* tersebut dapat digunakan sebagai digital masukan ataupun digital *keluaran* dengan menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*. Beroperasi pada tegangan 5V. Setiap *pin arduino* dapat menerima atau menyediakan maksimal 40 mA dan juga memiliki resistor *pull-up internal* 20-50 kOhms. *Arduino nano* juga memiliki beberapa *pin* yang memiliki

fungsi khusus beberapa diantaranya Serial: 0(RX) dan 1(TX). Sebagai penerima (RX) dan pengirim (TX) TTL serial data.

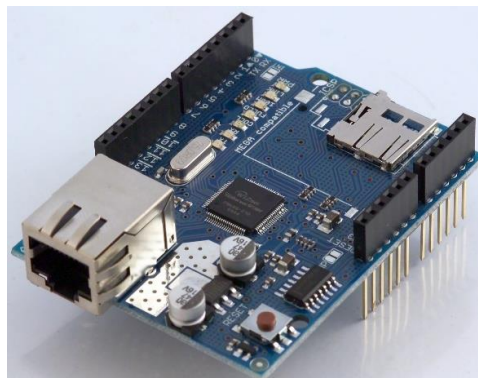


**Gambar 4.6 Arduino nano**

Gambar 4.6 merupakan bentuk dari *arduino nano*, eksternal *Interrupt*: 2 dan 3. *Pin* ini berfungsi sebagai konfigurasi *trigger* ketika terjadi interupsi value rendah, tinggi dan tepi (berubah-ubah). PWM: 3,5,6,9,10,11. Berfungsi untuk melayani *keluaran* 8-bit PWM dengan menggunakan fungsi *analogWrite()*. SPI: 10(SS), 11 (MOSI), 12 (MISO), 13 (SCK). *Pin* yang mendukung komunikasi SPI menggunakan *SPI library*. LED: 13. *Pin* ini merupakan indikator bawaan (*built-in*), dimana ketika nilai *value*=HIGH maka *led* akan ON, saat *value*=LOW maka *led* akan OFF

#### 4.2.2.2 Ethernet Shield

Merupakan board yang disediakan oleh arduino agar sistem tertanam dapat terhubung ke internet. Berdasarkan kebutuhan yang akan menghubungkan node pusat ke smartphone pengguna, maka pada penelitian ini akan menghubungkan antara ethernet shield dengan arduino.



**Gambar 4.7 Ethernet Shield**

Gambar 4.7 merupakan Ethernet shield, beroperasi dengan tegangan 5V, dan pengendali ethernet tipe W5500 dengan buffer internal 32K. perangkat ini bekerja dengan kecepatan 10-100Mb dan dikoneksikan dengan arduino pada port SPI.

#### 4.2.2.3 Modul Radio nRF24L01

Pada penelitian ini menggunakan sebuah modul wireless nRF24L01. nRF24L01 merupakan modul komunikasi jarak jauh yang bekerja pada frekuensi 2.4 GHz secara *wireless* dengan sedikit mengonsumsi daya. Modul tersebut dapat digunakan untuk komunikasi dua arah yaitu mengirim (*transmitter*) dan menerima



(receiver). Penggunaan daya dari nRF24L01 tergantung dari jarak *node* yang ingin dihubungkan. Arsitektur tersebut memenuhi persyaratan daya rendah dan ukurannya yang minimalis sehingga memudahkan alat tersebut untuk diletakkan dengan perangkat-perangkat yang lain.



**Gambar 4.8** Antar muka modul nRF24L01

sumber : Nordic (2016)

Gambar 4.8 merupakan modul nRF24L01. Menurut Unsal et al. (2016), algoritma daya adaptif yang menggunakan kedua *keluaran* daya RF dan tingkat transmisi harus disesuaikan dengan jarak antara *node* yang akan memaksimalkan efektivitas penggunaan daya. Hal tersebut memungkinkan nRF24L01 bertahan lama karena pemakaian dayanya yang rendah. Pada nRF24L01 terdapat *pipe* yang berfungsi sebagai saluran logika fisik frekuensi radio dimana setiap saluran memiliki alamat penguraian fisik tersendiri.

#### 4.2.2.4 Relay

Untuk menggerakkan pompa yang memiliki masukan tegangan 360V dibutuhkan sebuah *relay* yang mampu untuk mengendalikan pompa air. Secara umum tugas relai pada penelitian ini adalah untuk menggerakkan *relay* yang mendukung daya 360V. *Relay* dengan spesifikasi yang sesuai dengan *arduino nano* adalah *relay Sngle* dengan tipe SRD-5VDC-SL-C. Pemilihan relai tersebut berdasarkan *keluaran* yang terdapat pada *arduino nano* dengan tegangan 5V. Relai tipe ini memiliki VCC 5V. Keluaran dari relay ini sampai 250V atau 30VDC.



**Gambar 4.9** Relay

Pada *board relay* Gambar 4.9 terdapat 3 *pin* yang akan dihubungkan ke *arduino*, dimana 1 *pin ground*, 1 *pin VCC* dimana VCC tersebut dihubungkan dengan *pin 5V* yang terdapat pada *arduino*. Sedangkan *pin in relay* akan dihubungkan dengan *pin*

*keluaran* pada *arduino*. Untuk *keluaran* dari *relay* sendiri memiliki 2 pilihan, yakni *normally open* (NO) dan *normally closed*. Untuk penjelasan dari masing-masing tipe ini sudah dijelaskan pada bab 3, bagian penjelasan *relay*.

#### **4.2.2.5 Router**

Penelitian ini akan menghubungkan *smartphone* dengan internet. Sebagai pemancar dari sinyal internet dibutuhkan sebuah *router*. Wireless pada Gambar 4.10, memiliki kecepatan 150Mbps. Sangat baik digunakan sehari-hari tanpa mengistirahatkan sama sekali. Sesuai dengan standar IEEE 802. 11n, TD-W8951 ND dapat membangun jaringan *wireless* yang lebih baik, tingkat transmisi, stabilitas dan jangkauan.



**Gambar 4.10 Router**