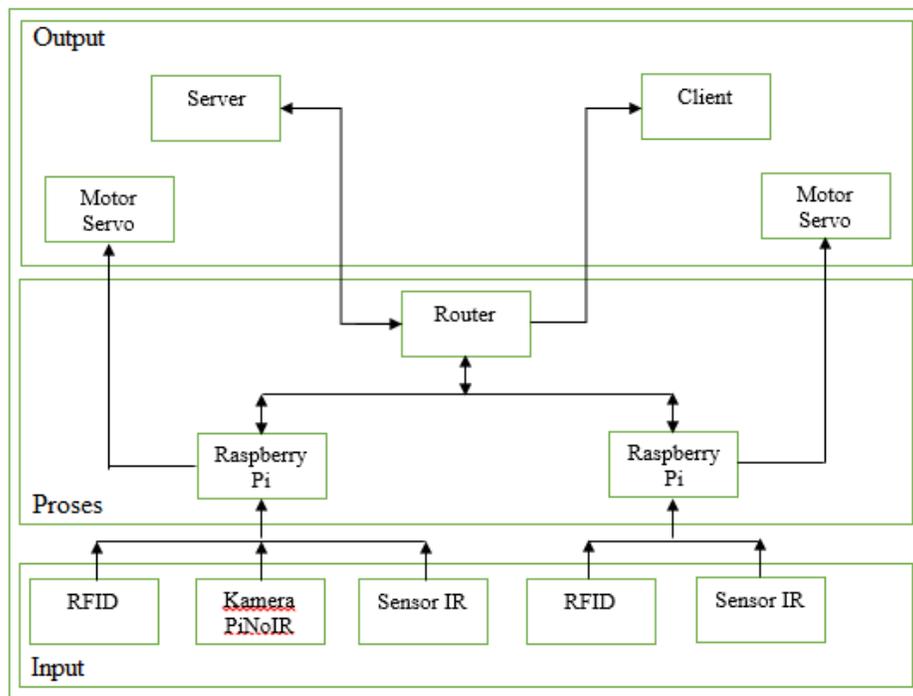


BAB III

PERANCANGAN ALAT

3.1 Blok Diagram Sistem

Pada sistem keamanan parkir berbasis PC menggunakan *Camera* dan RFID (*Radio Frequency Identification*) ini, terdapat blok diagram yang didalamnya terdapat beberapa komponen yang digunakan untuk sistem keamanan parkir ini seperti berikut.



Gambar 3.1 Blok Diagram Sistem

Secara keseluruhan cara kerja dari sistem parkir ini berfungsi pada saat menerima masukan dari *tag* RFID yang kemudian diproses oleh Raspberry. Pada blok diagram sistem untuk keluaran dimana setelah mikroprocessor

memproses data akan dikeluarkan pada blok diagram keluaran ini. Terdapat beberapa komponen pada blok diagram keluaran ini yaitu motor servo yang berfungsi untuk membuka dan menutup portal, lalu yang terakhir keluaran akan masuk ke sebuah komputer dimana semua data pengendara yang terdaftar akan diproses di dalam sebuah *software* komputer *server* yang bisa di *sharing* dengan komputer *client* menggunakan *router*. Berikut penjelasan masing-masing komponen pada blok diagram diatas.

1. Masukan (*input*)

Pada blok diagram diatas, terdapat beberapa komponen *input* yaitu.

a. RFID

RFID merupakan salah satu teknologi favorit saat ini, RFID bekerja dengan cara menggunakan gelombang radio yang dapat berfungsi untuk mengidentifikasi suatu objek atau manusia.

b. Kamera

Kamera Pi NoiR berfungsi untuk mengambil gambar suatu objek atau manusia.

c. Sensor IR

Sensor IR berfungsi untuk mendeteksi suatu objek atau benda yang lewat.

2. Proses

Pada blok diagram diatas, terdapat satu komponen proses yaitu.

a. Raspberry Pi

Sebagai komponen utama, Raspberry Pi digunakan untuk mengontrol dan memproses sistem keamanan parkir ini.

3. Keluaran (*output*)

Pada blok diagram diatas, terdapat beberapa *input* yaitu.

a. Motor Servo

Motor Servo berfungsi untuk menggerakkan portal atau palang parkir pada sistem keamanan parkir ini.

b. Server

Server berfungsi untuk menyimpan database mahasiswa yang sudah didaftarkan yang nantinya data tersebut akan dikirimkan ke *client*.

c. Router

Router berfungsi untuk mengirimkan data yang terdapat pada server ke *client*.

d. Client

Client berfungsi untuk menampilkan data yang dikirimkan oleh *server* melalui *router*.

3.2 Perancangan Sistem

Cara kerja dari sistem keamanan parkir berbasis Raspberry Pi menggunakan *Camera* dan RFID ini yaitu data pengendara seperti foto wajah, nama, NIM, prodi dan plat nomor kendaraan pengemudi tersebut sudah terlebih dulu dimasukkan ke database melalui kartu RFID. Kartu RFID tersebut nantinya akan diberikan kepada pengemudi yang telah terdata pada *database*.

Kartu RFID digunakan saat pengemudi akan parkir dengan cara kartu tersebut ditempelkan pada RFID *reader* yang telah terlebih dahulu tersedia disekitar pintu masuk area parkir. Setelah kartu RFID berhasil terbaca, maka *Camera* akan secara otomatis mengambil gambar pengemudi yang akan masuk ke area parkir yang nantinya gambar dan yang telah berhasil diambil akan dikirimkan ke Raspberry Pi yang nantinya akan di proses.

Pengemudi diperbolehkan masuk ke area parkir apabila portal atau palang parkir terangkat secara otomatis menggunakan motor servo. Portal atau palang parkir akan terangkat secara otomatis apabila perintah-perintah sebelumnya berhasil dilakukan. Setelah pengemudi melewati palang parkir, maka sensor PIR akan mendeteksi ada tidaknya kendaraan yang melewati portal atau palang parkir, apabila ada kendaraan yang terdeteksi maka sensor IR akan memberikan perintah yang nantinya diproses oleh Raspberry Pi agar portal atau palang parkir dapat tertutup secara otomatis.

Saat pengemudi akan keluar dari area parkir, kartu RFID yang tadi digunakan pada pintu masuk kembali digunakan pada pintu keluar. Kartu RFID ditempelkan kembali pada RFID *reader* yang telah tersedia disekitar pintu keluar. Setelah kartu

RFID berhasil terbaca, maka pada layar monitor akan menampilkan informasi berupa data pengemudi yang mencakup nomor plat, nama, NIM dan foto pemilik kartu RFID. Gambar yang diambil dari pintu masuk juga akan ditampilkan agar petugas parkir dapat mencocokkan identitas pengemudi dan kendaraan yang masuk dan keluar dari area parkir.

3.3 Pemilihan Komponen

Pemilihan jenis komponen yang akan digunakan pada sistem keamanan parkir berbasis Raspberry Pi menggunakan *Camera* dan RFID ini berdasarkan perbandingan dari komponen-komponen yang telah digunakan sebelumnya yang berkaitan dengan sistem keamanan parkir ini yang berbeda dari segi spesifikasinya dan harga yang dapat mempengaruhi tingkat keberhasilan komponen yang digunakan.

3.3.1 *Radio Frequency Identification (RFID)*

Pada bagian RFID, ada dua komponen penting yang akan digunakan untuk sistem keamanan parkir berbasis Raspberry Pi menggunakan *Camera* dan RFID, yaitu.

a. RFID *tag*

Pemilihan jenis kartu RFID merupakan salah satu bagian paling penting karena kartu RFID berfungsi sebagai masukan (*input*) pada sistem keamanan parkir ini. RFID *tag* yang digunakan berupa *tag* pasif. Berikut merupakan perbandingan kartu RFID *tag* pasif yang ada dipasaran.

Tabel 3.1 Spesifikasi RFID *tag* pasif

Spesifikasi	Jenis Tag		
	Tag Pasif		
Frekuensi Kerja	LF (125-134 Khz)	HF (13,56 Mhz)	UHF (868-956 Mhz)
	GK4001/EM4001	ISO15693	S250
Jarak Baca	Max 16 CM	Max 30 cm	90 cm-4,5m
Dimensi	86 x 54 x 1,9mm	86 x 54 x 1,9mm	86 x 54 x 0,84mm
Harga	Rp. 10.000	Rp. 10.000	Rp. 15.000

RFID *tag* yang digunakan pada sistem keamanan parkir ini adalah ISO15693 yang berjumlah 10 buah kartu. RFID *tag* ISO15693 dipilih karena kartu tersebut cocok digunakan pada RFID *reader* yang nantinya akan dipakai.

b. RFID *reader*

Pemilihan RFID *reader* yang digunakan juga penting untuk sistem keamanan parkir ini karena berfungsi sebagai pembaca data dari kartu RFID yang digunakan. RFID *reader* yang digunakan yaitu Mifare RC522 yang berjumlah 2 buah RFID *reader* yang nantinya akan diletakkan di area sekitar pintu masuk dan pintu keluar parkir. Berikut spesifikasi RFID *reader* Mifare RC522.

Tabel 3.2 spesifikasi Mifare RC522

Nama Produk	Mifare RC 522
Frekuensi Kerja	13,56 Mhz
Jarak Baca	< 50 mm
Format Kartu	S50/S70, ISO 15693
Catu Daya	3,3 V
Dimensi	40 x 50 mm
Harga	Rp. 40.000

3.3.2 Kamera PiNoIR

Kamera PiNoir merupakan kamera khusus yang digunakan pada Raspberry Pi sehingga kamera jenis ini dipilih untuk pembuatan sistem keamanan parkir berbasis Raspberry Pi menggunakan *Camera* dan RFID. Kamera Pi NoIR mempunyai tingkat kualitas resolusi gambar sebesar 5 MP dengan kecepatan fps (*frame per second*) mencapai 60 fps.

Kamera Pi Noir nantinya akan difungsikan seperti *Camera* sebagai pengambil gambar dan video pengemudi yang masuk area parkir. Berikut merupakan table spesifikasi kamera Pi NoIR.

Tabel 3.3 spesifikasi kamera PiNoIR

Nama Produk	Kamera Pi NoIR
Resolusi	5 MP
Resolusi Gambar	2592x1944
Video	1080p @30fps, 720p @60fps dan 640x480p 60/90 Recording
Dimensi	20 x 25 x 9 mm
Harga	Rp. 150.000

3.3.3 Motor Servo

Motor servo digunakan untuk menggerakkan portal atau palang parkir pada sistem keamanan parkir ini. Banyak motor servo yang telah dijual dipasaran. Berikut merupakan beberapa jenis motor servo yang telah dijual di pasaran.

Tabel 3.4 Perbandingan Spesifikasi Motor Servo

Spesifikasi	Jenis Motor Servo		
	Tower Pro SG90	Tower Pro MG996R	HD- 3688 HB
Kecepatan	0,1 detik	0,17 detik	0,8 detik
Torsi	1,6 kg	13 kg-15 kg	2,3 kg-2,8 kg
Bahan	Nylon	Metal	Plastik
Tegangan	4,8 V	4,8 V	4,5 V
Rotasi	60 derajat	180 derajat	
Dimensi	23 x 29 x 12,2 mm	40,7 x 19,7 x 42,9 mm	40,3 x 20,2 x 26,1 mm
Harga	Rp. 40.000	Rp. 70.000	Rp. 400.000

Motor servo yang digunakan yaitu motor servo Tower Pro SG90 walaupun spesifikasinya kurang dibandingkan jenis motor servo yang lain tetapi motor servo jenis ini bisa digunakan untuk pembuatan sistem keamanan parkir ini dan harganya juga lebih murah dibandingkan jenis motor servo yang lain. Motor servo Tower Pro SG90 berjumlah 2 buah unit yang berfungsi untuk menggerakkan portal atau palang parkir pada pintu masuk dan pintu keluar.

3.3.4 Sensor IR

Sensor IR merupakan sensor inframerah yang digunakan untuk mendeteksi kendaraan yang melewati portal atau palang parkir. Sensor yang digunakan yaitu sensor infra merah (*Obstacle Avoidance Sensor*) karena sensor tersebut memiliki keluaran (*output*) berupa sinyal digital yang berupa high dan low dengan jarak sekitar 2 cm – 80 cm. Sensor tersebut memiliki sebuah LED yang berfungsi untuk mengidentifikasi suatu kendaraan yang melewati portal atau palang parkir. Sensor IR yang digunakan berjumlah 2 buah sensor yang diletakkan disekitar pintu masuk dan pintu keluar parkir.

3.3.5 Raspberry Pi

Raspberry Pi merupakan perangkat yang paling penting dalam sistem keamanan parkir ini karena semua perintah yang masuk akan di proses pada alat ini. Ada beberapa jenis Raspberry Pi yang beredar dipasaran, berikut merupakan contoh spesifikasi Raspberry Pi yang beredar di pasaran

Tabel 3.5 Perbandingan Model Raspberry Pi

Spesifikasi	Nama Produk			
	Model A	Model B	Model A+	Model B+
Chip	BRCM2835	BRCM2835	BRCM2835	BRCM2835
Standar SoC	700 Mhz	700 Mhz	700 Mhz	700 Mhz
RAM	256 MB	512 MB	256 MB	512 MB
Memori	Full SD	Full SD	Micro SD	Micro SD
Port USB 2.0	1	2	1	4
GPIO	17	17	26	26
Dimensi	85 x 56 x 15 mm	85 x 56 x 17 mm	65 x 56 x 12 mm	85 x 56 x 17 mm
Harga	-	Rp. 550.000	Rp. 500.000	-

Raspberry Pi yang akan digunakan untuk sistem keamanan parkir ini yaitu Raspberry Pi 3 Model B karena dapat digunakan untuk sistem keamanan parkir ini. Raspberry Pi 3 model B juga memiliki ukuran RAM yang lumayan besar dan mudah dicari di pasaran.

3.3.6 Router

Router merupakan perangkat jaringan komputer yang berfungsi untuk menghubungkan beberapa jaringan agar dapat mengirimkan data. Pada sistem keamanan parkir ini, jenis router yang digunakan adalah TP-Link model TL-WR840N.

Tabel 3.6 Spesifikasi Tp-link TL-WR840N

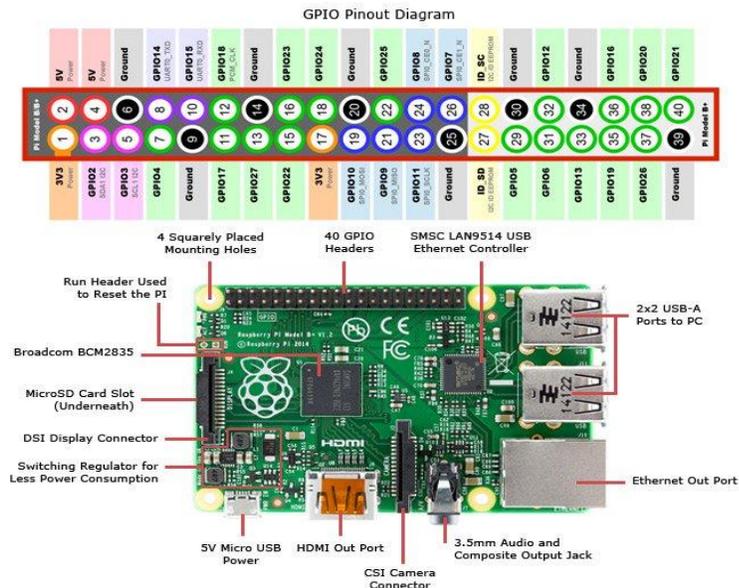
Nama produk	Tp-Link TL-WR840N
Kecepatan	300 Mbps
Port WAN	1
Port LAN	1
Dimensi	182 x 128 x35 mm
Harga	Rp. 155.000

3.4 Perancangan Perangkat Keras (*Hardware*)

Dalam perancangan sistem keamanan parkir berbasis Raspberry Pi menggunakan *Camera* dan RFID ini berupa prototype lokasi parkir untuk meletakkan komponen-komponen diatas.

3.4.1 Raspberry Pi 3 Model B

Raspberry Pi 3 model B ini berfungsi sebagai pengendali pada sistem keamanan parkir ini. Di bawah ini merupakan diagram pin dan port pada Raspberry Pi 3 model b.

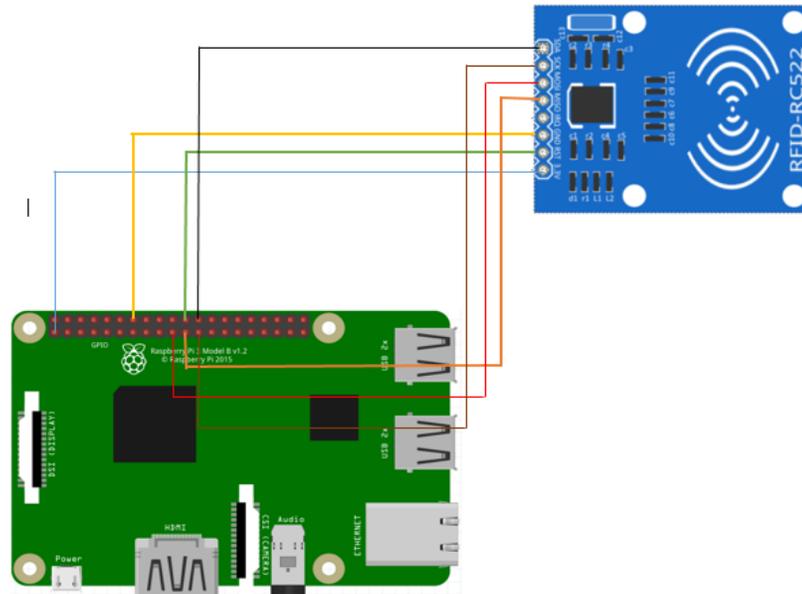


Gambar 3.2 Diagram pin dan port pada Raspberry Pi 3

Raspberry Pi 3 model B memiliki beberapa port yang nantinya akan dihubungkan ke perangkat yang lain yang digunakan pada sistem keamanan parkir ini. Berikut merupakan perancangan pin dan port dari Raspberry Pi yang akan dihubungkan ke perangkat lain.

3.4.2 RFID Reader Mifare RC522

RFID reader mifare RC522 berfungsi untuk membaca data yang telah terdaftar pada RFID card. RFID reader diletakkan didekat pintu masuk dan pintu keluar area parkir.

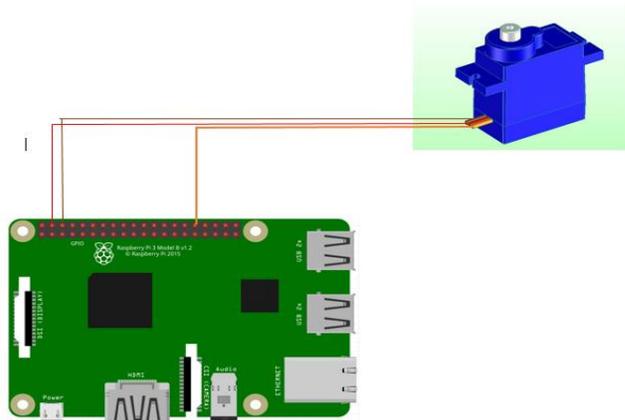


Gambar 3.3 RFID reader Mifare RC522 dihubungkan ke Raspberry Pi

Berdasarkan gambar 3.3 diatas, port SDA terhubung ke pin GPIO 8 pada Raspberry Pi dan port MISO terhubung pada pin GPIO 9.

3.4.3 Motor Servo

Motor servo berfungsi untuk menggerakkan portal parkir saat pengemudi masuk atau keluar, motor servo di letakkan di dekat posisi RFID reader.

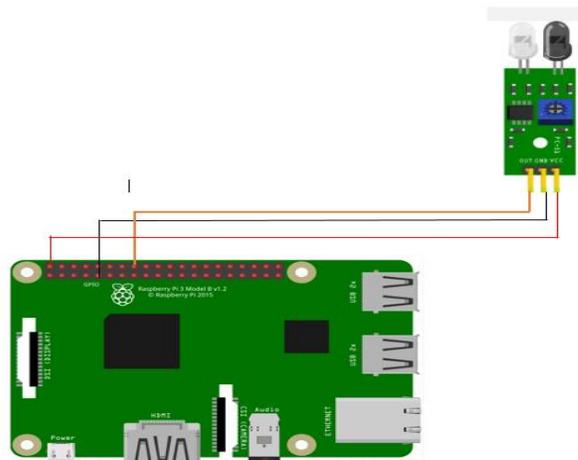


Gambar 3.4 Motor Servo dihubungkan ke Raspberry Pi

Berdasarkan gambar 3.4 diatas, port PWM pada motor servo dihubungkan ke pin GPIO 12 pada Raspberry Pi.

3.4.4 Sensor IR

Sensor IR berfungsi untuk mendeteksi apabila pengemudi lewat agar portal parkir dapat tertutup.

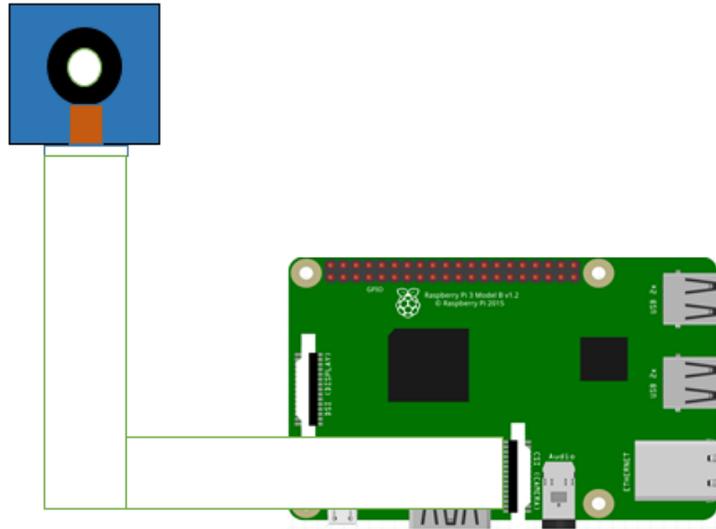


Gambar 3.5 Sensor IR dihubungkan ke Raspberry Pi

Berdasarkan gambar 3.5 diatas, port OUT pada sensor IR dihubungkan ke pin GPIO 23 pada Raspberry Pi.

3.4.5 Kamera PiNoIR

Kamera PiNoIR berfungsi untuk pengambilan gambar saat pengemudi masuk ke area parkir.

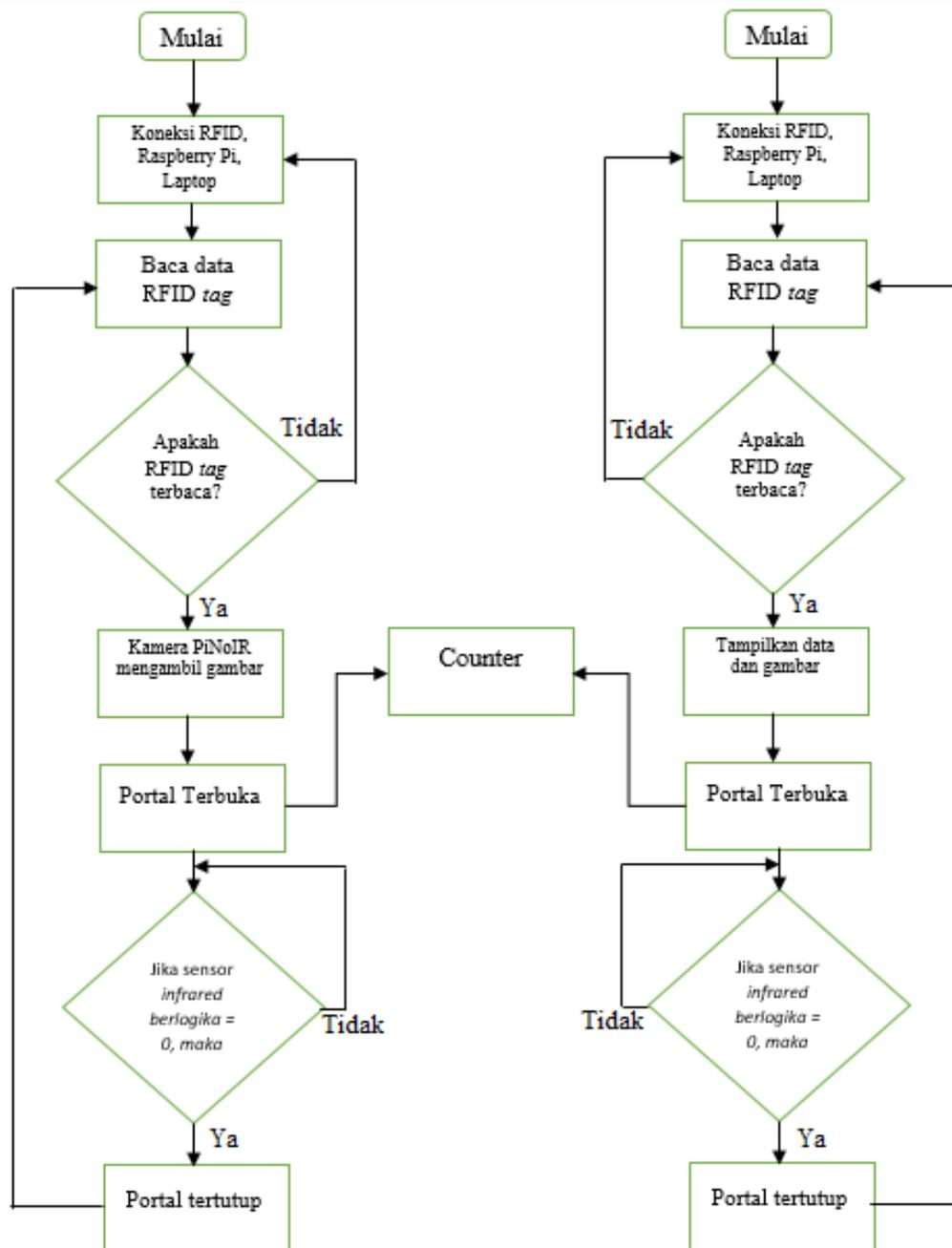


Gambar 3.6 Kamera PiNoIR dihubungkan ke Raspberry

Pada gambar 3.6 diatas, kamera PiNoIR dihubungkan ke port kamera yang sudah tersedia pada Raspberry.

3.5 Perancangan Perangkat Lunak (*Software*)

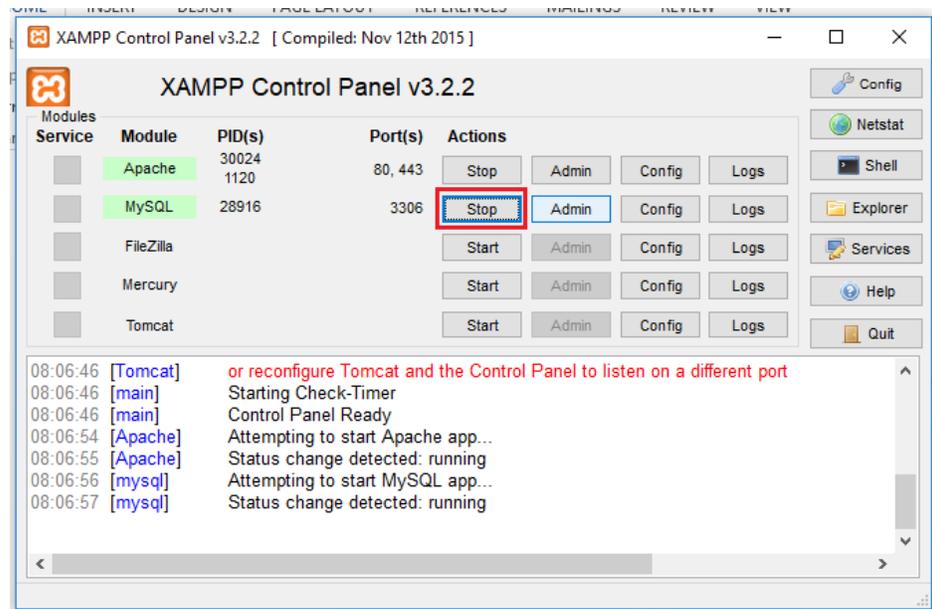
Perancangan perangkat lunak digunakan untuk menentukan fungsi-fungsi dan alur kerja sistem keamanan parkir berbasis Raspberry Pi menggunakan *Camera* dan *RFID* ini. Masukan (*input*) dari komponen-komponen tadi akan di proses oleh perangkat lunak yang nantinya *input* tersebut akan menentukan hasil keluaran (*output*) dari sistem yang telah dirancang. Berikut ini merupakan alur kerja (*flow chart*) dari sistem yang akan di rancang.



Gambar 3.7 Flow Chart sistem parkir

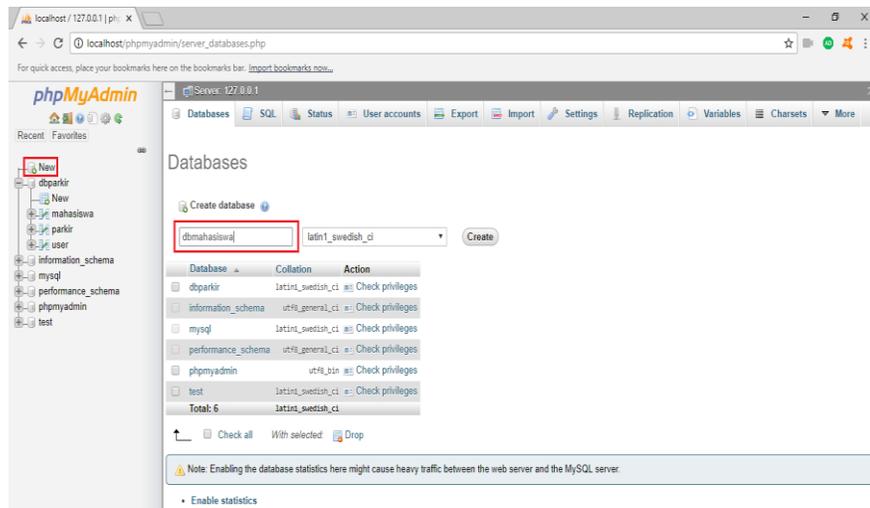
3.5.1 Perancangan Pendaftaran *Database* Mahasiswa

Pada perancangan pendaftaran *database* mahasiswa, terlebih dahulu mengaktifkan MySQL dengan cara klik *Start* pada bagian MySQL yang terdapat pada XAMPP seperti di perlihatkan pada gambar 3.8 dibawah ini.



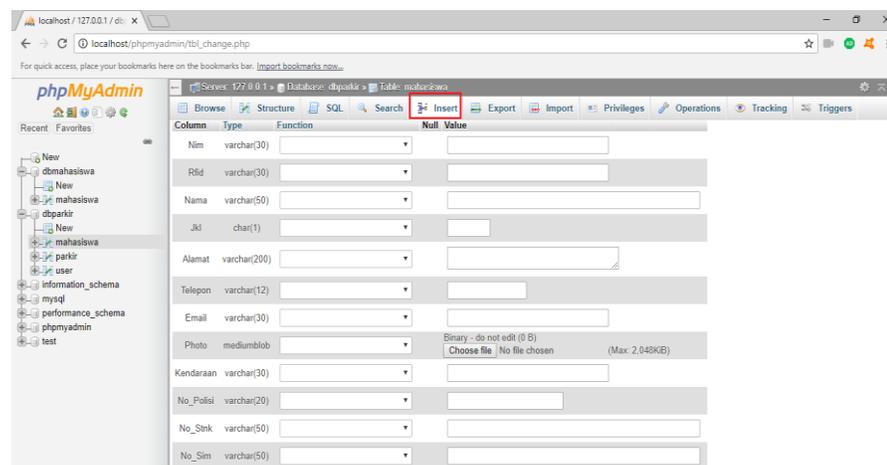
Gambar 3.8 Mengaktifkan MySQL

Setelah MySQL diaktifkan, selanjutnya klik *admin* yang terletak disebelah *stop* untuk membuat *database* pada halaman MySQL. Setelah halaman pembuatan *database* ditampilkan, selanjutnya klik *new* untuk membuat *database* baru dan masukkan nama *database* yang akan dibuat.



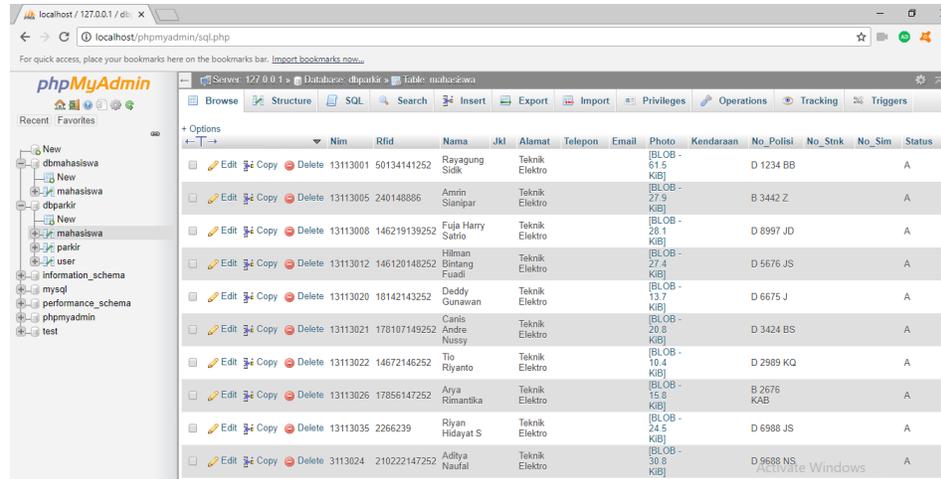
Gambar 3.9 Membuat *database* baru

Setelah *database* baru dibuat, kemudian masukkan beberapa jumlah kolom yang di butuhkan untuk memasukkan data seperti nama, nim, nomor rfid, nomor polisi dan foto. Setelah kolom dibuat, kemudia ke menu *insert* untuk memasukkan data pengemudi yang akan didaftarkan pada RFID *card*.



Gambar 3.10 Memasukkan Data Pengemudi

Data pengemudi yang sudah terdaftar ke *database* dapat dilihat pada gambar 3.11 dibawah ini.

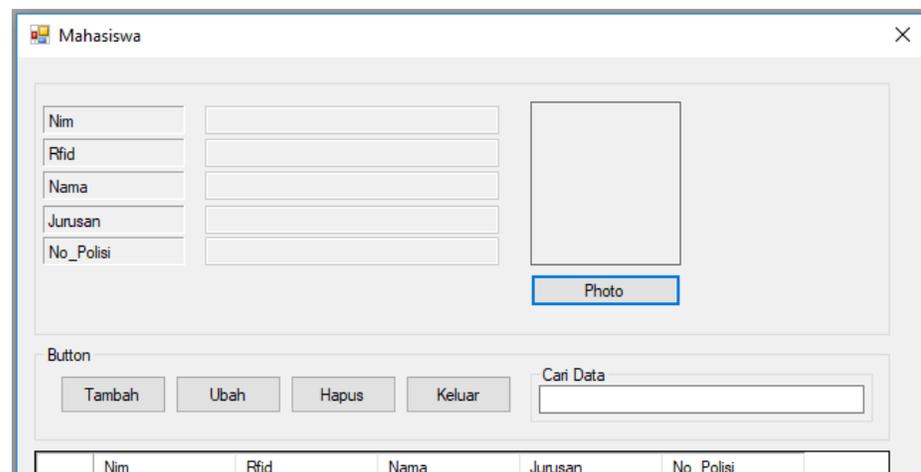


Nim	Rfid	Nama	Jkl	Alamat	Telepon	Email	Photo	Kendaraan	No_Polisi	No_Stnk	No_Sim	Status
13113001	50134141252	Rayagung Siak	Teknik Elektro				[BL.OB - 61.5 KB]		D 1234 BB			A
13113005	240148886	Amin Siangpar	Teknik Elektro				[BL.OB - 27.9 KB]		B 3442 Z			A
13113008	146219139252	Fuja Harry Saito	Teknik Elektro				[BL.OB - 28.1 KB]		D 8997 JD			A
13113012	146120148252	Haiman Bintang Fuadi	Teknik Elektro				[BL.OB - 27.4 KB]		D 5676 JS			A
13113020	18142143252	Deddy Gunawan	Teknik Elektro				[BL.OB - 13.7 KB]		D 6675 J			A
13113021	178107145252	Canis Andre Nussy	Teknik Elektro				[BL.OB - 20.8 KB]		D 3424 BS			A
13113022	14672146252	Tio Riyanto	Teknik Elektro				[BL.OB - 10.4 KB]		D 2989 KO			A
13113026	17856147252	Arya Rimantika	Teknik Elektro				[BL.OB - 15.8 KB]		B 2676 KAB			A
13113035	2266239	Ryhan Hidayat S	Teknik Elektro				[BL.OB - 24.5 KB]		D 6988 JS			A
3113024	210222147252	Aditya Naufal	Teknik Elektro				[BL.OB - 30.8 KB]		D 9688 NS			A

Gambar 3.11 Pengemudi yang sudah terdaftar

3.5.2 Perancangan Data Mahasiswa

Perancangan data mahasiswa untuk menambah, mengubah dan menghapus data dibuat menggunakan Visual Basic seperti di perlihatkan pada gambar 3.12 dibawah ini.



The form 'Mahasiswa' contains the following elements:

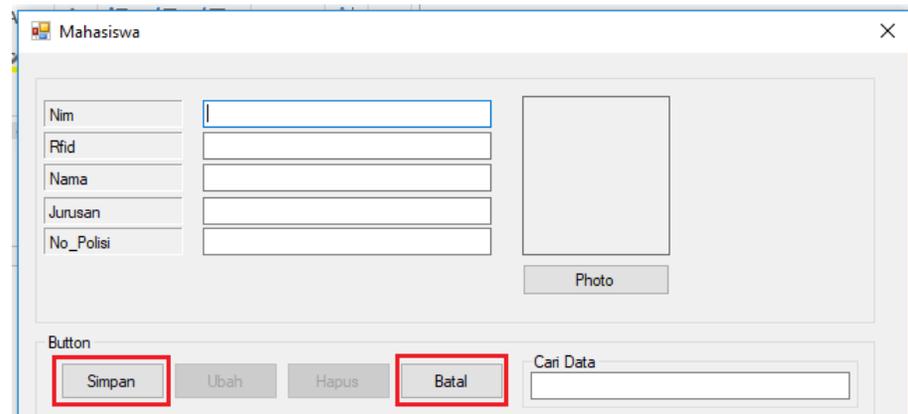
- Input fields for: Nim, Rfid, Nama, Jurusan, No_Polisi.
- A 'Photo' field with a 'Photo' button below it.
- A 'Button' section containing: Tambah, Ubah, Hapus, and Keluar.
- A 'Cari Data' search field.
- A table at the bottom with columns: Nim, Rfid, Nama, Jurusan, No_Polisi.

Gambar 3.12 Tampilan Data Mahasiswa

Pada gambar 3.12 diatas, terdapat beberapa tombol yang mempunyai fungsi masing-masing. Berikut fungsi dan cara menggunakannya.

1. *Input* data mahasiswa

Pada tampilan data mahasiswa terdapat tombol tambah yang berfungsi untuk meng-*input* data mahasiswa yang ingin mendaftar. Setelah tombol tambah di klik, maka akan muncul tampilan seperti gambar dibawah ini.



The image shows a software window titled "Mahasiswa" with a close button (X) in the top right corner. The window contains a form with the following elements:

- Five input fields labeled "Nim", "Rfid", "Nama", "Jurusan", and "No_Polisi" stacked vertically on the left.
- A large empty rectangular box on the right, with a "Photo" button centered below it.
- A "Button" section at the bottom containing four buttons: "Simpan", "Ubah", "Hapus", and "Batal". The "Simpan" and "Batal" buttons are highlighted with red rectangular boxes.
- A search field labeled "Cari Data" on the far right of the bottom section.

Gambar 3.13 Tampilan tambah data mahasiswa

Untuk menambahkan data mahasiswa, maka diperlukan nim, nomor RFID, nama, jurusan, nomor polisi dan foto agar mahasiswa dapat terdaftar. Setelah semua ditambahkan, klik simpan untuk meng-*input* data dan klik batal untuk membatalkan data yang telah diisi.

Nim	13113035	 Photo
Rfid	2266239	
Nama	Riyan Hidayat S	
Jurusan	Teknik Elektro	
No_Polisi	D 6988 JS	

Button

Tambah Ubah Hapus Keluar

Cari Data

Gambar 3.14 Mahasiswa yang telah terdaftar

2. *Edit* data mahasiswa

Pada tampilan data mahasiswa terdapat tombol ubah yang berfungsi untuk meng-*edit* data mahasiswa yang telah terdaftar. Setelah tombol ubah di klik, maka akan muncul tampilan seperti gambar dibawah ini.

Nim	13113035	 Photo
Rfid	2266239	
Nama	Riyan Hidayat S	
Jurusan	Teknik Elektro	
No_Polisi	D 6988 JS	

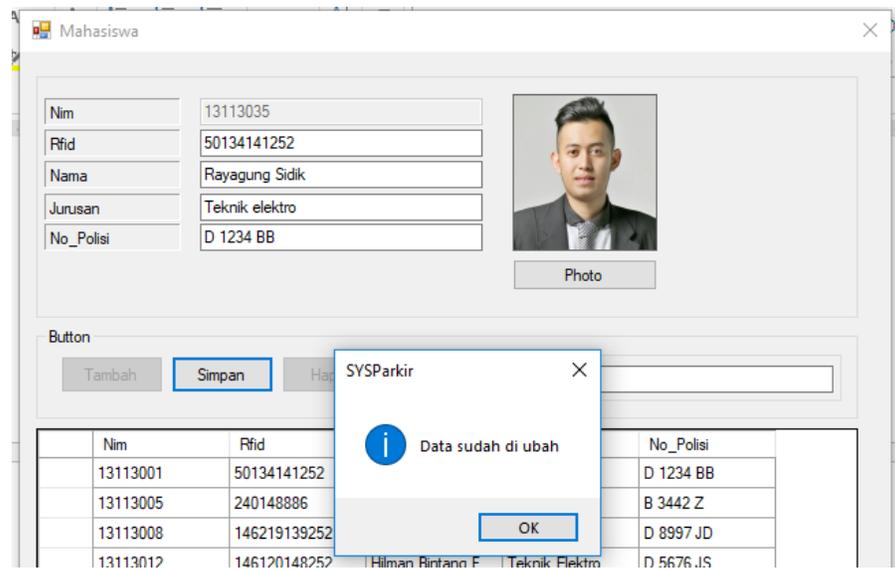
Button

Tambah Simpan Hapus Batal

Cari Data

Gambar 3.15 Mengubah data mahasiswa

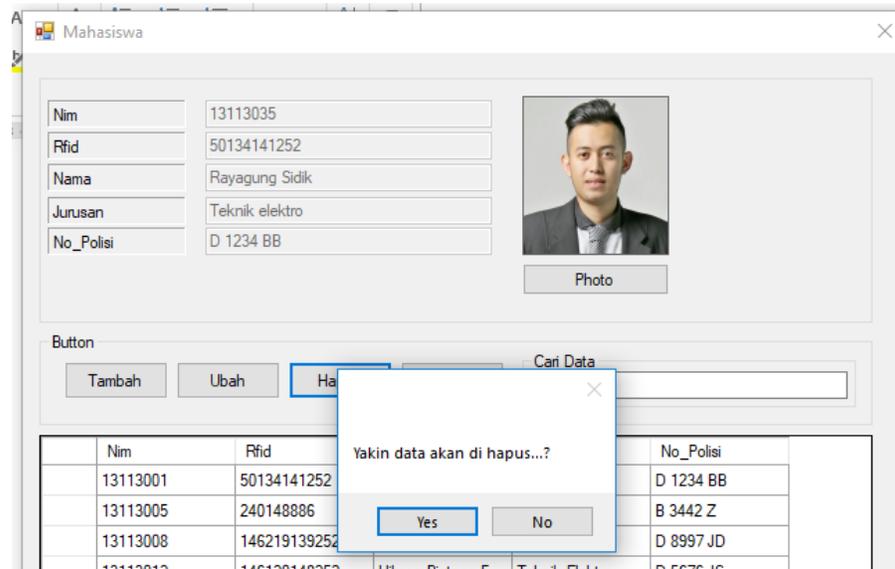
Untuk meng-*edit* data mahasiswa yang sudah terdaftar, maka data yang sudah terisi sebelumnya di *edit* menjadi menjadi identitas mahasiswa yang baru. Setelah data awal sudah di *edit*, selanjutnya klik simpan agar data yang baru dapat tersimpan.



Gambar 3.16 Data mahasiswa berhasil diubah

3. Delete data mahasiswa

Pada tampilan data mahasiswa terdapat tombol hapus yang berfungsi untuk menghapus data mahasiswa yang telah terdaftar. Setelah tombol hapus di klik, maka akan muncul tampilan seperti gambar dibawah ini.

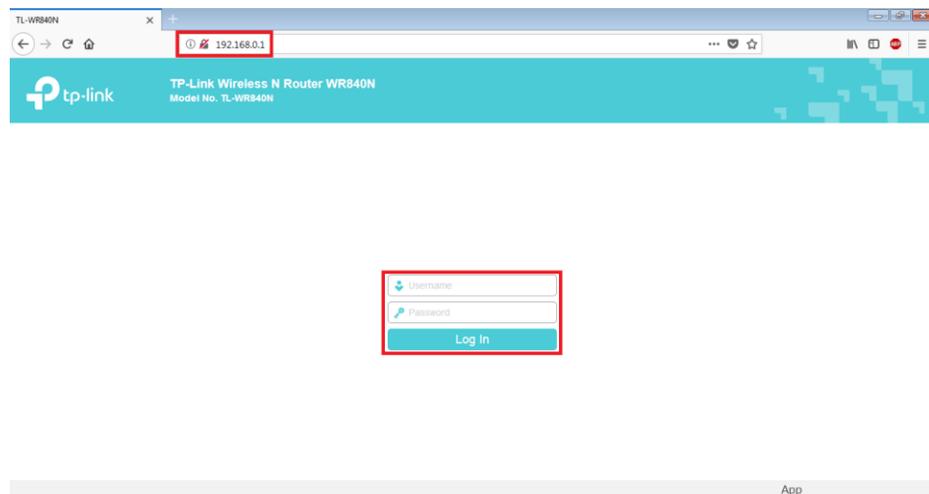


Gambar 3.17 Menghapus data mahasiswa

Setelah tombol hapus di klik, maka akan muncul pertanyaan “Yakin data akan di hapus?”. Klik *Yes* untuk menghapus data, apabila data tidak ingin di hapus maka klik tombol *No*.

3.5.3 Konfigurasi Router

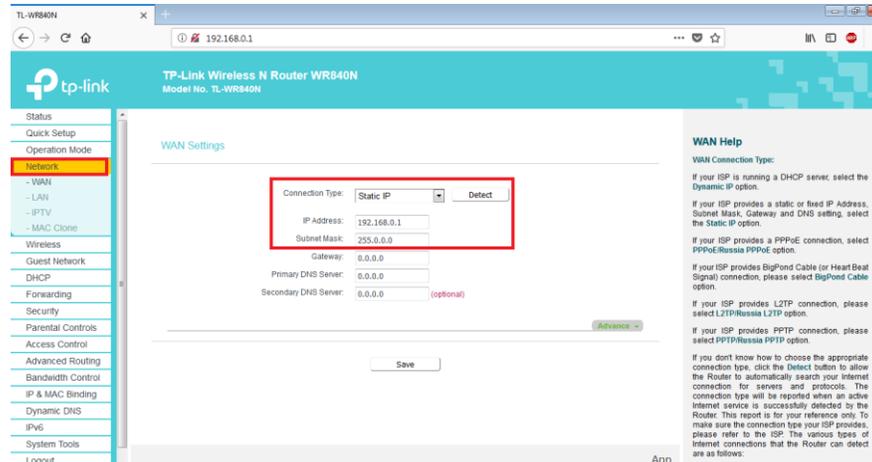
Router berfungsi untuk menghubungkan beberapa jaringan dan mengirimkan data melalui jaringan internet. Sebelum digunakan, router terlebih dahulu di konfigurasi agar data dapat dikirimkan. Untuk melakukan konfigurasi router, terlebih dahulu masuk ke *web browser* dan ketikkan alamat *IP Default* router tersebut seperti yang diperlihatkan pada gambar di bawah ini.



Gambar 3.18 Tampilan pada *web browser*

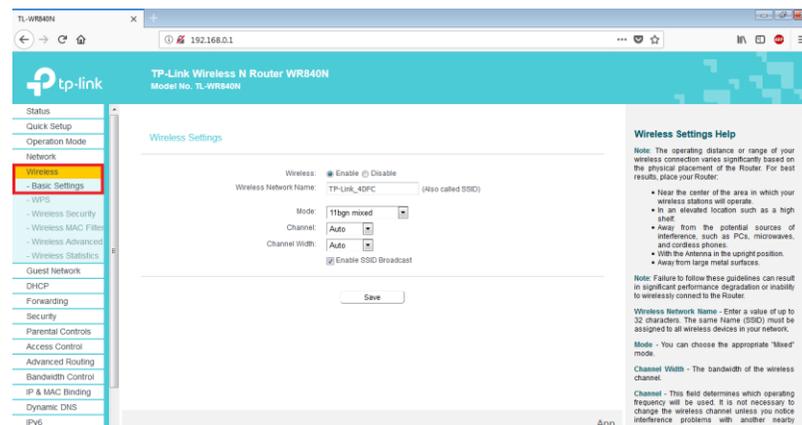
Pada gambar 3.18 diatas, dapat dilihat bahwa *IP Default router* yang digunakan adalah 192.168.0.1. Kemudian masukkan *username* dan *password* yang terdapat pada *router*, biasanya terletak pada bagian

belakang router. Setelah *username* dan *password* yang dimasukkan benar, maka akan muncul tampilan untuk mengkonfigurasi *router*.



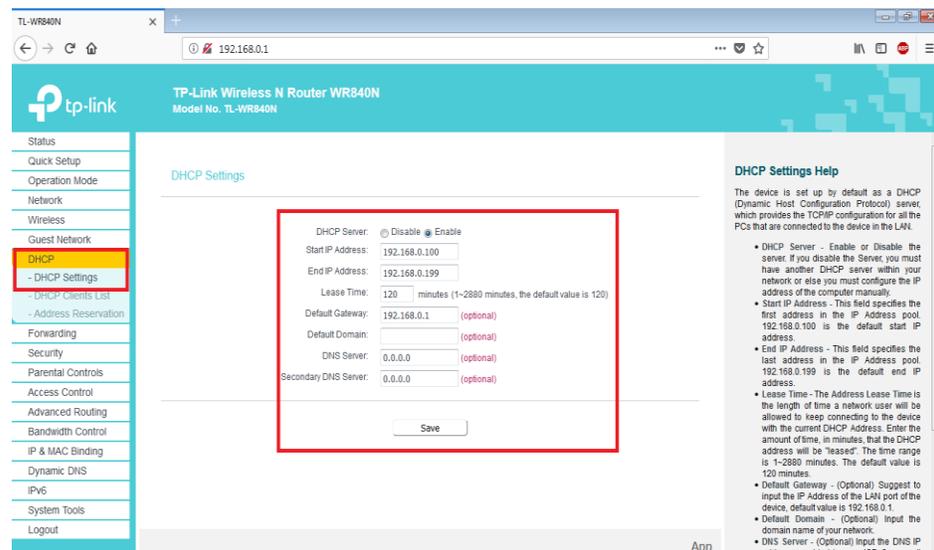
Gambar 3.19 Tampilan *Network*

Setelah tampilan konfigurasi *router* muncul, kemudian pilih *Network* untuk mengatur tipe jaringan yang diinginkan, *IP Address* dan *subnet mask* seperti yang diperlihatkan pada gambar 3.19 diatas. Setelah *Network* telah diatur, kemudian pilih *wireless* untuk mengatur *wireless* lalu pilih *basic setting* seperti yang diperlihatkan pada gambar di bawah ini.



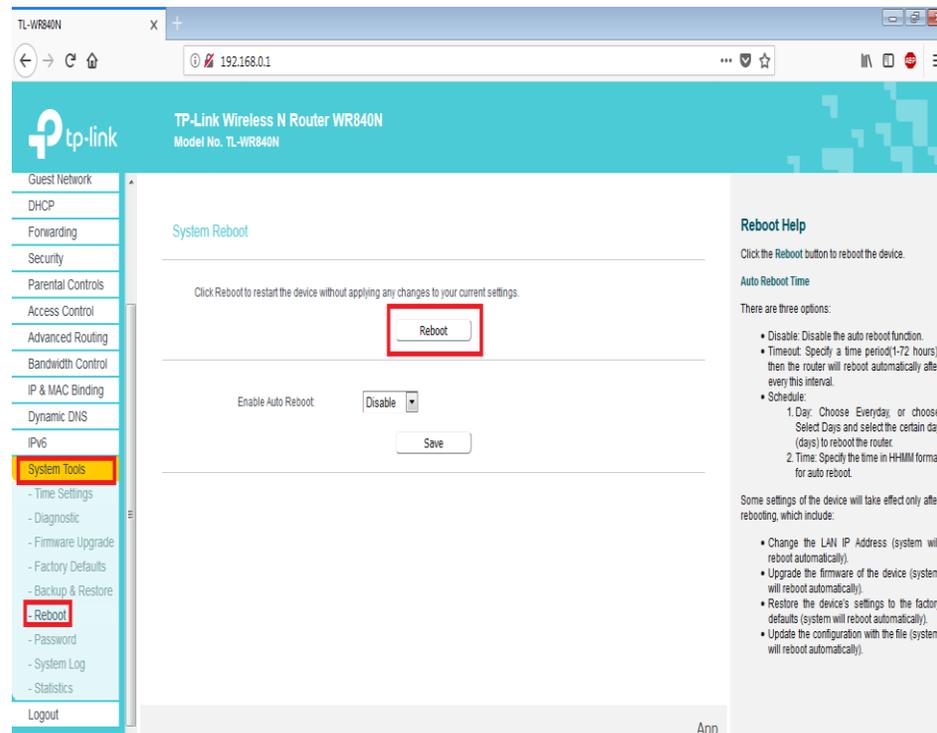
Gambar 3.20 Tampilan *wireless basic setting*

Pada gambar 3.20 diatas, dapat dilihat bahwa *basic setting* berfungsi untuk mengatur nama jaringan (SSID) *router* yang digunakan, kemudian mengatur *mode* dan *channel*. Setelah diatur, kemudian klik *save* untuk menyimpan. Langkah selanjutnya untuk konfigurasi *router* yaitu melakukan pengaturan pada DHCP agar perangkat yang dihubungkan ke *router* mendapat *IP Addrress* secara otomatis.



Gambar 3.21 Tampilan DHCP

Pada gambar 3.21 diatas, dapat dilihat bahwa alamat *IP Address* yang diatur berada pada 192.168.0.100 sampai 192.168.0.199, kemudian klik *save* untuk menyimpan. Setelah DHCP diatur, kemudian *reboot router* pada *system tools* seperti gambar dibawah ini.



Gambar 3.22 Tampilan *system tools*

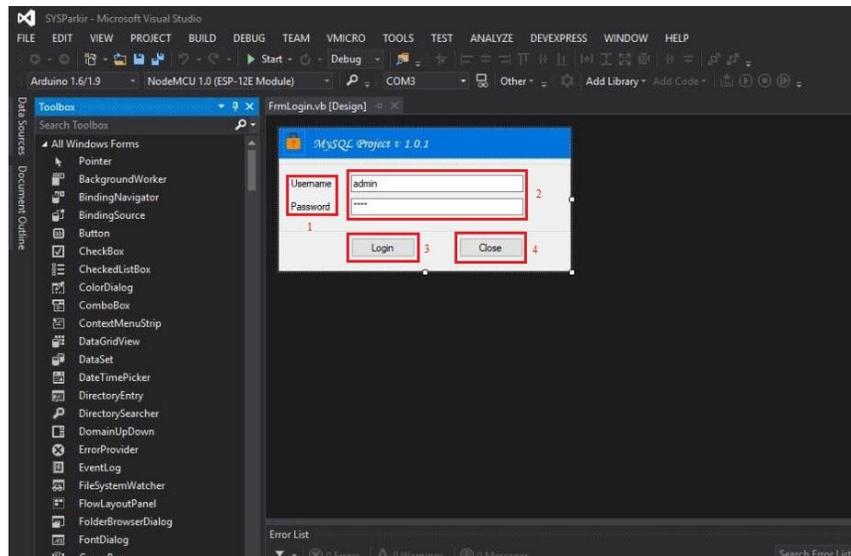
Setelah di *reboot*, maka *router* akan melakukan proses *restart* agar *router* dapat digunakan sesuai dengan konfigurasi yang telah di atur sebelumnya.

3.5.4 Perancangan *Software Interface*

Pada perancangan *software interface* parkir ini berfungsi untuk menampilkan data pengendara yang telah terdaftar pada *database* menggunakan *Visual Basic*. Berikut merupakan perancangan *software interface* yang dibuat.

A. Perancangan *form login*

Form login digunakan agar petugas parkir dapat masuk ke aplikasi parkir yang akan dibuat. Perancangan *form login* yang dibuat dapat dilihat pada gambar dibawah ini.

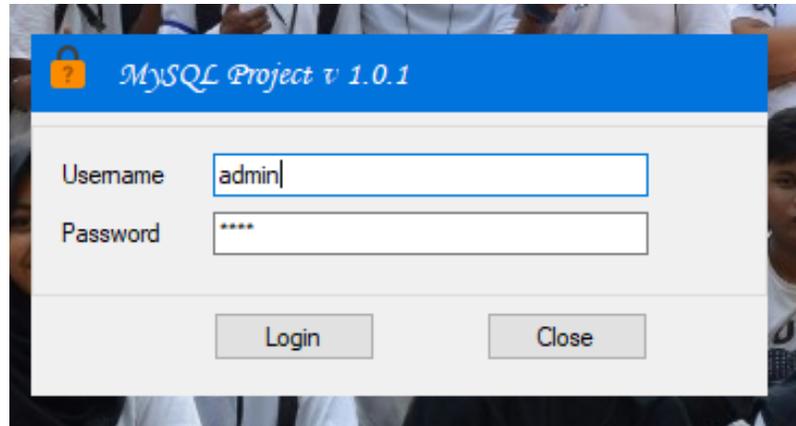


Gambar 3.23 Perancangan *form login*

Fungsi-fungsi kolom pada gambar 3.23 diatas yaitu sebagai berikut.

1. Kolom 1 merupakan 2 buah *label* untuk *username* dan *password*.
2. Kolom 2 merupakan 2 buah *textbox* untuk mengisi *username* dan *password* yang telah di rancang..
3. Kolom 3 merupakan *button* yang berfungsi untuk *login* setelah *username* dan *password* yang di isi benar.
4. Kolom 4 merupakan *button* yang berfungsi untuk *close* apabila tidak ingin menutup *form login*.

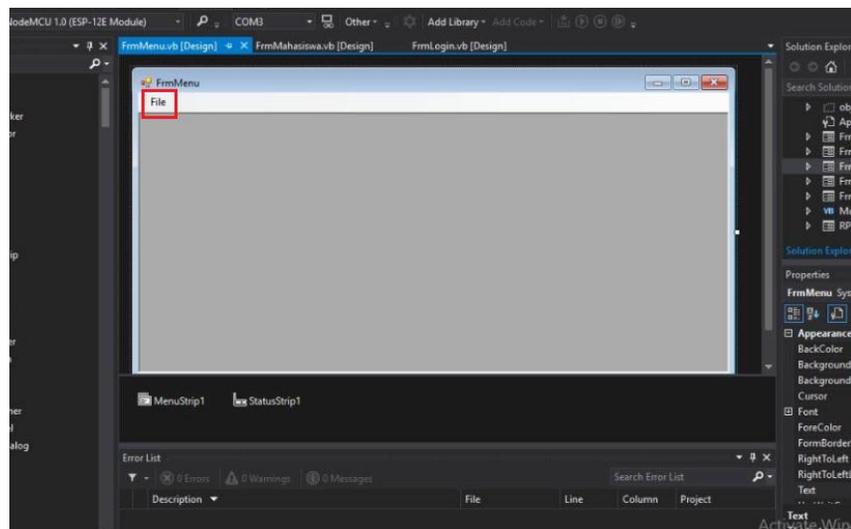
Setelah *form login* dirancang, kemudian di *run* maka akan terlihat seperti pada gambar 3.24 dibawah ini.



Gambar 3.24 Tampilan *form login*

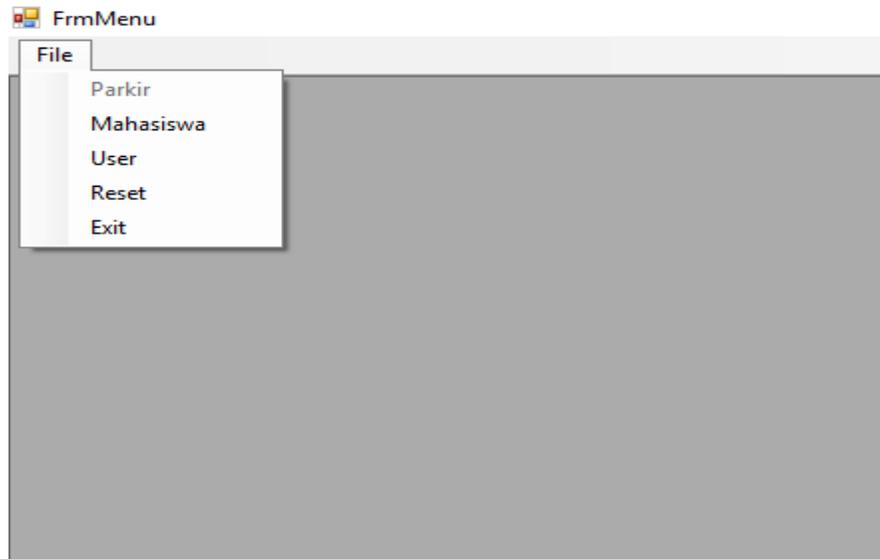
B. Perancangan *form* menu utama

Perancangan *form* menu utama berisi file yang memuat informasi parkir, mahasiswa, *user*, *reset* dan *exit*. Perancangan *form* menu utama dapat dilihat pada gambar di bawah ini.



Gambar 3.25 Perancangan *form* menu utama

Setelah *form* menu utama dirancang, kemudian di *run* maka akan terlihat seperti gambar dibawah ini.



Gambar 3.26 Tampilan *form* menu utama

Fungsi-fungsi file pada gambar diatas yaitu sebagai berikut.

1. *Form* Parkir berfungsi untuk membuka tampilan aplikasi parkir masuk atau parkir keluar.
2. *Form* Mahasiswa berfungsi untuk memasukkan data mahasiswa yang akan didaftarkan.
3. *Form* User berfungsi sebagai pengguna aplikasi parkir.
4. *Form* Reset berfungsi untuk mengatur ulang data mahasiswa.
5. *Form* Exit berfungsi untuk menutup aplikasi parkir.

C. Perancangan *form file* parkir

Perancangan *form file* parkir ini berisi data mahasiswa yang masuk dan keluar dari area parkir. Berikut ini merupakan gambar perancangan *form file* parkir yang dibuat.



Gambar 3.27 Perancangan *form file* parkir

Setelah *form file* parkir dirancang, kemudian di *run* maka akan terlihat seperti gambar di bawah ini.



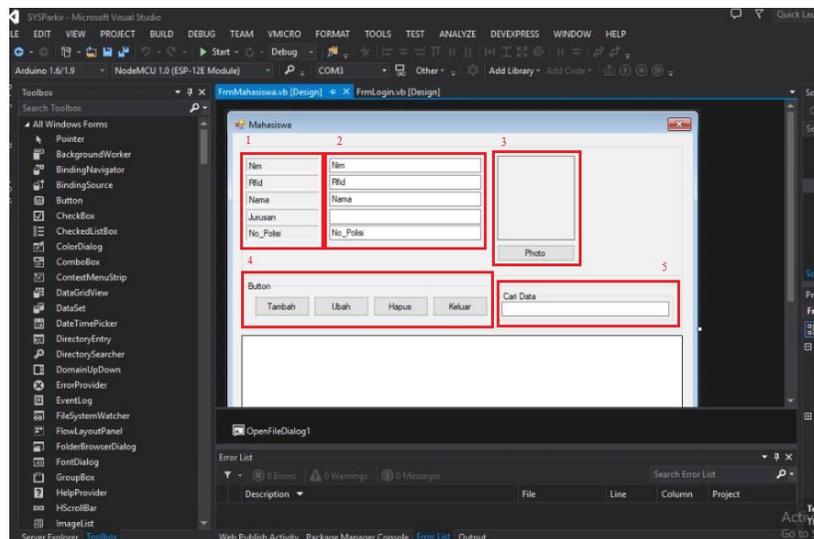
Gambar 3.28 Tampilan *form file* parkir

Fungsi-fungsi kolom pada gambar diatas yaitu sebagai berikut.

1. Kolom 1 berfungsi untuk menampilkan data mahasiswa berupa nim, nama, jurusan dan nomor polisi.
2. Kolom 2 berfungsi untuk menampilkan waktu.
3. Kolom 3 berfungsi untuk menampilkan tanggal.
4. Kolom 4 berfungsi untuk menampilkan jumlah kendaraan yang parkir.
5. Kolom 5 berfungsi untuk menampilkan foto mahasiswa.
6. Kolom 6 berfungsi untuk menampilkan foto mahasiswa saat masuk area parkir.

D. Perancangan *form file* mahasiswa

Pada perancangan *form file* mahasiswa ini berfungsi untuk memasukkan data baru ke *database*. Perancangan *form file* mahasiswa dapat dilihat pada gambar di bawah ini.

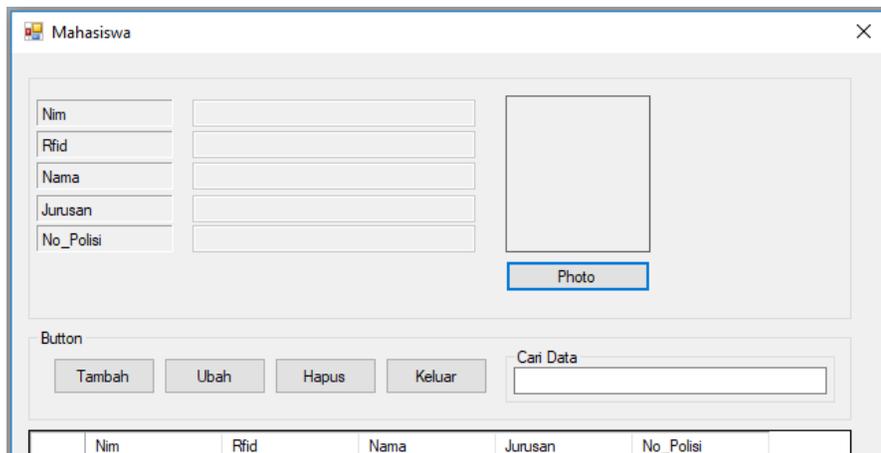


Gambar 3.29 Perancangan *form file* mahasiswa

Fungsi kolom pada gambar 3.29 diatas yaitu sebagai berikut.

1. Kolom 1 yaitu sebagai *label* berupa nim, rfid, nama, jurusan dan nomor polisi.
2. Kolom 2 yaitu sebagai *text* yang berfungsi untuk pengisian data berupa nim, rfid, nama, jurusan dan nomor polisi.
3. Kolom 3 berfungsi untuk memasukkan foto mahasiswa.
4. Kolom 4 yaitu sebagai *button* yang berfungsi untuk *input*, *edit* dan *delete* data mahasiswa.
5. Kolom 5 berfungsi untuk mencari data mahasiswa yang telah terdaftar.

Setelah *form file* mahasiswa selesai dirancang, kemudian di *run* maka akan terlihat seperti pada gambar dibawah ini.

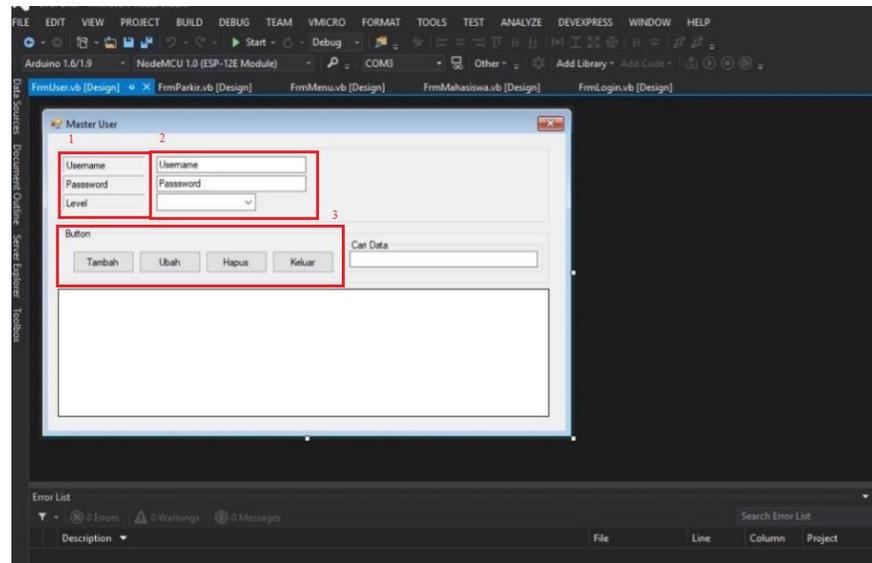


Nim	Rfid	Nama	Jurusan	No_Polisi
-----	------	------	---------	-----------

Gambar 3.30 Tampilan *form file* mahasiswa

E. Perancangan *form file user*

Pada perancangan *form file user* ini berfungsi sebagai pengguna aplikasi parkir. Perancangan *form file user* dapat dilihat pada gambar di bawah ini.

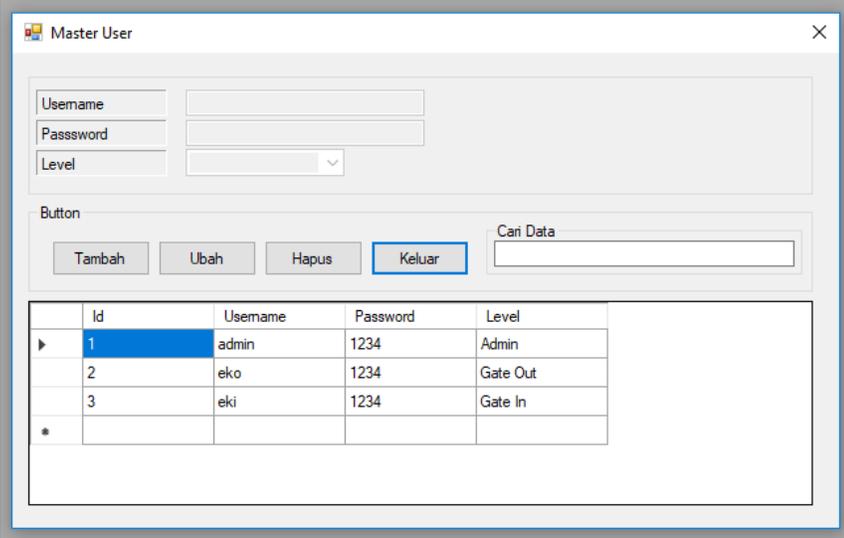


Gambar 3.31 Perancangan *form file user*

Fungsi-fungsi kolom pada gambar 3.31 diatas yaitu sebagai berikut.

1. Kolom 1 sebagai *label* berupa *username*, *password* dan *level*.
2. Kolom 2 sebagai *text* berfungsi untuk pengisian data berupa *username*, *password* dan *level*.
3. Kolom 3 sebagai *button* berfungsi untuk *input*, *edit* dan *delete* pengguna aplikasi parkir.
4. Kolom 4 berfungsi untuk mencari data pengguna aplikasi parkir.

Setelah *form file user* selesai dirancang, kemudia *run* maka akan terlihat seperti gambar dibawah ini.



	Id	Username	Password	Level
▶	1	admin	1234	Admin
	2	eko	1234	Gate Out
	3	eki	1234	Gate In
*				

Gambar 3.32 Tampilan *form file user*