



INSTITUT TEKNOLOGI DEL

E-KTP Scanner Using Raspberry

DOKUMEN TUGAS AKHIR

Oleh :

13317003 Anjelina Putri Napitu
13317015 Ruth Cindy F. Panjaitan
13317019 Talenta Mesianna Sitorus

FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO
PROGRAM STUDI DIII TEKNOLOGI KOMPUTER

LAGUBOTI

2020



INSTITUT TEKNOLOGI DEL

E-KTP Scanner Using Raspberry

DOKUMEN TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Diploma Teknik**

Oleh :

**13317003 Anjelina Putri Napitu
13317015 Ruth Cindy F. Panjaitan
13317019 Talenta Mesianna Sitorus**

**FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO
PROGRAM STUDI D III TEKNOLOGI KOMPUTER**

LAGUBOTI

2020

HALAMAN PERNYATAAN ORISINALISITAS

Tugas Akhir ini adalah hasil karya saya sendiri dan semua sumber yang dikutip maupun dirujuk telah saya nyatak dengan benar.

Nama : Anjelina Putri Napitu

NIM : 13317003

Tanda Tangan :



Nama : Ruth Cindy F. Panjaitan

NIM : 13317015

Tanda Tangan :



Nama : Talenta Mesianna Sitorus

NIM : 13317019

Tanda Tangan :



Tanggal : 03 Agustus 2020

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh :

1. Nama : Anjelina Putri Napitu

NIM : 13317003

2. Nama : Ruth Cindy F. Panjaitan

NIM : 13317015

3. Nama : Talenta Mesianna Sitorus

NIM : 13317019

Program Studi : D3 Teknologi Komputer

Judul Tugas Akhir : **E-KTP Scanner using Raspberry**

Telah berhasil dipertahankan di hadapan Dewan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Diploma Teknik, pada program studi Diploma 3 Teknologi Komputer, Fakultas Informatika dan Teknik Elektro, Institut Teknologi Del

DEWAN PENGUJI

Pembimbing 1 : Marojahan MT. Sigiro, ST., M.Sc ()

Pembimbing 2 : Istas Manalu, S.Si., M.Sc ()

Penguji 1 : Sari Muthia Silalahi, S.Pd., M.Ed ()

Penguji 2 : Ahmad Zatnika Purwalaksana, S.Si.,M.Si ()

Ditetapkan di : Laguboti

Tanggal : Agustus 2020

KATA PENGANTAR

Puji dan syukur penulis haturkan kepada Tuhan Yang Maha Esa karena rahmat yang diberikan kepada penulis, sehingga pengerjaan Tugas Akhir ini dapat diselesaikan dengan baik dan tepat waktu. Laporan Tugas akhir ini bertujuan untuk memberikan informasi bagi pembaca mengenai *E-KTP Scanner using Raspberry*.

Tugas Akhir ini merupakan salah satu matakuliah yang wajib ditempuh di program studi Teknologi Komputer di Institut Teknologi Del. Dengan selesainya laporan Tugas Akhir ini tidak terlepas dari berbagai pihak yang telah memberi masukan, ilmu, semangat, serta bimbingan kepada penulis. Sehingga dalam penulisan Tugas Akhir ini, penulis menyampaikan terimakasih kepada:

1. Bapak Marojahan Sigiro sebagai dosen pembimbing I serta Bapak Istas Pratomo Manalu sebagai pembimbing II
2. Bapak Istas Manalu sebagai koordinator Tugas Akhir 2019/2020
3. Orangtua, saudara, teman, dan seluruh pihak yang terlibat dalam pengerjaan Tugas Akhir ini.

Penulis juga menyadari bahwa laporan Tugas Akhir ini masih jauh dari sempurna baik dari materi maupun teknik penyajiannya. Sehingga penulis mengharapkan kritik, saran, dan masukan untuk menjadikan laporan Tugas Akhir ini jauh lebih baik. Semoga laporan Tugas Akhir ini dapat memberikan manfaat bagi penulis maupun pembaca.

Laguboti, 03 Agustus 2020

Anjelina Putri Napitu

Ruth Cindy F. Panjaitan

Talenta Mesianna Sitorus

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI DOKUMEN TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Institut Teknologi Del, penulis yang bertanda tangan di bawah ini :

1. Nama : Anjelina Putri Napitu
NIM : 13317003
2. Nama : Ruth Cindy F. Panjaitan
NIM : 13317015
3. Nama : Talenta Mesianna Sitorus
NIM : 13317019
Program Studi : Diploma III Teknologi Komputer
Fakultas : Informatika dan Teknik Elektro
Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Institut Teknologi Del **Hak Bebas Royalti Noneksklusif** (*Nonexclusive Royalty-Fee Right*) atas karya ilmiah penulis yang berjudul :

E-KTP Scanner Using Raspberry

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini

Institut Teknologi Del berhak menyimpan, mengalih/media-format dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir penulis selama tetap mencantumkan nama penulis sebagai penulis/pencipta dan sebagai pemilik Hak.

Demikian pernyataan ini penulis buat dengan sebenarnya.

Dibuat di : Laguboti

Tanggal : 03 Agustus 2020

Yang menyatakan,



Anjelina Putri Napitu



Ruth Cindy F. Panjaitan



Talenta Mesianna Sitorus

ABSTRAK

Program Studi : D3 Teknologi Komputer

Judul : E-KTP Scanner using Raspberry

KTP Elektronik adalah kartu tanda penduduk yang kini telah dilengkapi dengan sebuah teknologi, dimana pada e-KTP terdapat sebuah *chip*. *Chip* e-KTP tersebut adalah kartu pintar berbasis mikroposeor dengan kapasitas memori 8 KB. Cip tersebut menyimpan data pemilik e-KTP. *Chip* ini memiliki standar ISO 14443 A dan 14443 B, sehingga hanya dapat dibaca oleh alat pembaca tertentu. Alat yang mampu membaca cip tersebut ialah sebuah alat yang berbasis teknologi RFID (*Radio Frequency Identification*) dan mendukung tag ISO 14443 A dan 14433 B.

Tugas Akhir dengan judul E-KTP Scanner using raspberry Pi ini bertujuan untuk membangun sebuah sistem menggunakan *Optical Character Recognition* Tesseract untuk mengekstrak informasi dari e-KTP. Alat ini menggunakan RFID *reader* dan juga kamera digital untuk membaca informasi dari e-KTP seseorang.

E-KTP Scanner ini memiliki kelemahan misalnya saat camera telah menangkap gambar dari e-KTP maka proses ekstrak gambar menjadi teks oleh Tesseract OCR membutuhkan waktu dua detik untuk mendapatkan hasilnya. Namun, ketika *reader* membaca dan mengambil id dari e-KTP hanya memerlukan waktu 1 detik saja.

Sistem ini dilengkapi dengan sebuah *web* contoh yang dapat digunakan di pelabuhan dan dapat diakses oleh administrator.

Kata Kunci - E-KTP, RFID, *Camera*, OCR, Tesseract, Python.

ABSTRACT

Study Program : Diploma of Computer Technology

Title : E-KTP Scanner using Raspberry

Electronic KTP is a resident card that is now equipped with a technology, where in the e-KTP there is a chip. Cip e-KTP is a microprocessor-based smart card with 8 KB of memory. The CIP stores the data of the e-KTP owner. The chip have the ISO 14443 A and 14443 B standards, so it can only be read by certain readers. The tool that is able to read the chips is a device based on RFID (Radio Frequency Identification) technology and supports ISO 14443 A and 14433 B. tags.

This Final Project, titled E-KTP Scanner using Raspberry Pi, aims to build a system using Optical Character Recognition Test to extract information from e-KTP. This tool uses an RFID reader and also a digital camera to read information from someone's e-KTP.

E-KTP Scanner has a weakness, for example when the camera has captured images from e-KTP, the process of extracting images into text by Tesseract OCR takes two seconds to get the results. However, when the reader reads and retrieves the id from the e-KTP it only takes 1 second.

This system is equipped with a sample web that can be used at the port and can be accessed by administrators.

Keywords - E-KTP, RFID, Camera, OCR, Tesseract, Python

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALSITAS	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR.....	v
ABSTRAK	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Lingkup	3
1.3 Rumusan Masalah.....	3
1.4 Tujuan	3
1.5 Pendekatan	3
1.6 Sistematika Penyajian	4
1.7 Istilah, Definisi, dan Singkatan	5
BAB II TINJAUAN PUSTAKA	6
2.1 Landasan Teori.....	6
2.1.1 Internet of Things.....	6
2.1.2 Raspberry Pi.....	6
2.1.3 Optical Character Recognition (OCR).....	8
2.1.4 Tesseract OCR	10
2.1.5 RFID Reader	11
2.1.6 KTP Elektronik	12
2.1.7 JSON (<i>Javascript Object Notation</i>)	14
2.1.8 Database MySQL.....	15
2.2 Related Work	15
2.2.1 Sistem Akses Kontrol Kunci Elektrik Menggunakan Pembacaan e-KTP ...	15
2.2.2 Sistem Akses Pintu Menggunakan e-KTP Sebagai Kunci Elektronik Berbasis Near Field Communication Dimonitor Melalui Jaringan Komputer	16
BAB III ANALISIS DAN PERANCANGAN.....	17
3.1. Analisis	17
3.1.1 Analisis Masalah.....	17
3.1.2 Analisis Pemecahan Masalah.....	18
3.1.3 Analisis Kebutuhan Sistem	18
3.2 Desain	27
3.2.1 Desain Prototipe.....	27
3.2.3 Desain Contoh Implementasi Aplikasi Pencatatan Penumpang Kapal.....	29
3.2.4 Desain Arsitektur Sistem	32
3.2.5 Perancangan Desain Database	32
3.3 Business Process Modelling Notation (BPMN)	39
3.3.1 BPMN administrator melakukan registrasi penumpang	39
3.3.2 BPMN penumpang melakukan scan e-KTP	39
3.3.3 BPMN administrator melihat daftar penumpang	40
BAB IV IMPLEMENTASI DAN PENGUJIAN	41
4.1 Implementasi.....	41
4.1.1 Implementasi Raspberry dengan RFID RC522	41
4.1.2 Implementasi <i>Raspberry Pi</i> dengan Camera.....	43
4.1.3 Implementasi Aplikasi	44
4.2 Pengujian.....	45
4.2.1 Pengujian Terhadap Kamera.....	45
4.2.2 Pengujian Terhadap RFID	47

BAB V. KESIMPULAN DAN SARAN	49
5.1 Kesimpulan.....	49
5.2 Saran	49
Daftar Pustaka	xiii
BAB VI LAMPIRAN.....	xiv
6.1 Instalasi Sistem Operasi Raspbian	xiv
6.2 Install Tesseract	xv
6.3 Konfigurasi RFID pada Raspberry	xv
6.4 Konfigurasi <i>Camera</i> modul pada Raspberry:	xvi
6.5 Kode Program E-KTP Scanner using Raspberry	xvii

DAFTAR TABEL

Tabel 1. 1 Daftar Definisi	5
Tabel 1. 2 Daftar Singkatan	5
Tabel 3. 1 Daftar Spesifikasi <i>Hardware</i>	20
Tabel 3. 2 Tabel id	33
Tabel 3. 3 Tabel Ktp	33
Tabel 3. 4 Tabel Kapal	35
Tabel 3. 5 Daftar Penumpang	35
Tabel 3. 6 Dengke	36
Tabel 3. 7 Ferry	36
Tabel 3. 8 Ihan Batak	37
Tabel 3. 9 E-KTP	37
Tabel 4. 1 Pin RFID Reader	41
Tabel 4. 2 Pengujian RFID	47

DAFTAR GAMBAR

Gambar 2. 1 Sisi Depan Raspberry Pi 3 Model B+	7
Gambar 2. 2 Sisi Belakang Raspberry Pi 3 Model B+	7
Gambar 2. 3 Skema OCR.....	8
Gambar 2. 4 Proses Tesseract OCR	10
Gambar 2. 5 RFID Reader	11
Gambar 2. 6 E-KTP Layer	13
Gambar 3. 1 Kamera	19
Gambar 3. 2 RFID Reader	19
Gambar 3. 3 Skema Alur Kerja Sistem.....	24
Gambar 3. 4 Flowchart OCR	26
Gambar 3. 5 Flowchart RFID Reader	26
Gambar 3. 6 Prototipe e-KTP Scanner	27
Gambar 3. 7 Use Case.....	28
Gambar 3. 8 Desain menu registrasi	29
Gambar 3. 9 Dashboard Aplikasi.....	30
Gambar 3. 10 Tampilan Menu Kapal Ihan Batak	30
Gambar 3. 11 Tampilan Sebelum Melakukan Scan.....	31
Gambar 3. 12 Informasi Hasil scan e-KTP	31
Gambar 3. 13 Desain Arsitektur Sistem	32
Gambar 3. 14 BPMN Registrasi Penumpang	39
Gambar 3. 15 BPMN Scan e-KTP	39
Gambar 3. 16 BPMN Admin Melihat Daftar Penumpang.....	40
Gambar 4. 1 Enable SPI.....	42
Gambar 4. 2 Proses Membaca id	42
Gambar 4. 3 Tabel id	43
Gambar 4. 4 <i>Enable Camera</i>	43
Gambar 4. 5 <i>Camera</i> pada Raspberry	44
Gambar 4. 6 Tampilan Halaman e-KTP	45
Gambar 4. 7 Tampilan Halaman Daftar Kapal	45
Gambar 4. 8 Proses Pengambilan Gambar	46
Gambar 4. 9 Hasil Konversi Gambar.....	47
Gambar 4. 10 Data E-KTP.....	47
Gambar 4. 11 ID Berhasil Dibaca.....	47

BAB I PENDAHULUAN

1.1 Latar Belakang

KTP adalah kartu yang digunakan mengetahui identitas penduduk. Kartu ini sangat perlu, karena akan digunakan untuk berbagai macam kebutuhan. Misalnya mengisi dokumen di bank, di rumah sakit, di hotel dan juga untuk mengurus passport, SIM dan sebagainya. Sekarang KTP sudah berkembang menjadi e-KTP. Dengan e-KTP kita hanya perlu membuat KTP sekali saja seumur hidup. Kartu Tanda Penduduk Elektronik (*e-KTP*) di Indonesia dibekali dengan teknologi *chip* RFID. Informasi pribadi mengenai pemiliknya, termasuk data kependudukan dan biometrik, tersimpan di dalam *chip* tersebut. RFID atau *Radio Frequency Identification* adalah teknologi pengiriman informasi lewat gelombang radio.

Fungsi e-KTP:

- Digunakan sebagai identitas atau pengenalan jati diri.
- Berlaku secara nasional.
- Mencegah KTP ganda maupun pemalsuan KTP.
- Dapat digunakan oleh masyarakat sebagai kartu suara dalam pelaksanaan PEMILU.

Sebagai masyarakat Indonesia yang hidup di era 4.0, masyarakat telah menggunakan teknologi dalam kehidupan sehari – hari. Misalnya adalah IoT untuk memudahkan seseorang mengontrol perangkat miliknya melalui internet [3]. Contohnya melakukan *scanning barcode* untuk mengabsen mahasiswa dan lainnya. Selain *barcode*, e-KTP juga dapat di *scan* sehingga dapat menampilkan data diri dari pemilik e-KTP, hal ini sangat berguna dalam kehidupan sehari-hari. Dengan adanya e-KTP *scanner* ini kita tidak perlu mengisi formulir data diri secara manual lagi.

Salah satu contoh yaitu ketika kita menjadi penumpang sebuah kapal, dimana ketika ingin memasuki kapal tersebut maka kita harus menunjukkan e-KTP kepada petugas yang ada di kapal. Petugas kapal pastinya tidak dapat memastikan bahwa e-KTP yang kita bawa adalah e-KTP asli hanya dengan kasat mata. Selain menunjukkan e-KTP tersebut, kita juga akan mengisi data kita sesuai dengan e-KTP secara manual. Hal ini akan menjadi rumit ketika ada 1000 penumpang yang ingin memasuki kapal, maka akan menyebabkan antrian yang begitu panjang. Untuk memastikan kedua hal tersebut, maka dibutuhkan sebuah perangkat yang mampu mengenali e-KTP yang kita bawa adalah asli atau tidak dan perangkat tersebut merupakan perangkat yang mampu mempermudah kita memperoleh informasi yang akurat mengenai penumpang.

Contoh lain yang tidak beda jauh ialah kasus penyeberangan kapal di Danau Toba. Kapal penyeberangan ialah transportasi utama di kawasan pariwisata Danau Toba dan menjadi pilihan favorit masyarakat di sekitar Danau Toba maupun wisatawan yang ingin berkunjung ke sekitaran Danau Toba. Data manifest penumpang menggunakan kapal penyeberangan di Danau Toba saat ini hanya mencatat jumlah penumpang dan jumlah kendaraan yang menaiki kapal, namun tidak mencatat nama penumpang. Laporan yang disajikan hanya berupa rekapitulasi daftar perjalanan kapal dan daftar muatan kapal secara umum, sehingga tingkat keakuratan data yang dihasilkan akan sangat diragukan kebenarannya. Sistem pencatatan yang tidak terlaksana dengan baik tersebut menyebabkan ketidakjelasan jumlah, nama, alamat, dan sebagainya terkait penumpang sebuah kapal. Berdasarkan kasus tersebut, maka dibutuhkan sebuah sistem yang mencatat data berdasarkan e-KTP untuk membantu pihak terkait dalam melakukan pendataan [10].

KTP *scanner* ini nantinya akan membaca dan mengekstrak informasi yang ada di e-KTP menjadi informasi digital dan dapat menampilkan informasi yang ada pada KTP tersebut menggunakan API yang akan kami bangun, informasi yang ditampilkan seperti foto yang ada di dalam KTP, nama, alamat, NIK dan lainnya dalam bentuk teks. Data yang di peroleh ini akan ditampilkan di API (*Application Programming Interface*), sehingga *user* dapat melihat data dari hasil *scan* itu. Untuk membuat *web API*, hal yang harus disediakan ialah *response* dengan format JSON, XML atau format apapun yang kita inginkan. Namun untuk menyelesaikan sistem ini kami menggunakan format JSON.

Dengan begitu e-KTP *scanner* ini dapat dipakai dalam kehidupan sehari-hari untuk menghindari peng-*input*-an informasi pribadi secara manual. Contohnya, digunakan ketika penyelenggaraan pemilihan umum. Pemilih datang dengan membawa e-KTP, kemudian e-KTP akan di scan terhadap RFID *reader* yang berada di meja panitia, yang kemudian akan mengkonfirmasi identitas pemilih tersebut, apakah dia termasuk dalam DPT (Daftar Pemilih Tetap) di lokasi pemungutan suara yang ditujukan atau tidak, sehingga tidak perlu lagi menggunakan kertas fotokopian berisikan data diri pemilih yang bahkan masih kita isi dengan tandatangan kita, karena semua data kita telah tercatat secara elektronik.

1.2 Lingkup

Implementasi e-KTP *scanner* ini memiliki batasan seperti berikut:

1. Produk ini akan membaca dan mengekstrak informasi dari *chip* yang berada dalam e-KTP menggunakan RFID *reader* ke informasi digital dan menampilkan informasi tersebut menggunakan *framework* PHP yaitu Yii.
2. Produk ini akan membaca dan mengekstrak gambar dari e-KTP yang diambil oleh *camera* ke informasi digital menggunakan teknologi OCR dengan *tools* Tesseract

1.3 Rumusan Masalah

Bagaimana menggunakan RFID *reader* dan camera digital untuk membaca dan mengekstrak informasi dari e-KTP menjadi informasi digital?

1.4 Tujuan

Dengan membangun e-KTP *scanner* ini, maka akan mempermudah pengambilan informasi dari e-KTP dan *chip* e-KTP. Perangkat ini bisa di hubungkan terhadap aplikasi lain yang membutuhkan inputan data berupa informasi dari e-KTP.

1.5 Pendekatan

1. *Research Problem*

Membuat beberapa pertanyaan-pertanyaan terkait judul tugas akhir yang bisa dijadikan sebagai alasan untuk membangun sistem ini.

2. *Study Literature*

Pada tahap ini dilakukan pengumpulan beberapa referensi untuk mendapatkan informasi tentang topik tugas akhir yang akan dilakukan. *Study literature* dilakukan dengan cara mencari, mempelajari, dan memahami beberapa jurnal mengenai konsep dan cara kerja OCR (*Optical Character Recognition*), cara penggunaan Raspberry serta proses pembacaan data oleh RFID *reader*.

3. Perancangan Sistem

Tahap perancangan sistem dilakukan untuk merancang apa saja yang diperlukan untuk membangun sebuah sistem yang mampu melakukan pembacaan data pada e-KTP menggunakan kamera dan RFID *reader*.

4. Pengumpulan Data

Pada tahap pengumpulan data dilakukan untuk memperoleh objek yang akan digunakan untuk membangun sistem. Pada pengumpulan data, *developer* harus mengetahui *software* yang digunakan untuk membaca tulisan yang ada di e-KTP.

5. Pembuatan Sistem

Pada tahap ini merupakan tahap implementasi untuk membangun suatu sistem yang sebelumnya telah dirancang pada tahap perancangan sistem didukung oleh informasi yang sudah diperoleh pada tahap pengumpulan informasi.

6. Testing

Pada tahap ini *developer* melakukan uji coba terhadap sistem yang telah jadi dan mengadakan evaluasi apakah seluruh fungsi berjalan dengan baik.

1.6 Sistematika Penyajian

1. Bab 1 Pendahuluan

Pada bab ini berisi penjelasan diuraikan latar belakang, ruang lingkup, tujuan, pendekatan yang dilakukan oleh *developer* dalam penyelesaian tugas akhir. Selain itu juga terdapat beberapa definisi, akronim, singkatan yang digunakan dalam dokumen tugas akhir ini.

2. Bab 2 Tinjauan Pustaka

Pada bab ini dideskripsikan tinjauan pustaka yang mencakup sumber informasi, rangkuman metode, dan penelitian yang memiliki hubungan dengan sistem yang akan dibangun oleh *developer* untuk menyelesaikan tugas akhir ini.

3. Bab 3 Analisis dan Perencanaan Sistem

Pada bab ini menjelaskan analisis yang dilakukan terhadap pengerjaan tugas akhir, tahapan perancangan sistem, desain perangkat, dan desain *user interface*.

1.7 Istilah, Definisi, dan Singkatan

Pada bagian ini terdapat istilah, definisi, dan singkatan yang digunakan untuk menulis dokumen tugas akhir. Daftar istilah, definisi dan singkatan ini dapat mempermudah pembaca untuk memahami isi dokumen.

Tabel 1. 1 Daftar Definisi

No.	Istilah	Definisi
1.	IoT	Konsep perangkat yang memiliki kemampuan untuk mentransfer data tanpa perlu terhubung dengan manusia, melainkan internet sebagai medianya.
2.	<i>Flowchart</i>	Bagan yang dilengkapi dengan bentuk atau simbol tertentu yang merepresentasikan urutan proses atau instruksi tertentu dengan mendetail.
3.	<i>Use case</i>	Merupakan gambaran dengan simbol tertentu yang memperlihatkan interaksi yang terjadi antara aktor dan inisiator pada sistem yang ada.

Tabel 1. 2 Daftar Singkatan

No.	Singkatan	Deskripsi
1.	API	<i>Application Programming Interface</i>
2.	SPI	<i>Serial Peripheral Interface</i>
3.	OCR	<i>Optical Character Recognition</i>
4.	RFID	<i>Radio Frequency Identification</i>
5.	KTP	Kartu Tanda Penduduk
6.	TCP	<i>Transmission Control Protocol</i>
7.	JSON	<i>JavaScript Object Notation</i>
8.	RGB	<i>Red, Green, Blue</i>
9.	LED	<i>Light Emitting Diode</i>
10.	NIK	Nomor Induk Kependudukan

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan landasan teori yang digunakan sebagai dasar teori dalam pengerjaan tugas akhir.

2.1 Landasan Teori

Pada landasan teori akan dijelaskan mengenai dasar teori yang berhubungan dengan tugas akhir seperti pengertian dan konsep dasar mengenai sistem.

2.1.1 Internet of Things

Internet of Things ialah konsep otomatisasi dimana suatu objek yang memiliki kemampuan untuk men-*transfer* data melalui jaringan tanpa memerlukan interaksi manusia dengan manusia ataupun manusia dengan komputer. Konsep IoT diterapkan dengan mengandalkan beberapa teknologi yang secara dominan digabungkan menjadi satu kesatuan diantaranya sensor, *connectivity*, aktuator dan lain sebagainya. Sensor merupakan teknologi atau komponen yang dapat berfungsi untuk mengkonversi suatu besaran tertentu menjadi satuan analog yang kemudian dapat dibaca oleh suatu rangkaian elektronik, contohnya sensor gerak, suhu, udara, panas, dan jarak. Konektivitas atau jaringan berfungsi sebagai penghubung dan media pertukaran informasi yang terjadi pada perangkat IoT. Aktuator merupakan alat yang digunakan untuk mengubah listrik menjadi sebuah gerakan mekanis. Aktuator ini juga merupakan proses lanjutan dari *output* sebuah proses olah data yang dihasilkan oleh sensor atau *controller*.

2.1.2 Raspberry Pi

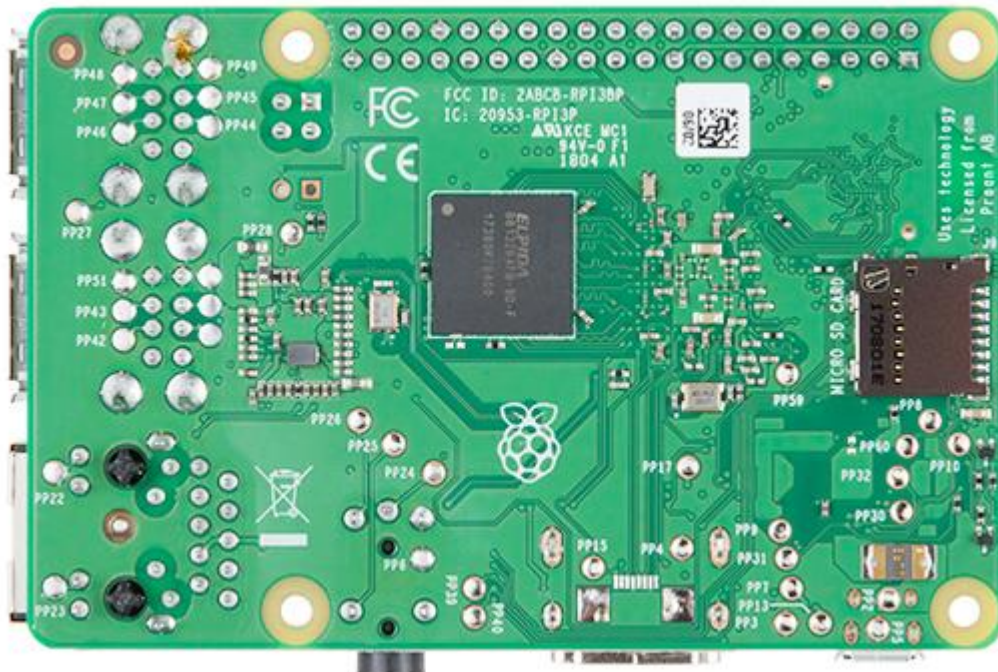
Raspberry Pi ialah komputer papan tunggal (*single board circuit*) yang memiliki mbentuk fisik berukuran mini. Raspberry Pi mempunyai dua model, yaitu model A dan model B. Raspberry Pi model B memiliki kapasitas penyimpanan RAM 512 MB dan telah dilengkapi dengan teknologi portal *ethernet* yang tidak dimiliki oleh model A. Sedangkan Raspberry Pi model A memiliki kapasitas penyimpanan 256 MB. Dalam mengerjakan tugas akhir ini, kami menggunakan Raspberry Pi Model B+. Adapun yang membuat Raspberri Pi 3 Model B+ lebih unggul daripada tipe sebelumnya ialah karena beberapa kelebihan dan peningkatan yang dimiliki seperti berikut :

1. Raspberry Pi Model B+ ini menggunakan *chipset* yang baru yakni *Broadcom BCM2873B0 Cortex A53 64-bit* dilengkapi denga kecepatan 1,4GHz. *Chipset* tersebut memiliki kualitas manajemen suhu yang lebih baik sehingga *chipset* bisa bekerja dengan kecepatan penuh dalam jangka waktu yang lebih lama sebelum mengalami sesi *throttling* akibat panas.

2. Raspberry Pi Model B ini juga memiliki tingkat kemampuan jaringan yang lebih memadai dengan koneksi *wireless dual band* yang telah mendukung versi 802.11ac dan *Bluetooth* versi 4.2.



Gambar 2. 1 Sisi Depan Raspberry Pi 3 Model B+



Gambar 2. 2 Sisi Belakang Raspberry Pi 3 Model B+

2.1.3 Optical Character Recognition (OCR)

Teknologi OCR ialah sebuah teknologi dengan pendekatan yang menyediakan pengenalan *alphanumeric character* baik yang berupa tulisan tangan, *image*, maupun *computer text* hanya dengan memindai (*scanning*) citra pada *image*, *computer text*, dan atau tulisan tangan tersebut secara digital dan mengubahnya menjadi bentuk yang dapat di-*scan* melalui pemindai (*scanner*), kemudian teknologi OCR akan menafsirkan citra tersebut dan mengkonversinya menjadi bentuk data ASCII.



Gambar 2. 3 Skema OCR

Secara umum proses yang dilakukan oleh OCR ialah seperti Gambar 3 dengan keterangan seperti berikut :

a. *File Input*

File input merupakan *file* citra digital yang memiliki ekstensi *.bmp, *.png, ataupun *.jpg.

b. *Preprocessing*

Tahap *preprocessing* merupakan proses yang dilakukan untuk menghilangkan bagian – bagian yang sebenarnya tidak dibutuhkan pada *gambar input* atau *file input* untuk memasuki proses selanjutnya. Pada tahap preprocessing ini ada beberapa sub tahap lagi yang dilakukan, yakni :

1. Grayscale

Grayscale merupakan proses konversi *file* citra dari citra yang berwarna *Red-Green-Blue (RGB)* menjadi citra *grayscale* atau citra yang memiliki nilai warna abu-abu.

2. Binerisasi

Pada tahap ini, *file* citra digital akan dikonversi menjadi citra biner. Citra biner (*binary image*) merupakan citra yang pikselnya hanya memiliki dua jenis nilai *grayscale* atau derajat keabuan, diantaranya adalah hitam dan putih. Piksel-piksel objek pada citra biner ini bernilai 1 dan pixel-pixel-pixel yang menjafi latar belakangnya bernilai 0. Ketika gambar ditampilkan, nilai 0 adalah warna putih dan nilai 1 adalah warna hitam. Sehingga dapat disimpulkan bahwa pada citra biner yang menjadi latar belakang ialah wilayah *object* yang warna putih, sebaliknya objek (tulisan) akan memiliki warna hitam.

c. Segmentasi

Segmentasi adalah tahapan dimana terjadinya pemisahan area pengamatan (*region*) di setiap karakter yang diidentifikasi atau dideteksi oleh OCR. Proses segmentasi file citra biner oleh OCR ini bertujuan untuk mengelompokkan piksel-piksel *object* menjadi area yang menunjukkan atau merepresentasikan sebuah *object*. Batas antara *object* dengan latar belakang terlihat jelas pada citra biner [4]. Piksel objek berwarna hitam, sebaliknya *pixel* latar belakang berwarna putih. Pertemuan antara piksel hitam dengan piksel putih dapat dimodelkan sebagai segmen garis.

d. Normalisasi

Normalisasi adalah proses konversi dimensi *region* setiap karakter yang dilakukan dengan tujuan memperoleh *input* citra yang lebih baik dan juga penyamaan ketebalan setiap karakter. Proses ini dilakukan dengan tujuan untuk menyesuaikan data citra inputan dengan data citra yang ada pada *database* [7]. Proses normalisasi akan disesuaikan dengan kebutuhan pada proses pengenalan citra yang digunakan. Salah satu proses normalisasi yang paling umum dalam pengenalan pola adalah normalisasi ukuran citra. Sebagai contoh, pada tahapan pengenalan pola ini menggunakan algoritma *template matching* dimana ukuran citra *input* harus disesuaikan dengan ukuran citra yang ada di basis data.

e. Ekstraksi ciri

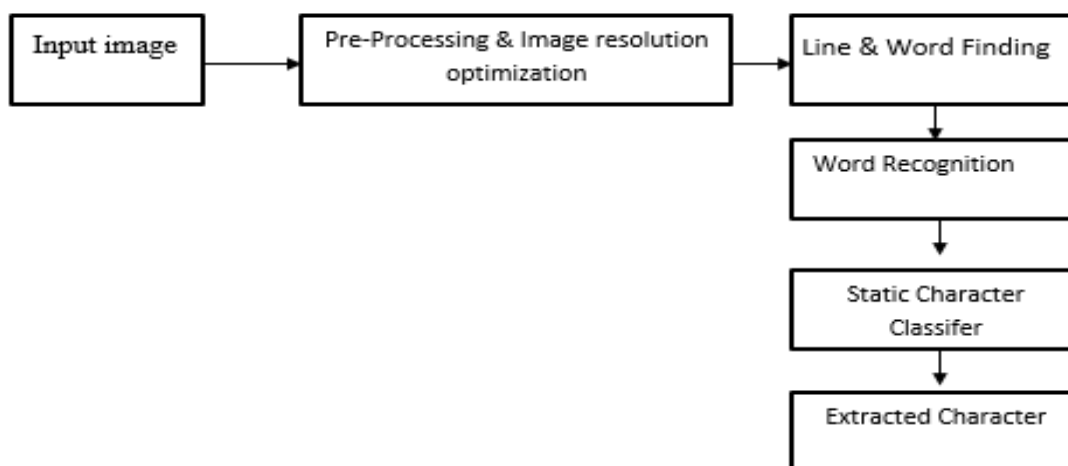
Ekstraksi ciri merupakan tahapan yang dilakukan untuk mengekstrak dan mengambil informasi berupa ciri tertentu dari karakter atau objek yang diamati. Melalui proses ekstraksi ciri ini, maka OCR akan memperoleh karakteristik suatu karakter yang membedakannya dari objek atau karakter lainnya.

f. *Recognition*

Recognition merupakan proses untuk mengenali karakter yang diamati dengan cara membandingkan ciri – ciri karakter yang ada di dalam *database*.

2.1.4 Tesseract OCR

Tesseract adalah salah satu teknologi OCR *engine open source* yang memiliki tingkat keakuratan yang baik dalam hal pendeteksian. Tesseract ini dapat diterapkan berbagai sistem operasi seperti Linux, Windows dan Mac OS. Selain itu, Tesseract juga dapat diterapkan pada perangkat-perangkat yang lain, termasuk Android dan iPhone.



Gambar 2. 4 Proses Tesseract OCR

Step pertama yang dilakukan oleh Tesseract OCR ialah tahap *pre-processing* kemudian yang selanjutnya adalah tahap Tesseract OCR. Tahapan yang menjadi titik fokus Tesseract adalah tahap *pre-processing* yang merupakan tahap dimana sebelum melalui proses pengenalan citra. *Timing processing* yang digunakan juga harus ditinjau pada saat tahap *pre-processing* agar tidak berpengaruh pada penambahan *running-time*. Proses pengenalan karakter atau citra dilakukan dengan menggunakan data latih seperti kamus data *character* yang tersedia pada Tesseract. Setelah proses *pre-processing* tersebut, tahap yang dilakukan oleh Tesseract selanjutnya ialah tahap *line and word finding*. Pada tahap ini Tesseract OCR

akan melakukan pencarian *object* seperti teks dan juga memperkirakan kemiringan teks. Setelah kemiringan teks berhasil ditemukan, selanjutnya Tesseract OCR akan menciptakan beberapa garis dari berbagai sudut seperti garis batas atas, garis tengah, dan garis bawah dari *object* yang diidentifikasi. Kemudian akan dilakukan pendeteksian jarak maupun spasi huruf dan dilakukan proses *cutting* dengan tujuan untuk membuat masing-masing huruf menjadi terpisah. Pada tahap *word recognition*, bagian dari pengenalan karakter (*character recognition*) akan mengidentifikasi bagaimana kata-kata bisa disegmentasi ke karakter. Selanjutnya, pada *static character classifier*, Tesseract OCR selanjutnya melakukan pengidentifikasian huruf baik dari segi warna, *font*, ukuran, dan atribut. Sehingga, begitu teks yang diidentifikasi oleh Tesseract OCR memiliki *font* berbeda-beda, mampu dibaca dan diekstrak oleh Tesseract OCR.

2.1.5 RFID Reader

RFID (*Radio Frequency identification*) adalah sebuah teknologi yang menggunakan metode auto-ID atau *Automatic Identification*. Auto-ID adalah metode pengambilan data melalui identifikasi *object* yang dilakukan secara otomatis tanpa adanya keterlibatan manusia. RFID ialah teknologi penangkapan data yang dapat digunakan secara elektronik untuk mengidentifikasi, melacak, memindai, serta menyimpan informasi yang sebelumnya tersimpan di dalam sebuah *tag* dengan menggunakan gelombang radio [5].



Gambar 2. 5 RFID Reader

Proses identifikasi yang terjadi pada RFID *tag* dilakukan oleh RFID *reader* dan RFID *tag*. RFID *tag* terdiri dari sebuah antena dan *chip* rangkaian sirkuit yang saling terintegrasi. Pada hakikatnya, rangkaian elektronik pada RFID memiliki kapasitas memori yang memungkinkan RFID *tag* mampu menyimpan data. Seluruh RFID *tag* mampu mendapatkan ID *number* dari sebuah *tag* atau kartu ketika *tag* tersebut didekatkan dengan RFID *reader*. Terdapat dua jenis RFID *tag*, yaitu *tag* aktif yang memiliki sebuah catu daya. *Tag* aktif memiliki kelebihan dimana *tag* ini dapat di *read* atau *write* sementara *tag* pasif

merupakan *tag* yang tidak memiliki catu daya dan hanya bisa di *read*. Sumber tenaga yang digunakan untuk mengaktifkan RFID *tag* pasif diperoleh dari sinyal yang dipancarkan oleh RFID *reader*.

RFID *reader* merupakan alat penghubung antara *software* aplikasi dengan antena yang akan memancarkan radiasi gelombang radio terhadap RFID *tag*. Gelombang radio yang ditransmisikan oleh antena tersebut berpropagasi pada ruangan di sekitarnya sehingga data yang berada pada *tag* RFID dapat berpindah secara *wireless* dari ke RFID *reader* [5]. Data pada RFID *tag* tersebut merupakan susunan nomor unik seperti ID, yang terdapat pada suatu produk yang memiliki *tag*.

RFID *reader* memiliki antena yang berfungsi untuk memancarkan radiasi gelombang radio terhadap *tag* yang berada di sekitarnya. Gelombang radio yang ditransmisikan oleh antena RFID *reader* tersebut secara cepat akan berpropagasi pada sekitarnya dan hal inilah yang membuat data dari *reader* mampu berpindah secara *wireless* ke *tag* RFID yang memiliki posisi berdekatan dengan antena [1].

2.1.6 KTP Elektronik

KTP Elektronik atau dikenal dengan *e-KTP* adalah dokumen kependudukan yang memuat sistem keamanan atau pengendalian baik dari sisi administrasi ataupun teknologi informasi dengan berbasis pada *database* kependudukan nasional. Realisasi *e-KTP* di Indonesia memiliki tujuan untuk mewujudkan sistem kepemilikan satu identitas pengenal (KTP) untuk satu penduduk dalam periode hingga seumur hidup, dimana *e-KTP* ini telah dilengkapi dengan kode keamanan dan rekaman elektronik data kependudukan berbasis *database* dan nomor induk nasional (NIK) secara nasional. Adapun data yang dimaksud adalah biodata, foto, iris mata, sidik jari, dan juga tanda tangan yang diambil ketika rekaman dan tersimpan dalam fisik KTP Elektronik berupa *chip e-KTPs*[6]. Dengan menggunakan sistem biometrik sidik jari, maka setiap pemilik *e-KTP* dapat terhubung ke dalam satu *database* nasional, sehingga setiap penduduk hanya memerlukan satu KTP saja di seluruh Indonesia.

E-KTP merupakan program pemerintah yang dilatarbelakangi oleh sistem pembuatan KTP konvensional di Indonesia yang bisa saja membuat seseorang dapat memiliki lebih dari satu KTP [8]. Hal ini tentu disebabkan karena tidak adanya basis data terpadu yang merampungkan data setiap penduduk dari seluruh Indonesia. Fakta tersebut membuka peluang bagi penduduk yang memiliki niat untuk berbuat curang dalam hal tertentu dengan menggandakan KTP konvensional yang dimilikinya.

E-KTP ini terdiri dari sembilan lapisan yang tentu memiliki tingkat keamanan lebih bagus dibandingkan KTP konvensional dan pada lapisan kelima e-KTP terdapat *chip* yang menyimpan ID yang bersifat unik dari setiap e-KTP. Pada *chip* yang tertanam di dalam sebuah e-KTP mengandung data berupa biodata pemilik seperti sidik jari, tanda tangan, dan pas foto yang telah disimpan di dalam *database* kependudukan nasional dan juga telah dienkripsi dengan menggunakan algoritma kriptografi, sehingga data tersebut tidak dapat diperoleh oleh sembarang orang.



Gambar 2. 6 E-KTP Layer

Sumber: <https://slideplayer.info/slide/2950268/>

Di dalam fisik e-KTP terdapat *chip* e-KTP yang merupakan kartu pintar (*smart card*) berbasis mikroprosesor yang dilengkapi dengan besaran *memory* 8 kilobytes dan memiliki metode pengamanan data berupa autentikasi antara *chip* dengan *reader/writer* yang bersifat anti *cloning* dan dapat menjaga kerahasiaan data (*encryption*) serta tanda tangan digital pemilik e-KTP. *Chip* tersebut tidak dapat dideteksi hanya dengan kasat mata, *chip* tersebut hanya bisa dibaca oleh *card reader* seperti RFID.

Perlu diketahui bahwa, antarmuka *chip* e-KTP tersebut memenuhi standar ISO 14443 A atau ISO 14443 B. *Chip* e-KTP tersebut ditanam di antara lapisan yang terdapat di dalam fisik e-KTP berupa plastik putih yang transparan pada dua layer teratas (dilihat dari depan). *Chip* ini juga memiliki antena di dalamnya yang dapat mengeluarkan gelombang apabila

digesek. Gelombang inilah yang akan dikenali oleh alat pendeteksi *e-KTP* seperti *RFID reader*, sehingga dapat diketahui apakah *e-KTP* tersebut berada di tangan orang yang benar atau tidak.

Teknologi yang ada pada *e-KTP* memberikan berbagai manfaat bagi masyarakat, karena *KTP elektronik* dipergunakan untuk beragam keperluan seperti mengurus registrasi pelayanan kesehatan, mempermudah registrasi di berbagai instansi, pengurusan membuat paspor, dan lain sebagainya. Namun dengan adanya *e-KTP* membutuhkan pembaca *standard* berupa *RFID reader* untuk dapat mengidentifikasi dan membaca isi dari *chip* yang terdapat pada fisik *e-KTP* tersebut.

2.1.7 JSON (*Javascript Object Notation*)

JSON adalah sebuah format data yang digunakan untuk pertukaran data antar *device* atau sistem operasi atau bahasa pemrograman yang berbeda dan juga digunakan sebagai media penyimpanan data. JSON merupakan bagian dari Javascript dan bisa dibaca oleh berbagai macam bahasa pemrograman seperti C, C++, Java, Javascript Perl, Python dan banyak lagi. Jika dilihat dari strukturnya, JSON dibagi menjadi dua, yaitu :

1. Objek

Sebuah objek JSON ialah format data dengan bentuk pasangan *key:value* yang biasanya di *render* di dalam sebuah kurung kurawal. *Key* dapat berupa *string* apapun yang *valid*. Namun di dalam objek ini, *key* haruslah unik, *key* juga dapat memiliki spasi namun akan membuat kita lebih sulit ketika akan mengaksesnya di proses ngoding sehingga perlu menggunakan *underscore* seperti *golongan_darah*. *Value* boleh diisi dengan enam tipe data dasar, seperti *strings*, *numbers*, *objects*, *arrays*, *booleans* (*true* atau *false*), dan *null*. Contoh :

```
{"nama":"Budi Sitorus", "golongan_darah":"A"}
```

2. Array adalah bentuk sederetan *value* (meskipun *value* tersebut juga dapat berbentuk objek). Dalam format JSON, *value* disimpan dengan menggunakan karakter kurung siku pembuka dan penutup. Antara *value* yang satu dengan *value* yang lain dipisahkan dengan tanda koma. Contoh:

```
["Budi Sitorus", "A"]
```

2.1.8 Database MySQL

Database ialah kumpulan atau kesatuan dari data yang membentuk sebuah berkas atau file yang memiliki keterhubungan (*relation*) dengan tata cara tertentu untuk membentuk suatu data baru atau informasi. MySQL adalah *software* atau perangkat lunak *database management system* (sistem manajemen basis data) yang menggunakan perintah dasar *Structure Query Language* (SQL) dan bersifat *open source*. MySQL menjadi pilihan yang dominan bagi para pengembang *software* dan aplikasi, karena MySQL sendiri memiliki berbagai kelebihan yaitu mudah dimengerti, mudah diprogram, dan juga didukung oleh banyak bahasa pemrograman seperti C, C++, Java, PHP, dan Python.

2.2 Related Work

2.2.1 Sistem Akses Kontrol Kunci Elektrik Menggunakan Pembacaan e-KTP

Sistem akses kontrol kunci elektrik menggunakan pembacaan e-KTP merupakan sebuah sistem kunci elektrik yang memanfaatkan RFID *reader* sebagai alat untuk membaca e-KTP dengan frekuensi gelombang kerja senilai 13.56 MHz. Kunci elektrik ini dilengkapi dengan modul *database* yang terintegrasi secara *wireless* UART (*Universal Asynchronous Receiver Transmitter*) [9]. Sistem ini menggunakan *database* dan juga *logger*. *Database* digunakan untuk menyimpan *unique* ID dari e-KTP yang berhasil dibaca oleh RFID dan *logger* digunakan untuk sistem pencatatan presensi dari setiap aktivitas masuk ruang. Proses kerja sistem ini ialah dengan membaca *unique* ID dari setiap chip yang tertanam di dalam e-KTP dan selanjutnya mencocokkan ID yang berhasil dibaca dengan ID yang telah diregistrasi dan tercatat pada *database*. Kemudian *system* akan mengirimkan umpan balik terhadap modul kunci. Hasil komparasi *unique* ID e-KTP tersebut dicatat dalam *logger* yang menyimpan data nama dari pengakses ruang, nomor *unique* yang dimiliki oleh setiap e-KTP, juga tanggal dan waktu akses ruang. Untuk proses penambahan ataupun pengurangan hak akses ruang dapat diterapkan dengan mengubah data pada *file* dalam *database*, sehingga proses tersebut dilakukan tanpa mengubah sedikitpun program yang ada pada mikrokontrolernya. Dari hasil pengujian tersebut RFID *reader* dapat mendeteksi e-KTP pada jarak maksimal 3 cm dan jarak tersebut bisa saja berkurang apabila antena RFID *reader* terhalang oleh *obstacle* bersifat non logam. Hasil pengujian pendeteksian data dari e-KTP menggunakan RFID *reader* ini menunjukkan bahwa sistem yang diuji dapat dipisahkan antar ruang dalam lingkup jarak 25 hingga 30 meter. Oleh karena itu, penulis berhasil untuk membangun sebuah sistem kunci *electric* untuk menjamin keamanan dengan memanfaatkan teknologi yang ada pada e-KTP dan *reader standard* RFID mewakili fungsi sebagai kunci yang memudahkan akses kontrol pintu dan menambah pemanfaatan dari e-KTP sebagai kemajuan dalam kehidupan sehari-hari.

2.2.2 Sistem Akses Pintu Menggunakan e-KTP Sebagai Kunci Elektronik Berbasis Near Field Communication Dimonitor Melalui Jaringan Komputer

Sistem ini memanfaatkan RFID *Reader* sebagai *reader* e-KTP (13.56 MHz) dan melakukan identifikasi sidik jari *user* menggunakan FPM10a yang terintegrasi pada mikrokontroler Arduino Mega 2560 digunakan untuk mengontrol solenoid sebagai kunci elektronik yang akan diterapkan pada pintu. Setiap mikrokontroler didukung oleh *ethernet shield* yang memiliki IP *address* untuk dapat terhubung ke monitor *server* melalui jaringan komputer. Monitor *server* berisi *database* berfungsi untuk menyimpan data *user* dan riwayat *user* yang mengakses pintu [12]. Proses membaca *Unique Identification* (UID) e-KTP oleh RFID efektif pada jarak maksimal 2 cm atau 3 cm. Setiap *user* yang memiliki hak dalam akses membuka pintu sebelumnya harus telah mendaftarkan e-KTP miliknya dan sidik jarinya sehingga sudah tercatat dalam sistem *database server*. Hak akses membuka pintu tersebut dilakukan melalui tahapan pembacaan UID dan identifikasi sidik jari *user* yang terdaftar di dalam basis data yang selanjutnya akan diproses oleh mikrokontroler dengan tujuan untuk mengaktifkan solenoid. Selanjutnya, akses keluar *user* dilakukan dengan menggunakan tombol *exit* pada sistem. Dengan penerapan sistem jaringan komputer tentu akan memudahkan pemantauan keseluruhan aktivitas akses ke semua pintu secara *real time* hanya dengan monitor *server* tunggal saja. Pemanfaatan e-KTP sebagai kunci elektronik pada sebuah pintu dan pengamanan dengan menggunakan *fingerprint* secara teknis menjadikan sistem akses keluar masuk melalui pintu menjadi tidak mudah untuk dibuka oleh orang lain.

BAB III ANALISIS DAN PERANCANGAN

Pada bab ini, akan dijelaskan analisis yang dilakukan terhadap tahapan sistem perangkat keras dengan OCR (*Optical Character Recognition*) menggunakan *tools* Tesseract dan juga tahapan perancangan perangkat lunak yang mencakup analisis masalah, analisis pemecahan masalah, analisis kebutuhan sistem, perancangan dan desain yang menjadi lingkup utama dalam tugas akhir ini.

3.1. Analisis

Analisis yang dibahas pada sub bab ini merupakan analisis masalah, analisis pemecahan masalah, dan analisis kebutuhan sistem untuk mendapatkan solusi dari permasalahan yang terjadi.

3.1.1 Analisis Masalah

Indonesia telah resmi memulai program KTP elektronik sejak tahun 2009. Program KTP elektronik dilatarbelakangi oleh sistem pembuatan KTP konvensional di Indonesia yang memungkinkan seseorang dapat memiliki lebih dari satu KTP. Sementara, kelebihan e-KTP ini belum digunakan secara maksimal dalam kehidupan sehari-hari.

Contohnya ialah ketika kita menjadi penumpang di sebuah kapal. Saat ini sistem pengarsipan data terkait penumpang sebuah kapal masih dilakukan secara konvensional yaitu menggunakan buku, kertas dan alat tulis, hal tersebut tentu tidak praktis dalam mengolah data. Ketika kita ingin menjadi penumpang sebuah kapal maka kita harus membeli tiket terlebih dahulu. Pada proses pembelian tiket ini data berupa informasi pribadi berdasarkan e-KTP tidak dibutuhkan. Namun, ketika kita ingin memasuki kapal, maka kita akan menunjukkan tiket tersebut dan juga menunjukkan e-KTP kita. Hal ini akan jadi masalah ketika penumpang sebuah kapal begitu ramai, sehingga penumpang dan petugas kapal akan kewalahan mencatat identitas diri dari e-KTP yang bisa saja menyebabkan terjadinya kesalahan penulisan.

Dalam proses mengambil data pribadi penumpang dari e-KTP, maka petugas kapal juga harus mengecek e-KTP milik penumpang tersebut. Petugas kapal belum tentu bisa mengenali apakah e-KTP yang ia bawa adalah asli maupun palsu. Sementara, data mengenai penumpang kapal haruslah valid.

Oleh karena itu kita memerlukan peningkatan sistem pencatatan informasi pribadi secara digital dengan menggunakan alat pembaca data e-KTP. Penggunaan alat ini juga bertujuan untuk memastikan data seseorang (penumpang) tercatat lebih lengkap, terhindar dari

kesalahan penulisan, valid, serta bisa dijadikan dasar dalam proses klaim jika terjadi kecelakaan transportasi (untuk *study case* kapal penumpang).

3.1.2 Analisis Pemecahan Masalah

Pemecahan masalah yang dilakukan untuk sistem yang berjalan saat ini yaitu penulis membangun e-KTP *Scanner using Raspberry*. Perangkat ini merupakan alat yang berfungsi untuk membaca dan mengekstrak informasi dari *chip* yang berada di dalam e-KTP menjadi informasi digital.

Sistem e-KTP *scanner* ini menggunakan *Optical Character Recognition* Tesseract untuk mengekstrak informasi dari e-KTP. Alat ini menggunakan RFID *reader* dan juga kamera digital untuk membaca informasi dari e-KTP seseorang. Dengan menggunakan alat ini, maka ketika ada proses registrasi yang membutuhkan data berupa informasi pribadi dari e-KTP, maka kita hanya perlu melakukan *scan* e-KTP kita terhadap sistem yang akan dibangun pada tugas akhir ini.

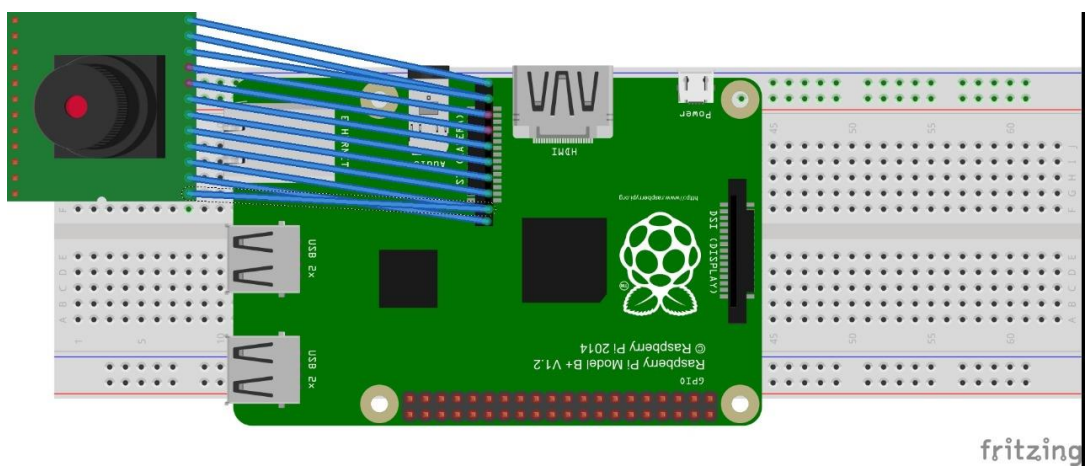
Sistem ini akan memperoleh data dari e-KTP dengan cepat dan memiliki tingkat keakuratan data penumpang yang tinggi. Informasi yang dibaca oleh kamera dan RFID *reader* akan di ekstrak dan ditampilkan di sebuah *web* yang dibangun dengan *Application Programming Interface* (API) dengan format JSON.

3.1.3 Analisis Kebutuhan Sistem

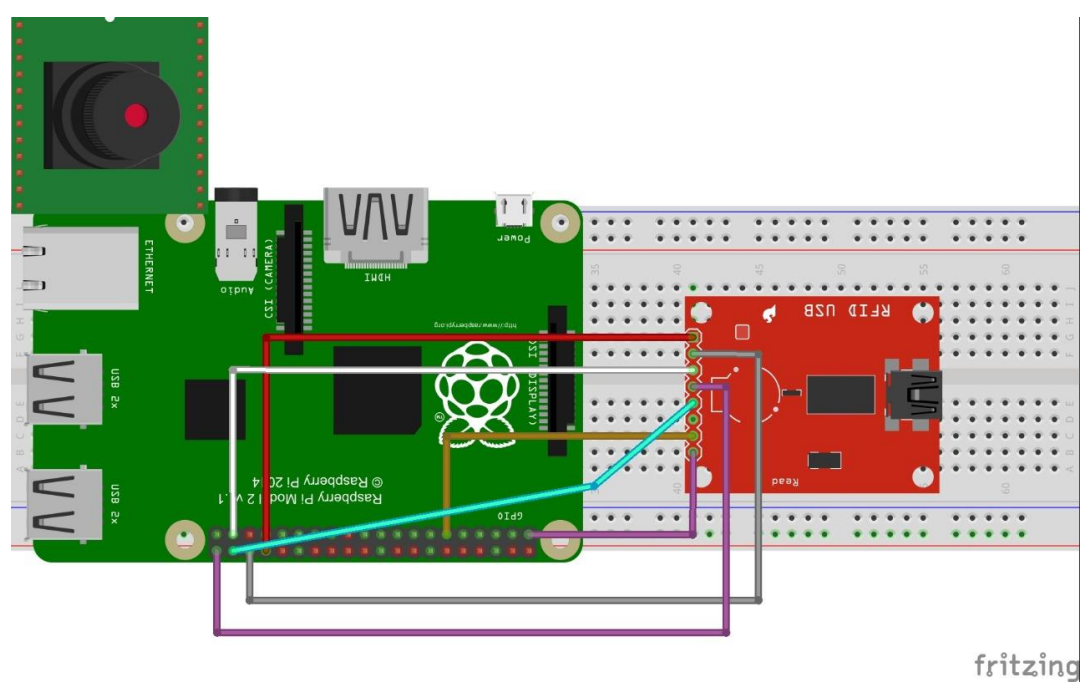
Pada sub bab ini dijelaskan kebutuhan apa saja yang digunakan untuk membangun e-KTP *Scanner using Raspberry*. Kebutuhan sistem mencakup kebutuhan perangkat keras dan perangkat lunak.

3.1.3.1 Analisis Kebutuhan Perangkat Keras (*Hardware*)

Terdapat 3 komponen utama yang diimplementasikan ke dalam *E-KTP Scanner using Raspberry*. Komponen pertama adalah Raspberry Pi Modul 3 B. *Raspberry Pi* Modul 3 B merupakan komputer yang berukuran kecil yang dihubungkan ke layar komputer. Raspberry Pi Modul 3 B berfungsi untuk media yang disambungkan dengan kamera modul dan juga RFID *reader*. Komponen kedua adalah kamera modul yang digunakan untuk mengambil gambar bagian depan e-KTP, dan kemudian mengekstrak gambar dari e-KTP yang diambil oleh *camera* ke informasi digital menggunakan teknologi OCR dengan tools Tesseract. Komponen ketiga adalah RFID *reader* yang digunakan untuk membaca dan mengekstrak informasi dari chip yang berada dalam e-KTP ke informasi digital dan menampilkan informasi tersebut menggunakan *web* API dengan format JSON.



Gambar 3. 1 Kamera



Gambar 3. 2 RFID Reader

Perangkat keras yang digunakan untuk membangun sistem e-KTP *Scanner using Raspberry* terdapat pada tabel.

Tabel 3. 1 Daftar Spesifikasi *Hardware*

Nama Alat dan Kebutuhan	Spesifikasi	Keterangan
Monitor	Model : D186wA A : 1,5A Frekuensi : 50/60 Hz Tegangan : 100 – 240VAC	Menjalankan Raspberry OS raspbian.
Keyboard	<i>Model : ku-1153</i>	Digunakan untuk meng- <i>input</i> semua konfigurasi.
<i>Mouse</i>	Merek : Logitech Tipe : <i>Wired</i>	Perangkat keras <i>input</i> yang mengatur pergerakan kursor dengan cepat.
Kabel <i>jumper</i>	Kabel jumper <i>male to female</i> <i>Panjang kabel : 20 cm</i>	Menyambungkan komponen elektronik dengan yang lainnya pada saat membuat <i>project</i> dengan menggunakan <i>breadboard</i> .
<i>Breadboard</i>	Breadboard 8.5x5.5cm Solderless PCB Circuit <i>Board 400 Lubang</i>	Dasar konstruksi sebuah sirkuit elektronik.
<i>Converter HDMI</i>	HDMI 1.3b dan <i>Analog Video output</i> hingga UXGA dan 1080P	Mengubah <i>input</i> dari perangkat HDMI menjadi <i>output</i> untuk disambungkan ke Raspberry.
<i>LED 5MM white</i>	Tegangan : 3,0 V – 3,6 V <i>Wavelength</i> : 450nm Semikonduktor : Gallium Indium Nitride (GaInN)	Sumber cahaya yang akan membantu camera dalam mengambil gambar e-KTP yang lebih jelas

Nama Alat dan Kebutuhan	Spesifikasi	Keterangan
<i>OV7670 Camera Board 0.3 Megapixel (B)</i>	<p><i>Chip : OV7670</i></p> <p><i>Power input : 2.5 V - 3.3 V</i></p> <p><i>Resolusi : 640 x 480 px (VGA), 320 x 240 px (QVGA), 352 x 240 px (CIF), dan 176 x 144 px (QCIF)</i></p> <p><i>Transfer rate : 30 fps</i></p> <p><i>Lensa : 1/6", dengan sudut penglihatan 24 derajat</i></p> <p><i>Sensitivitas : 1.3 V/Luc-sec</i></p> <ul style="list-style-type: none"> <i>• SNR : 46 dB</i> <i>• Dynamic Range : 52 dB</i> <i>• View Mode : Progressive</i> <i>• Ukuran board : 36 x 34 x 25 mm</i> 	<p>Mengambil gambar dari e-KTP.</p>
<p><i>USB 2.0</i></p>	<p><i>Power output 2.5W</i></p> <p><i>Max transfer speed 480 Mbit/s</i></p>	<p>Digunakan untuk menghubungkan Raspberry ke laptop</p>
<p><i>RC522 RFID Reader</i></p>	<p><i>Working current : 13—26mA/ DC 3.3V</i></p> <p><i>Standby current : 10-13mA/DC 3.3V</i></p> <p><i>Sleeping current : <80 uA</i></p> <p><i>Peak current : <30mA</i></p> <p><i>Frekuensi kerja : 13.56 MHz</i></p> <p><i>Jarak pembacaan : 0~60 mm mifare1 card</i></p> <p><i>Kecepatan komunikasi data hingga 10Mbit/s</i></p>	<p>Membaca id dari e-KTP</p>
<p>Papan akrilik</p>	<p><i>Ukuran : 10 x 6 cm</i></p> <p><i>Tebal : 1.5 mm</i></p>	<p>Sebagai media untuk meletakkan e-KTP saat di <i>scan</i></p>

Nama Alat dan Kebutuhan	Spesifikasi	Keterangan
Raspberry Pi 3 Model B	<p><i>Processor : Tipe Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz</i></p> <p><i>Memory : 1GB LPDDR2 SDRAM</i></p> <p><i>Power Supply Raspberry : 5V/2.5A DC power input (microUSB)</i></p> <p><i>Wireless Antena : 2.4GHz</i></p> <p><i>Wireless LAN : 5GHz IEEE 802.11.b/g/n/ac</i></p> <p><i>Bluetooth : Version 4.2 BLE</i></p> <p><i>Ethernet : Gigabit Ethernet over USB 2.0 with maximum throughput 300 Mbps</i></p> <p><i>GPIOs : Extended 40-pin GPIO header</i></p> <p><i>Video Output : Full-size HDMI</i></p> <p><i>Audio Output : 4-pole stereo output and composite video port</i></p> <p><i>USB Port : 4 USB 2.0 ports</i></p> <p><i>CSI camera port : for connecting a Raspberry Pi camera</i></p> <p><i>DSI display port : for connecting a Raspberry Pi touchscreen display</i></p> <p><i>Micro SD port for loading your operating system and storing data</i></p> <p><i>Power-over-Ethernet (PoE) that support (requires separate PoE HAT)</i></p> <p><i>Operating Temperature: 0~50 °C</i></p> <p><i>Dimension Rpi: 120mm x 75mm x 34mm</i></p> <p><i>Weight : 75g</i></p>	Mengendalikan dan mengelola data dari e-KTP yang diterima dari kamera digital dan RFID

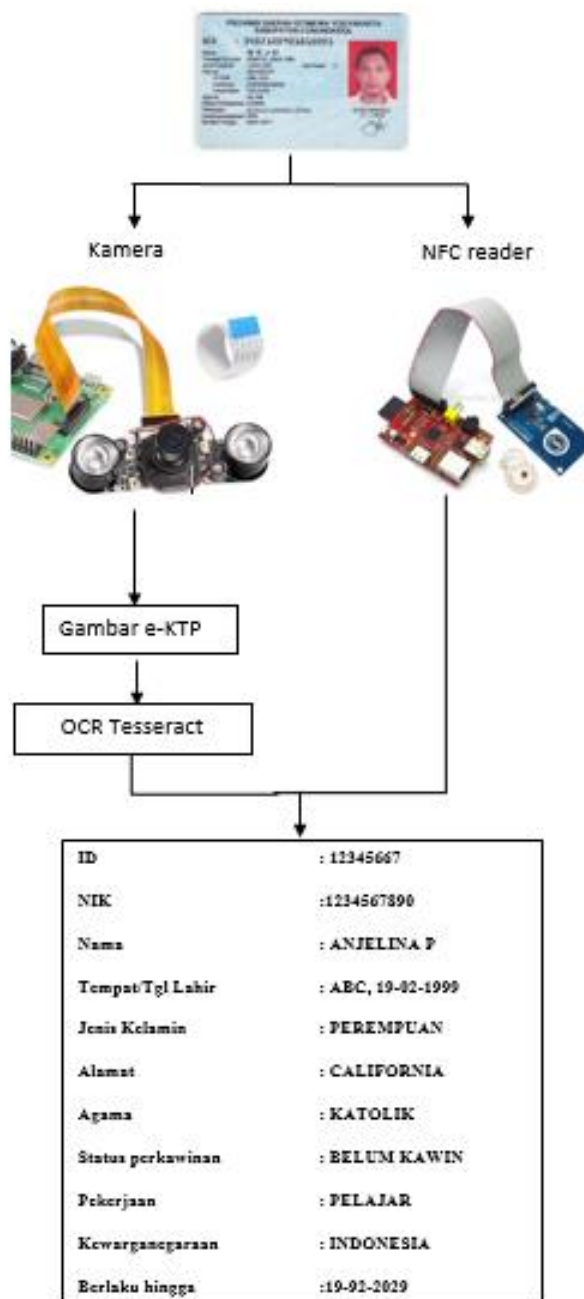
3.1.3.2 Analisis Kebutuhan Perangkat Lunak (Software)

Membangun sistem e-KTP *Scanner using Raspberry pi* membutuhkan perangkat lunak untuk melakukan pengoperasian dan proses dalam pengolahan masukan data. Perangkat lunak yang digunakan untuk mengolah data, menyimpan data, untuk membangun sistem *scanner* e-ktp yaitu:

1. Tesseract OCR merupakan salah satu jenis dari Optical Character Recognition (OCR) engine yang bersifat *open source*, memiliki fungsi untuk mengekstrak informasi yang diperoleh dari gambar atau *object* menjadi informasi berupa teks
2. Raspbian adalah OS yang berbasis distribusi Debian. Raspbian dirancang khusus untuk bekerja pada perangkat Raspberry Pi.
3. Autocad adalah sebuah program yang digunakan untuk menggambar 2 dimensi dan 3 dimensi
4. Sublime adalah sebuah *editor* pemrograman
5. *Browser* digunakan sebagai pengaksesan *web*
6. *Programming language* Python dan PHP digunakan untuk membangun *web*.

3.1.4 Perancangan Pembagunan Sistem

Perancangan e-KTP *Scanner using Raspberry Pi* ini mencakup perancangan perangkat keras sistem, perancangan perangkat lunak sistem, dan *design* sistem. Perancangan ini digunakan sebagai acuan dalam mengimplementasikan sistem pada tugas akhir. Dengan perancangan ini, segala kemungkinan yang dapat menghambat pengerjaan tugas akhir ini dapat diminimalkan.



Gambar 3. 3 Skema Alur Kerja Sistem

Berdasarkan Gambar 7, e-KTP akan di *scan* menggunakan RFID *reader* dan juga kamera digital, informasi dari e-KTP yang akan ditampilkan di *web API*. Pada Raspberry Pi akan dilakukan proses ekstrak informasi dari gambar e-KTP. Kemudian data yang sudah diproses akan ditampilkan di *web API* yang sudah dibangun yang diakses dengan komputer yang dibangun dengan format JSON.

Melalui perancangan sistem diatas, berikut proses yang dilakukan perangkat:

1. E-KTP akan diarahkan/di-*scan* ke perangkat yang telah dilengkapi oleh RFID *reader* dan juga kamera digital.
2. Kamera digital akan merekam dan memperoleh sebuah gambar e-KTP.
3. Gambar yang diperoleh tersebut akan diproses oleh Tesseract OCR sehingga bisa di ekstrak ke bentuk informasi digital.
4. Sedangkan RFID *reader* berfungsi untuk membaca dan memindai (*scanning*) informasi yang ada pada *chip* e-KTP yang berisikan ID e-KTP yang kemudian akan diproses oleh aplikasi komputer.
5. Pada Raspberry akan dilakukan pengolahan data informasi dari e-KTP tersebut. Pada Raspberry pi akan dibangun sebuah *web* API yang menampilkan informasi dari KTP. Dari *web* API *user* dapat mengetahui informasi dari hasil ekstraksi e-KTP yang menggunakan bahasa pemrograman Python dan diakses menggunakan IP *Address*. Bahasa Python adalah bahasa pemrograman yang memiliki banyak fungsi interaktif berorientasi objek dan merupakan bahasa pemrograman tingkat tinggi, merupakan bahasa pemrograman formal dengan aturan-aturan dan format spesifikasi sendiri.
6. Informasi yang ditampilkan pada *web* API adalah dalam bentuk format JSON.

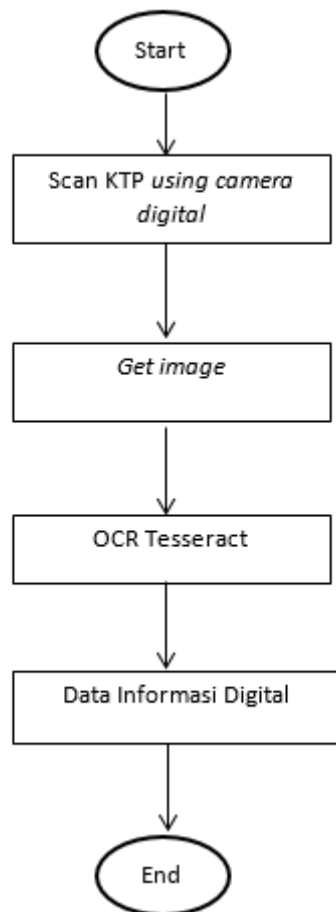
3.1.5 Flowchart Sistem

Perancangan *flowchart* sistem bertujuan untuk dapat melihat gambaran dari alur kerja suatu sistem, agar sistem dapat diimplementasikan sesuai alur kerja yang telah dirancang sebelumnya. Dalam *flowchart* ini digambarkan bagaimana alur kerja OCR dan RFID *reader* membaca dan mengekstrak informasi dari e-KTP.

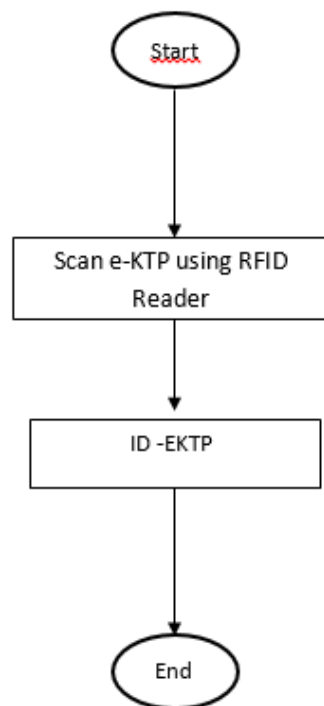
Pada *flowchart* digambarkan alur bagaimana proses masuknya data dari RFID dan *camera*. Foto bagian depan fisik e-KTP akan diambil *camera* modul Raspberry Pi dan selanjutnya diproses oleh *library* Tesseract OCR untuk diubah menjadi informasi digital.

Kemudian, *ccip* yang tertanam di dalam e-KTP akan di deteksi dan dibaca oleh RFID *reader*. Selanjutnya RFID *reader* akan memperoleh informasi berupa *unique* ID. ID tersebut akan dimasukkan ke dalam *database* tepatnya pada tabel id.

Kedua informasi tersebut akan dipadukan dan dilakukan *join query* pada *database* sehingga dapat ditampilkan pada *web* yang dibangun.



Gambar 3. 4 Flowchart OCR



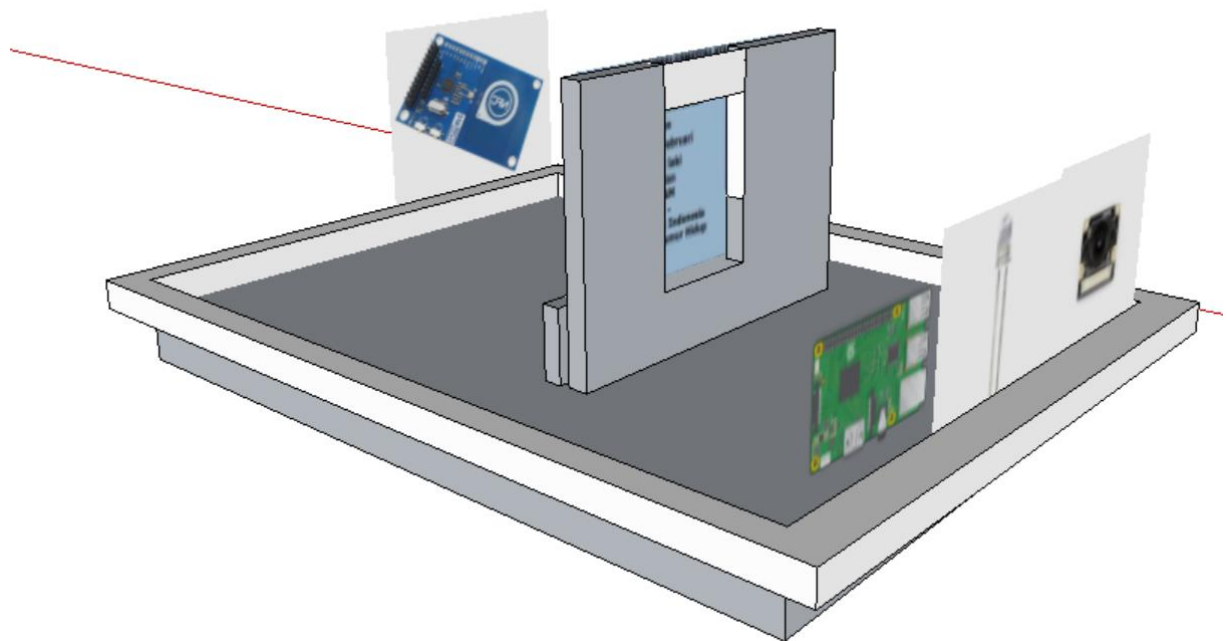
Gambar 3. 5 Flowchart RFID Reader

Gambar tersebut menjelaskan bahwa e-KTP di *scan* untuk mendapatkan informasi melalui dua tahapan yaitu yang pertama untuk membaca informasi dari e-KTP dengan menggunakan kamera dan untuk membaca ID dari *chip* e-KTP menggunakan RFID *reader*. Informasi yang diperoleh dari hasil *scan* akan di tampilkan di *web API (Application Programming Interface)* dengan format JSON.

3.2 Desain

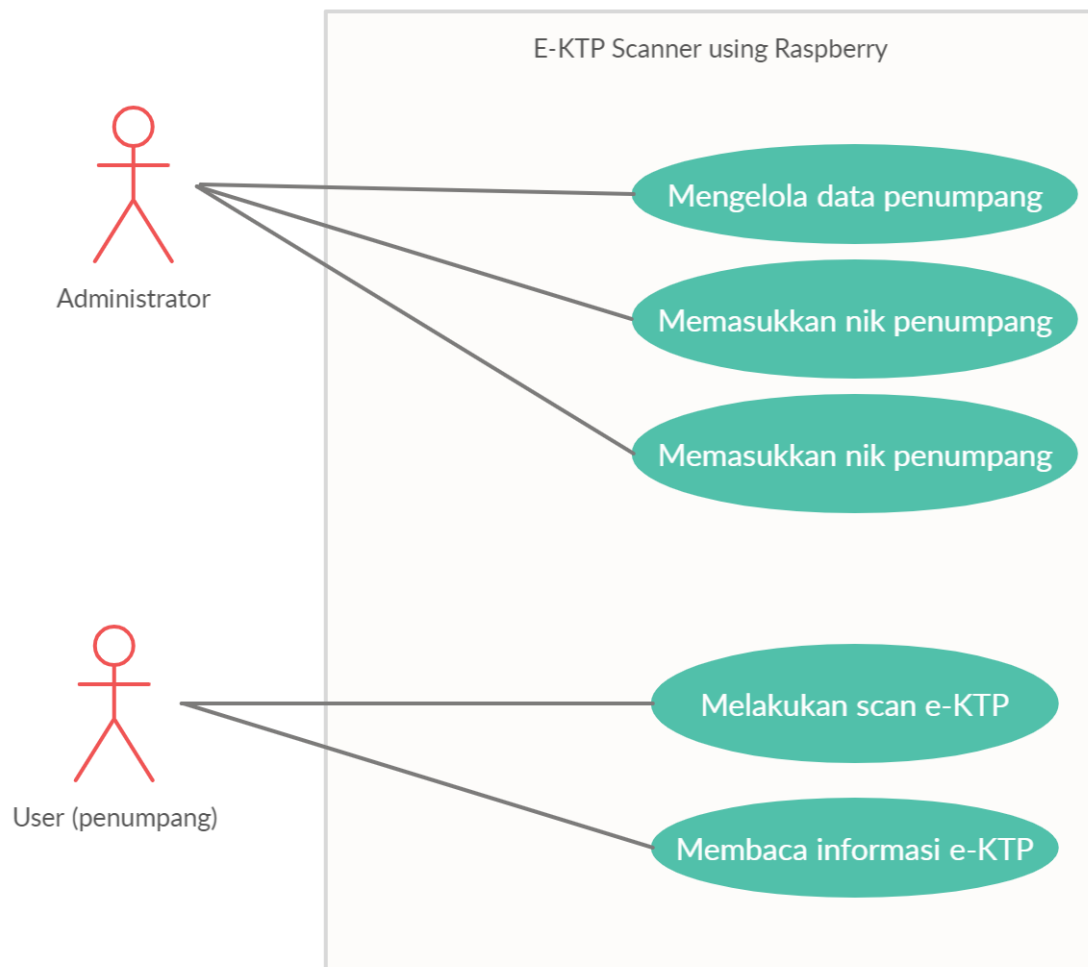
3.2.1 Desain Prototipe

Untuk membangun e-KTP *Scanner using Raspberry* ini, maka *developer* menggunakan RFID *reader* dan juga kamera digital. Pada gambar dibawah ini, dapat dilihat bahwa perangkat yang akan dibangun terlihat seperti papan permainan, pada bagian tengah terdapat slot yang akan digunakan untuk meletakkan e-KTP, pada bagian kanan terdapat Raspberry pi dan juga kamera, yang akan digunakan untuk mengambil *images* KTP. Di sebelah kiri, terdapat RFID *reader* yang akan digunakan untuk mengambil data dari id e-KTP.



Gambar 3. 6 Prototipe e-KTP Scanner

3.2.2 Use Case Diagram



Gambar 3. 7 Use Case

Dari gambar dijelaskan bahwa *user* dapat melakukan dua hal untuk memperoleh data dari e-KTP. Tahap pertama adalah melakukann *scanning*, selanjutnya hasil *scan* yang diperoleh oleh kamera ialah berupa *image* dan data yang akan dibaca oleh RFID ialah id e-KTP. *Image* yang diambil kamera akan diubah formatnya oleh OCR (*Optical Character Recognition*) sehingga text yang ada pada gambar akan diubah ke format digital. Kemudian hasil dari proses tersebut akan ditampilkan dalam *web* yang dibangun dan bisa dibaca langsung oleh pemilik e-KTP. Selain itu, untuk memperoleh data penumpang, administrator juga akan mendaftarkan NIK setiap penumpang dan id kapal yang ingin ditumpangi.

3.2.3 Desain Contoh Implementasi Aplikasi Pencatatan Penumpang Kapal

Sistem pencatatan data penumpang kapal berdasarkan e-KTP saat ini dilakukan secara manual yaitu mengisi formulir berupa buku atau kertas dengan tulisan tangan yang memakan banyak waktu dan menyebabkan antrian yang panjang. Hal ini dapat diatasi dengan adanya e-KTP *scanner* dan aplikasinya yang dibangun menggunakan bahasa pemrograman PHP.

Pada gambar dibawah ini terdapat desain menu registrasi yang akan dilakukan oleh administrator. Administrator akan memasukkan nik setiap penumpang yang akan membeli tiket dan jenis kapal apa yang akan dinaiki oleh penumpang.

Gambar 3. 8 Desain menu registrasi

Desain aplikasi penyebrangan kapal yaitu seperti gambar dibawah ini, terdapat beberapa jenis kapal dengan waktu keberangkatan yang berbeda. Masing masing menu (Ihan Batak, Ferry, dan Dengke) akan menampilkan informasi dari seluruh penumpang yang berhasil melakukan *scan* e-KTP.

Gambar 3. 9 Dashboard Aplikasi

Gambar 3.9 merupakan desain tampilan halaman untuk menu Ihan Batak. Adapun informasi yang ditampilkan ialah nama penumpang, alamat, dan nomor kendaraan (jika ada). Nama penumpang dan alamat akan diambil dari hasil *scan* yang dilakukan oleh penumpang itu sendiri, sedangkan nomor kendaraan akan diisi oleh petugas kapal secara manual.

Gambar 3. 10 Tampilan Menu Kapal Ihan Batak

Untuk kembali melakukan *scan* e-KTP bagi penumpang yang akan memasuki kapal sesuai dengan jenis atau id kapal yang akan ia pilih, maka petugas kapal akan klik *button* tambah penumpang, sehingga tampil halaman berikut.

The screenshot shows a web application titled "Informasi Kapal". At the top, there are navigation icons (back, forward, home) and a search bar. Below the header, the form contains the following empty input fields:

- NIK:
- Nama :
- Tempat/Tgl
- Alamat:
- RT/RW:
- Kel/Desa
- Kecamatan:
- Agama:
- Status
- Perkerjaan:
- Kewarganeg
- Berlaku

Gambar 3. 11 Tampilan Sebelum Melakukan Scan

Jika kita berhasil melakukan *scan* e-KTP, maka datanya akan tampil seperti gambar berikut.

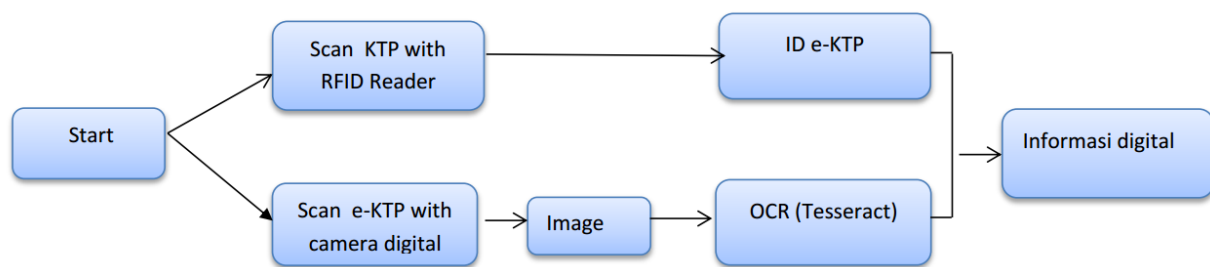
The screenshot shows the same "Informasi Kapal" web application, but now the input fields are populated with scanned data. The address bar shows the URL "localhost/ibad/TA2/log". The displayed data is as follows:

- NIK: 121214xxxxxxxxxx
- Nama: JONAR PANJAITAN
- Tempat/Tgl Lahir: MEDAN/ 26-04-1968
- Alamat: SIMPANG 4
- RT/RW: 000/000
- Kel/Desa: NARUMONDA I
- Kecamatan: SIANTAR NARUMONDA
- Agama: KRISTEN
- Status Perkawinan: BELUM KAWIN
- Perkerjaan: PELAJAR/MAHASISWA
- Kewarganegaraan: WNI
- Berlaku Hingga: SEUMUR HIDUP

Gambar 3. 12 Informasi Hasil scan e-KTP

3.2.4 Desain Arsitektur Sistem

Pengerjaan tugas akhir ini akan lebih mudah karena memiliki desain arsitektur yang dirancang menggunakan beberapa *hardware*. Desain untuk e-KTP *scanner* dapat dilihat pada gambar.



Gambar 3. 13 Desain Arsitektur Sistem

Gambar tersebut menjelaskan bahwa e-KTP di *scan* untuk mendapatkan informasi melalui dua tahapan yaitu yang pertama untuk membaca informasi dari e-KTP dengan menggunakan kamera dan untuk membaca ID dari *chip* e-KTP menggunakan *RFID reader*. Informasi yang diperoleh dari hasil *scan* akan ditampilkan di *web API (Application Programming Interface)*.

3.2.5 Perancangan Desain Database

3.2.5.1 Struktur Tabel

1. Nama : Tabel id
- Log Deskripsi Isi : Berisi no, idktp
- Primary Key : no
- Volume : 2 fields
- Detail Description : Detail tabel id dapat dilihat pada tabel

Tabel 3. 2 Tabel id

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
No	Menyimpan jumlah scan yang dilakukan	Int (11)	No	None	Auto Increment, Primary Key
Idktp	Menyimpan chip dari e-KTP	Varchar (25)	No	None	

2. Nama : Tabel ktp

Log Deskripsi Isi : Berisi nomor, nik, nama, ttl, jkl, alamat, rt, kel, kec, agama, status, pekerjaan, kewarganegaraan, berlaku

Primary Key : nomor

Volume : 14 fields

Detail Description : Detail tabel id dapat dilihat pada tabel

Tabel 3. 3 Tabel Ktp

ID Field	Deskripsi	Tipe & Length	Boleh nul	Default	Keterangan
Nomor	Menyimpan jumlah ktp yang telah di tangkap oleh <i>camera</i>	Int (12)	No	None	Auto Increment, Primary Key
NIK	Menyimpan nomor induk keluarga dari user	Varchar (25)	Yes	None	
Nama	Menyimpan data atribut nama dari <i>user</i>	Varchar (25)	Yes	None	
Ttl	Menyimpan data atribut tempat tanggal lahir	Varchar (25)	Yes	None	
Jkl	Menyimpan data atribut jenis kelamin dari <i>user</i>	Varchar (9)	Yes	None	

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
Alamat	Menyimpan data atribut alamat dari user	Varchar (25)	Yes	None	
RT	Menyimpan data atribut alamat RT dari user	Varchar (25)	Yes	None	
Kel	Menyimpan data atribut alamat kelurahan user	Varchar (25)	Yes	None	
Kec	Menyimpan data atribut alamat kecamatan dari user	Varchar (25)	Yes	None	
Agama	Menyimpan data atribut agama dari user	Varchar (25)	Yes	None	
Status	Menyimpan data atribut status dari user	Varchar (25)	Yes	None	
Pekerjaan	Menyimpan data atribut pekerjaan dari user	Varchar (25)	Yes	None	
Kewarganegaraan	Menyimpan data atribut kewarganegaraan dari user	Varchar (25)	Yes	None	
Berlaku	Menyimpan data atribut waktu berlaku e-KTP dari user	Varchar (25)	Yes	None	

3. Nama : Tabel kapal

Log Deskripsi Isi : Berisi idkapal, namakapal, waktu

Primary Key : idkapal

Volume : 3 fields

Detail Description : Detail tabel id dapat dilihat pada tabel

Tabel 3. 4 Tabel Kapal

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
Id_kapal	Menyimpan id unik dari kapal yang berbeda	Int (12)	No	None	Primary Key
Namakapal	Menyimpan nama kapal	Varchar (25)	No	None	
Waktu	Menyimpan data waktu keberangkatan kapal	Varchar (25)	Yes	None	

4. Nama : Tabel daftar penumpang

Log Deskripsi Isi : Berisi nomor, nama penumpang, nik penumpang, nama kapal, id kapal.

Primary Key : nomor

Volume : 5 fields

Tabel 3. 5 Daftar Penumpang

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
nomor	Menyimpan jumlah dari data penumpang yang bersifat <i>auto increment</i>	Int (12)	No	None	Primary Key
Nama penumpang	Menyimpan nama penumpang	Varchar (25)	No	None	
Nik penumpang	Menyimpan NIK penumpang	Varchar (25)	No	None	
Nama kapal	Mneyimpan nama kapal	Varchar (25)	No	None	
Id Kapal	Menyimpan id kapal	Int (11)	No	None	

5. Nama : Tabel Dengke

Log Deskripsi Isi : Berisi id_kapal, namapenumpang, alamat

Primary Key : id_kapal

Volume : 3 fields

Tabel 3. 6 Dengke

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
Id_kapal	Menyimpan informasi id kapal	Int (12)	No	None	Primary Key
namapenumpang	Menyimpan nama penumpang	Varchar (25)	No	None	
alamat	Menyimpan alamat penumpang	Varchar (25)	No	None	

6. Nama : Tabel Ferry

Log Deskripsi Isi : Berisi id_kapal, namapenumpang, alamat

Primary Key : id_kapal

Volume : 3 fields

Tabel 3. 7 Ferry

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
Id_kapal	Menyimpan informasi id masing-masing kapal	Int (12)	No	None	Primary Key
namapenumpang	Menyimpan nama penumpang	Varchar (25)	No	None	
alamat	Menyimpan alamat penumpang	Varchar (25)	No	None	

7. Nama : Tabel ihanbatak

Log Deskripsi Isi : Berisi id_kapal, namapenumpang, alamat

Primary Key : id_kapal

Volume : 3 fields

Tabel 3. 8 Ihan Batak

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
Id_kapal	Menyimpan informasi id kapal	Int (12)	No	None	Primary Key
namapenumpang	Menyimpan nama penumpang	Varchar (25)	No	None	
alamat	Menyimpan alamat penumpang	Varchar (25)	No	None	

8. Nama : Tabel ektp

Log Deskripsi Isi : Berisi nomor, idktp,nik, nama, ttl, jkl, alamat, rt, kel, kec, agama, status, pekerjaan, kewarganegaraan, berlaku

Foreign Key : nomor

Volume : 15 fields

Detail Description : Detail tabel id dapat dilihat pada tabel

Tabel 3. 9 E-KTP

ID Field	Deskripsi	Tipe & Length	Boleh null	Default	Keterangan
Nomor	Menyimpan jumlah e-ktp yang telah di tangkap oleh <i>camera</i>	Int (12)	No	None	Auto Increment, Foreign Key
Idktp	Menyimpan id e-KTP	Varchar (25)	NO	None	
NIK	Menyimpan nomor induk keluarga dari user	Varchar (25)	Yes	None	

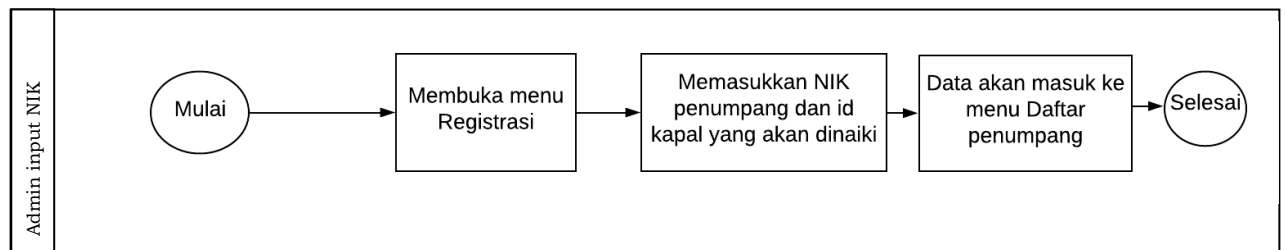
ID Field	Deskripsi	Tipe & Length	Boleh Null	Default	Keterangan
Nama	Menyimpan data atribut nama dari <i>user</i>	Varchar (25)	Yes	None	
Ttl	Menyimpan data atribut tempat tanggal lahir	Varchar (25)	Yes	None	
Jkl	Menyimpan data atribut jenis kelamin dari <i>user</i>	Varchar (9)	Yes	None	
Alamat	Menyimpan data atribut alamat dari <i>user</i>	Varchar (25)	Yes	None	
RT	Menyimpan data atribut alamat RT dari <i>user</i>	Varchar (25)	Yes	None	
Kel	Menyimpan data atribut alamat kelurahan <i>user</i>	Varchar (25)	Yes	None	
Kec	Menyimpan data atribut alamat kecamatan dari <i>user</i>	Varchar (25)	Yes	None	
Agama	Menyimpan data atribut agama dari <i>user</i>	Varchar (25)	Yes	None	
Status	Menyimpan data atribut status dari <i>user</i>	Varchar (25)	Yes	None	
Pekerjaan	Menyimpan data atribut pekerjaan dari <i>user</i>	Varchar (25)	Yes	None	
Kewarganegaraan	Menyimpan data atribut kewarganegaraan dari <i>user</i> atau pemilik <i>e-KTP</i>	Varchar (25)	Yes	None	

ID Field	Deskripsi	Tipe & Length	Boleh Null	Default	Keterangan
Berlaku	Menyimpan data atribut waktu berlaku e-KTP dari <i>user</i>	Varchar (25)	Yes	None	

3.3 Business Process Modelling Notation (BPMN)

3.3.1 BPMN administrator melakukan registrasi penumpang

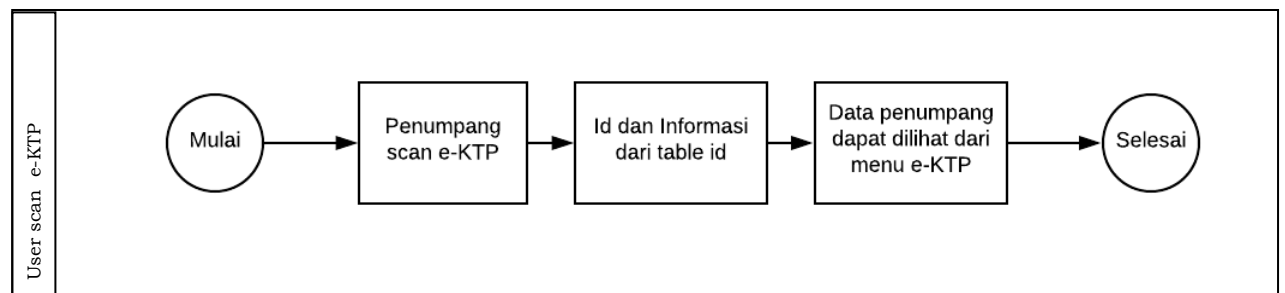
Pada sistem ini, administrator *web* akan mengakses *web e-KTP scanner* dan membuka halaman registrasi. Kemudian Administrator akan memasukkan NIK dan juga id kapal yang akan dinaiki oleh penumpang.



Gambar 3. 14 BPMN Registrasi Penumpang

3.3.2 BPMN penumpang melakukan scan e-KTP

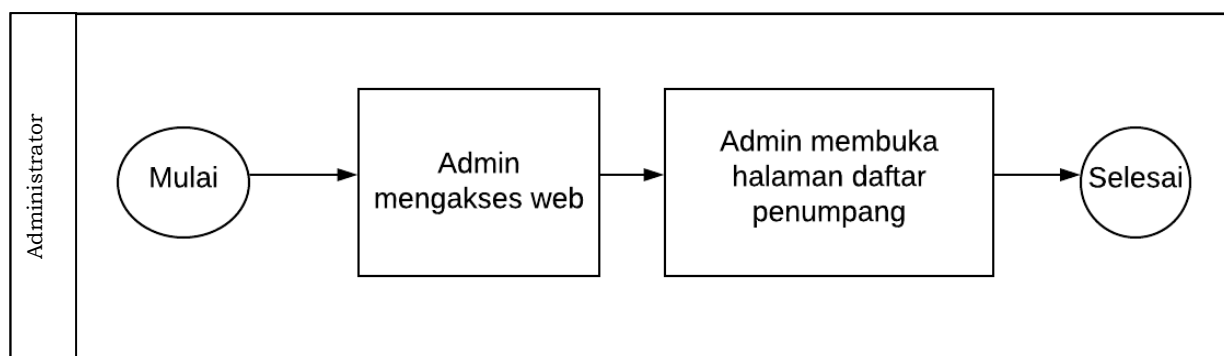
Penumpang atau *user* akan melakukan *scan e-KTP* terhadap perangkat yang telah dibangun. Pada awalnya, id dari chip e-KTP akan terdaftar di table id. Sedangkan nama, nik, alamat, dan informasi lainnya dari fisik e-KTP akan masuk ke dalam table ktp dalam *database*. Pada menu e-KTP user maupun administrator dapat melihat data dari setiap penumpang yang berhasil melakukan *scan e-KTP*. Data yang ditampilkan tersebut merupakan hasil *join table* id dan ktp berdasarkan nomor yang sama.



Gambar 3. 15 BPMN Scan e-KTP

3.3.3 BPMN administrator melihat daftar penumpang

Administrator dapat melihat seluruh penumpang kapal baik kapal id satu, dua, maupun tiga. Data tersebut diperoleh berdasarkan tabel registrasi. Informasi NIK penumpang dan id kapal yang ditumpangi dari tabel registrasi akan masuk ke tabel daftar penumpang. Dengan menggunakan NIK dari tabel ini, maka nama penumpang dari tabel ktp pun bisa ditampilkan pada halaman daftar penumpang.



Gambar 3. 16 BPMN Admin Melihat Daftar Penumpang

BAB IV IMPLEMENTASI DAN PENGUJIAN

Pada bab ini dijelaskan mengenai implementasi hardware dan software untuk membangun sistem. Tahap implementasi yang dilakukan berupa proses instalasi software, konfigurasi dan pengujian terhadap software dan hardware tersebut.

4.1 Implementasi

Pada sub bab ini dijelaskan mengenai implementasi yang dilakukan untuk membangun E-KTP *Scanner using Raspberry* sehingga semua *hardware* dapat berjalan dengan baik.

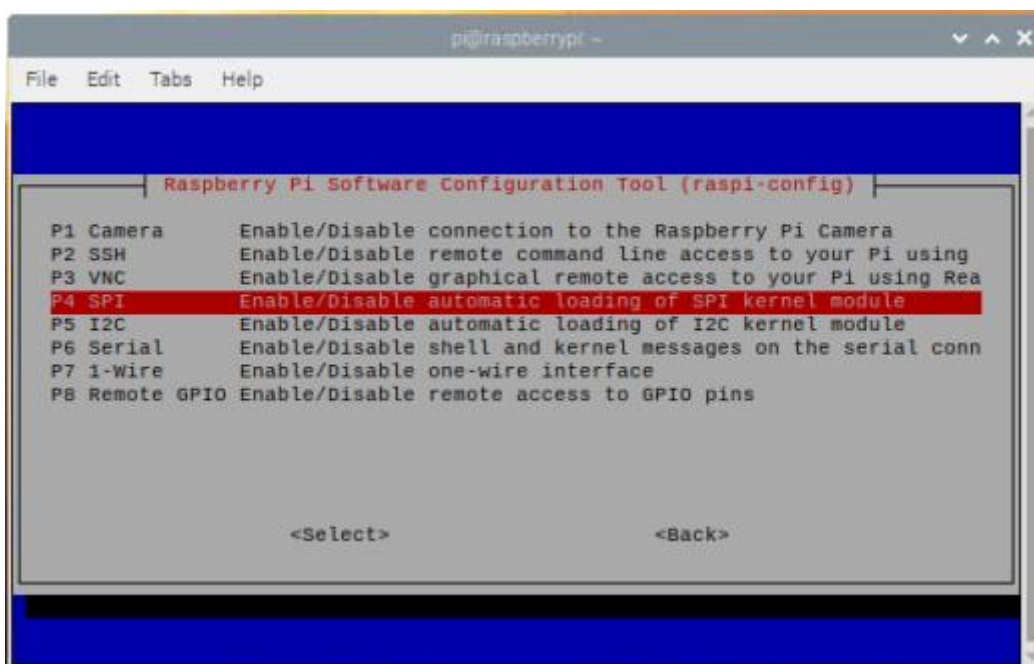
4.1.1 Implementasi Raspberry dengan RFID RC522

RFID RC522 merupakan alat yang digunakan untuk membaca dan mengekstrak informasi dari *chip* yang berada di dalam e-KTP. Pemasangan pin *Raspberry Pi* dengan *RFID RC522* dijelaskan pada tabel berikut.

Tabel 4. 1 Pin RFID Reader

Nama	Pin <i>Name</i>
SDA	24
SCK	23
MOSI	19
MISO	21
GND	6
RST	22
3.3V	1

Untuk menghubungkan *RFID reader* dengan raspberry, maka dilakukan pengaktifan (*enable*) port SPI terlebih dahulu melalui *Raspberry Pi Software Configuration Tool*.



Gambar 4. 1 Enable SPI

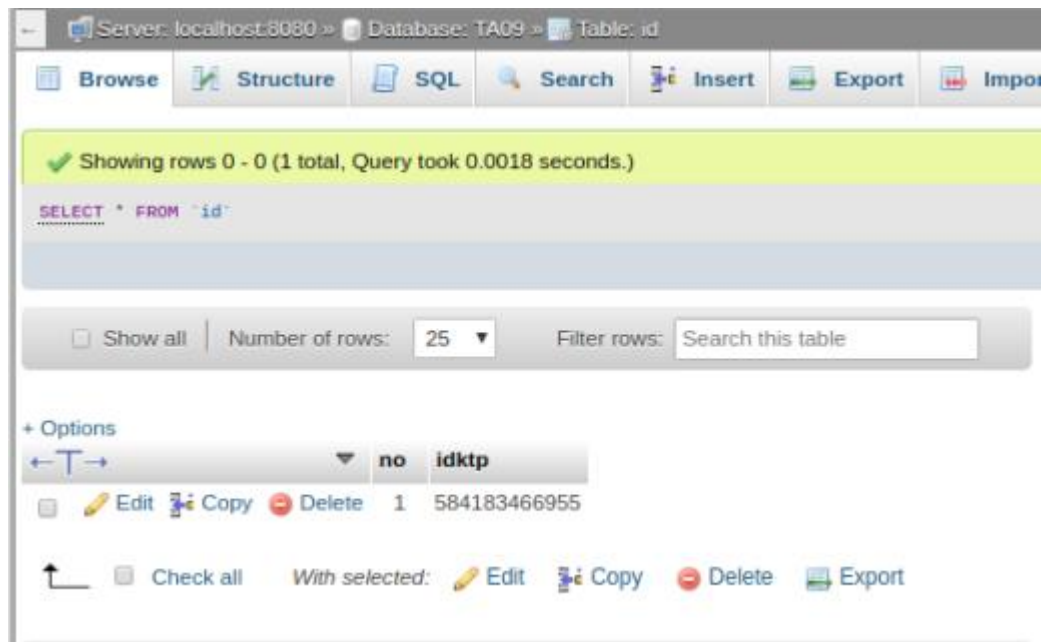
Rangkaian RFID RC522 dengan Raspberry Pi dihubungkan menggunakan kabel *jumper* dapat dilihat pada gambar berikut.



Gambar 4. 2 Proses Membaca id

Dalam RFID RC522 ada 8 koneksi di dalamnya yaitu SDA (Serial Data Signal), SCK (Serial Clock), MOSI (Master Out Slave In), MISO (Master In Slave Out), IRQ (Interrupt Request), GND (Ground Power), RST (Reset-Circuit) dan 3.3v (3.3v Power In). Namun, IRQ tidak perlu dihubungkan ke GPIO Raspberry Pi.

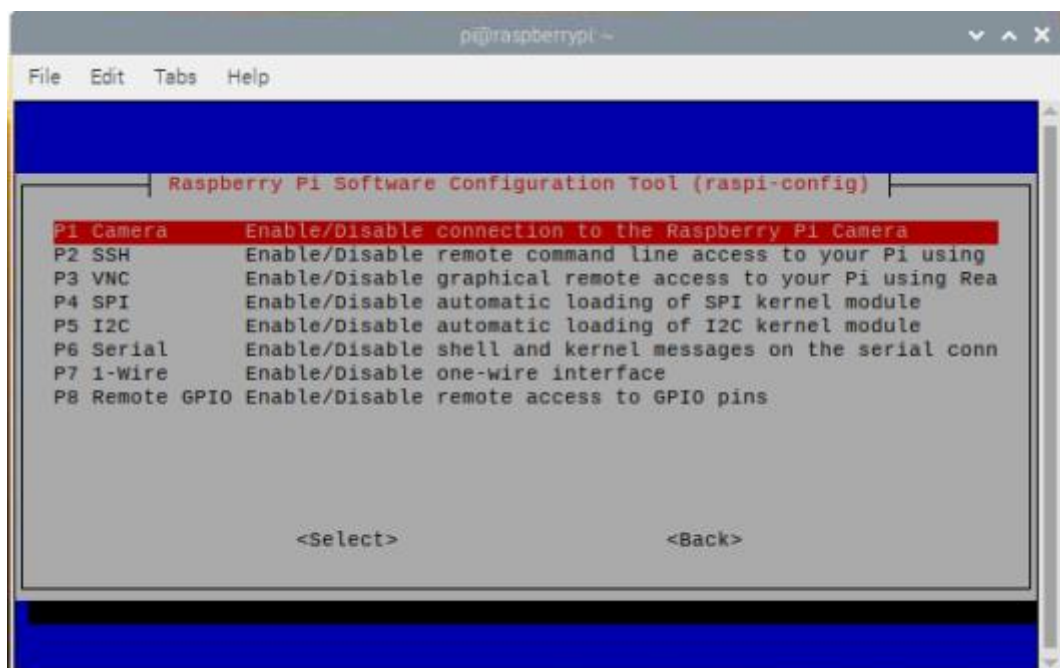
Script code untuk mengambil id e-KTP dan memasukkannya kedalam database dituliskan didalam `readktp.py`. Ketika e-KTP didekatkan dengan *reader* tersebut, maka dalam jarak 2 cm *reader* mampu membaca id dan selanjutnya akan di insert ke dalam database tepatnya pada tabel id.



Gambar 4. 3 Tabel id

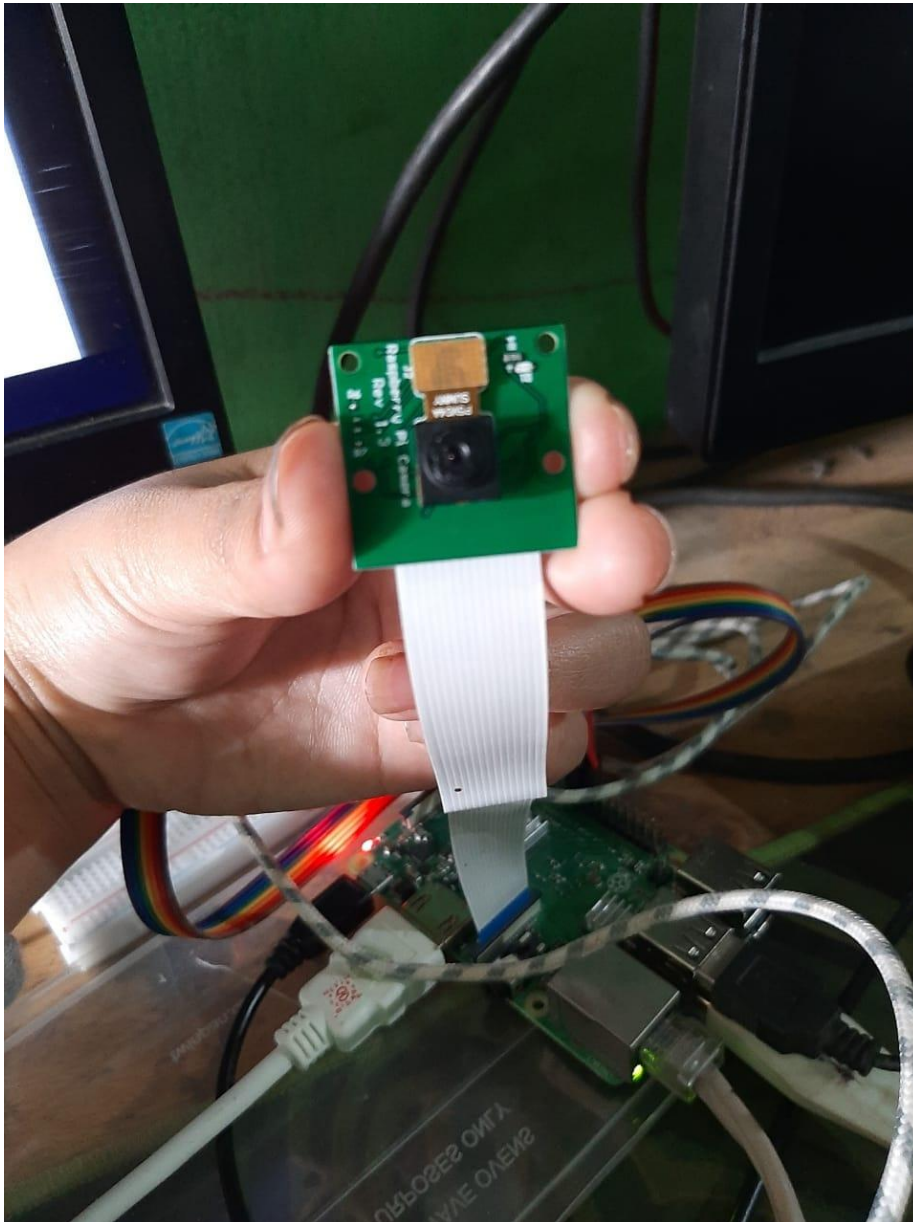
4.1.2 Implementasi *Raspberry Pi* dengan Camera

Camera digunakan untuk mengambil gambar dari bagian depan e-KTP. Untuk menghubungkan kamera *reader* dengan raspberry, maka dilakukan pengaktifan (*enable*) kamera terlebih dahulu melalui *Raspberry Pi Software Configuration Tool*.



Gambar 4. 4 Enable Camera

Rangkaian Raspberry pi dengan *camera* dapat dilihat pada gambar. Kabel kamera dimasukkan ke dalam *camera connector* yang berfungsi sebagai penghubung antara kamera modul dengan Raspberry.



Gambar 4. 5 Camera pada Raspberry

4.1.3 Implementasi Aplikasi

Pada sub bab ini akan dijelaskan implementasi yang dilakukan dalam pembangunan system. Pada menu e-KTP akan ditampilkan data seperti berikut. Apabila e-KTP yang telah berhasil di scan, maka datanya akan ditampilkan seperti berikut. Untuk menampilkan data tersebut, maka dibutuhkan 2 tabel yang saling berelasi, yaitu tabel id dan tabel ktp. Keduanya memiliki atribut yang bersifat *AUTO INCREMENT*, artinya index pada kedua kolom akan bertambah 1 setiap melakukan scan untuk e-KTP. Apabila nilai dari kolom yang bersifat *AUTO INCREMENT* tersebut adalah sama, maka datanya akan digabung dalam 1 baris seperti gambar dibawah ini.

Home / E-KTP

E-KTP

Create Id

Showing 1-1 of 1 item.










No	Idktp	Nik	Nama	Tempat & Tanggal Lahir	Jenis Kelamin	Alamat	Rt	Kel	Kecamatan	Agama	Status	Pekerjaan	Kewarganegaraan	Berlaku	
1	584183466955	121202420200002	TALENTA MESIANNA SITORUS	UJUNG TANDUK 02-02- 0200	PEREMPUAN	UJUNG TANDUK	000/000	UJUNG TANDUK	LAGUBOTI	KRISTEN	BELUM KAWIN	PELAJAR/MAHASISWA	WNI	SEUMUR HIDUP	 

Gambar 4. 6 Tampilan Halaman e-KTP

Kapals

Create Kapal

Showing 1-3 of 3 items.

#	Id Kapal	Namakapal	Waktu	
	<input type="text"/>	<input type="text"/>	<input type="text"/>	
1	1	Ihan Batak	2020-05-28 10:00:00	  
2	2	Ferry	2020-04-28 15:00:00	  
3	3	Dengke	2020-05-30 12:00:00	  

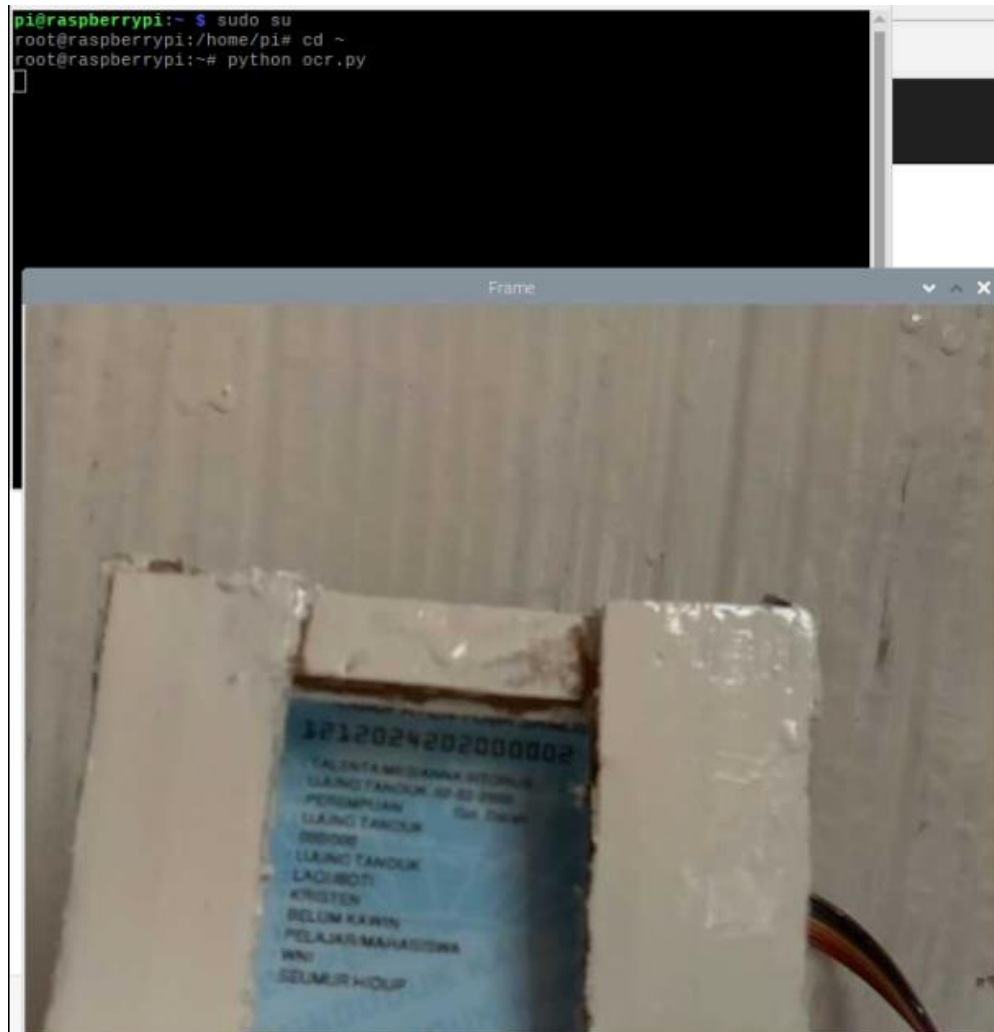
Gambar 4. 7 Tampilan Halaman Daftar Kapal

4.2 Pengujian

Pada sub bab ini dijelaskan bagaimana uji cara kerja kamera pada raspberry Pi. Ketika program ocr.py dijalankan pada terminal, maka akan tampil frame yang menunjukkan hasil dari tangkapan kamera. Hasil dari tangkapan kamera tersebut ialah bagian depan e-KTP yang mengandung nama, alamat, hingga masa berlaku e-KTP. Setelah gambar tersebut berhasil diambil oleh kamera, maka gambar tersebut akan langsung diproses oleh tesseract OCR untuk dikonversi menjadi bentuk teks. Proses tesseract OCR ini melalui beberapa tahap seperti yang telah dijalankan pada bab II.

4.2.1 Pengujian Terhadap Kamera

Untuk melihat apakah kamera sudah bekerja dapat di uji langsung dengan menjalankan file ocr.py. Kamera akan mengambil gambar e-KTP yang berada pada jarak 10 cm. Ketika kamera sudah bekerja, maka pada monitor akan muncul secara otomatis sebuah *frame* yang menampilkan gambar yang sedang diambil oleh *camera* modul Raspberry.



Gambar 4. 8 Proses Pengambilan Gambar

Hasil dari konversi gambar menjadi teks akan ditampilkan pada terminal seperti berikut. Untuk memasukkan data tersebut kedalam database, maka terlebih dahulu serangkaian teks tersebut di masukkan kedalam array. Developer juga memanfaatkan fungsi *splitlines()* untuk memisahkan data tersebut kedalam *index array* yang berbeda. Dengan memasukkan data tersebut kedalam array, maka insert data ke dalam *column* bisa dijalankan.

```
pi@raspberrypi:~ $ sudo su
root@raspberrypi:/home/pi# cd ~
root@raspberrypi:~# python ocr.py
1212024202000002

TALENTA MESIANNA SITORUS
UJUNG TANDUK 02-02-2000
PEREMPUAN      Gol Darah :-
UJUNG TANDUK
0000/000
UJUNG TANDUK
LAGUBOTI
KRISTEN
BELUM KAWIN
PELAJAR
MAHASISWA
WNI
SEUMUR HIDUP
root@raspberrypi:~#
```

Gambar 4. 9 Hasil Konversi Gambar

Hasil penyatuan data baik dari RFID *reader* dan Tesseract OCR yang mampu meng ekstrak informasi berupa gambar menjadi teks, akan masuk ke dalam *database* dan ditampilkan dalam *web* seperti berikut.

Home / E-KTP

E-KTP

Create Id

Showing 1-1 of 1 item.

No	Idktp	Nik	Nama	Tempat & Tanggal Lahir	Jenis Kelamin	Alamat	Rt	Kel	Kecamatan	Agama	Status	Pekerjaan	Kewarganegaraan	Berlaku
1	584183466955	121202420200002	TALENTA MESIANNA SITORUS	UJUNG TANDUK 02-02- 0200	PEREMPUAN	UJUNG TANDUK	000/000	UJUNG TANDUK	LAGUBOTI	KRISTEN	BELUM KAWIN	PELAJAR/MAHASISWA	WNI	SEUMUR HIDUP

Gambar 4. 10 Data E-KTP

4.2.2 Pengujian Terhadap RFID

Untuk menguji tingkat keberhasilan RFID dalam membaca ID e-KTP maka untuk setiap proses melakukan atau mengambil ID dilakukan dengan posisi dan jarak yang berbeda. Dalam uji coba ini, digunakan 5 kali percobaan untuk menentukan tingkat keberhasilan RFID dalam membaca ID e-KTP.

Tabel 4. 2 Pengujian RFID

No.	Posisi	Keterangan
1.	E-KTP berada 3 cm dari RFID	ID berhasil dibaca
2.	E-KTP berada 4 cm dari RFID	ID berhasil dibaca namun menunggu beberapa detik dahulu
3.	E-KTP berada 4 cm dari RFID	ID gagal dibaca
4.	RFID membaca tampak depan E-KTP	ID berhasil dibaca
5.	RFID membaca tampak belakang E-KTP	ID berhasil dibaca

Berdasarkan uji coba tersebut, maka sistem yang dibangun pun mengikuti standar keberhasilan RFID *reader* dalam membaca ID e-KTP. Ketika e-KTP berada dalam jarak 3 cm dari RFID *reader* dan RFID *reader* membaca tampak belakang E-KTP, maka ID e-KTP tersebut berhasil dibaca dan ditampilkan pada terminal.

```
root@raspberrypi:~# python readktp.py
Scan e-KTP
AUTH ERROR!!
AUTH ERROR(status2reg & 0x08) != 0
584183466955
```

Gambar 4. 11 ID Berhasil Dibaca

Untuk melihat keseluruhan data penumpang, dapat dilihat pada menu daftar penumpang yang menampilkan NIK penumpang dan informasi kapal yang dinaiki berdasarkan data yang dimasukkan oleh administrator, selain itu pada daftar penumpang akan tampil nama penumpang yang diambil dari relasi antara tabel ktp berdasarkan NIK yang serupa.

BAB V. KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan dan saran mengenai Tugas Akhir yang sudah dilakukan dan saran-saran yang akan disampaikan untuk masa yang akan datang bagi pengembang dan juga pembaca

5.1 Kesimpulan

Setelah melakukan pengerjaan pada E-KTP Scanner using Raspberry dapat diperoleh kesimpulan yaitu:

1. Dengan adanya E-KTP *Scanner using* Raspberry mempermudah pengambilan informasi dari e-KTP dan *chip* e-KTP. Perangkat ini bisa di hubungkan terhadap aplikasi lain yang membutuhkan inputan data berupa informasi dari e-KTP.
2. E-KTP *Scanner using* Raspberry adalah sistem pengambilan informasi dari e-KTP dengan data yang tidak bisa dimanipulasi dan ditiru.

5.2 Saran

Saran yang dapat diberikan adalah sebagai berikut:

1. Sistem ini dikembangkan dengan memperhatikan *service time* dari masing-masing *user*, apakah semakin cepat atau semakin lama
2. Sistem ini masih dikembangkan di satu perangkat dan contohnya adalah pelabuhan kapal, datanya masih disimpan di raspberry pi. Perangkat ini bisa dihubungkan dengan aplikasi lain yang membutuhkan inputan data berupa informasi yang ada di e-KTP

Daftar Pustaka

- [1] Christiawan, Fredy., Setyawan, Raden Arief., Siwindarto., Ponco., Pemanfaatan RFID sebagai pemeriksa jumlah ban di gudang penyimpanan berbasis arduino dengan sms sebagai media transmisi data, 2013.
- [2] Galih, Sandika., Erik., Lukman, Muhammad., *Penerapan teknik ocr (optical character recognition) pada aplikasi terjemahan kitab fiqih safinah an-naja menggunakan readiris*, 2014.
- [3] Hartanto, Suryo., dkk., *Optical character recognition menggunakan algoritma template matching correlation*, 2015.
- [4] Mirah, Sekar., *Pengenalan NIK pada e-KTP menggunakan segmentasi profil proyeksi dan ekstraksi ciri menggunakan invarian momen dan intensity of character*, 2018.
- [5] Aska, Febri Zahro., Satria Deny, Kasoep Werman, download pada Mei 2020, *Implementasi radio frequency identification (RFID) sebagai otomasi pada smartphone*. 2006.
- [6] Pancawisma, Mahmuda., *Penerapan program e-ktp dalam rangka tertib administrasi kependudukan*, Volume 5, 2016.
- [7] Patel, Chirag., *Optical character recognition by open source ocr tool tesseract: a case study*, 2012.
- [8] Putera, Roni Ekha., Valentina, Tengku Rika, download pada September 2019, *Implementasi program ktp elektronik di daerah percontohan*. 2011.
- [9] Puasandi, Tadu., *Sistem akses kontrol kunci elektrik menggunakan pembacaan e-ktp*, 2014.
- [10] Sinaga, Tantri Hidayati., *Perancangan aplikasi pencatatan data manifes penumpang kapal penyebrangan danau toba menggunakan web*, 2020.
- [11] Tamatjita, Nurmiyati., Mahastama, Wikan Aditya, download pada Oktober 2019, *Optical character recognition dengan algoritma ocrchie*, 2014.
- [12] Widodo, Saroni., Sasmita, Gahszan., Hasan, Abu, *Sistem akses pintu menggunakan e-KTP sebagai kunci elektronik berbasis near field communication dimonitor melalui jaringan komputer*, 2016.

BAB VI LAMPIRAN

Pada sub bab ini dijelaskan instalasi yang dilakukan untuk membangun E-KTP Scanner using Raspberry. Instalasi yang dilakukan yaitu instalasi sistem operasi raspbian dan database application MYSQL server yang digunakan untuk penyimpanan data user. Dalam E-KTP Scanner using Raspberry juga dilakukan instalasi dan konfigurasi kamera dan juga RFID reader.

6.1 Instalasi Sistem Operasi Raspbian

Berikut adalah langkah-langkah instalasi sistem operasi *Raspbian* pada Raspberry Pi 3 B.

1. Download sistem operasi raspbian di <https://www.raspberrypi.org/downloads/raspbian/>
2. Download SDFormatter di <https://www.sdcard.org/downloads/formatter/index.html> yang digunakan untuk format SD Card Raspberry Pi dan Win32DiskImager di <http://lauchpad.net/win-32-image-writer/> yang digunakan untuk write sistem operasi Raspbian ke dalam SD card. Setelah selesai, masukkan SD card ke SD card reader.
3. Buka SDFormatter dan hapus volume label kemudian pilih tombol option, pilih format size adjustment “on”.
4. Klik “OK” dan “format”
5. Buka Win32DiskImager, browser installer Raspbian Jessie dan pilih write
6. Tunggu sampai muncul notifikasi “write successful”
7. Eject SD card dan pindahkan ke Raspberry Pi
8. Lakukan *rebooting* pada raspberry pi

6.2 Install Tesseract

Langkah-langkah untuk melakukan instalasi Tesseract pada Raspberry adalah sebagai berikut:

Install packages di sistem dengan *command*:

```
sudo apt-get install libqtgui4 libqt4-test
```

Selanjutnya install Tesseract dengan *command* berikut:

```
cd ~  
git clone https://github.com/thortex/rpi3-tesseract  
cd rpi3-tesseract/release  
./install_requires_related2leptonica.sh  
./install_requires_related2tesseract.sh  
./install_tesseract.sh
```

Selanjutnya pindahkan ke direktori yang tepat dengan *command* berikut ini:

```
cd ~  
wget https://github.com/tesseract-ocr/tessdata/raw/master/eng.traineddata  
sudo mv -v eng.traineddata /usr/local/share/tessdata/
```

Setelah itu, buatlah Python virtual dengan *command* berikut:

```
mkvirtualenv cv_tesseract -p python3
```

Lalu *install packages* dengan *command*:

```
workon cv_tesseract  
pip install opencv-contrib-python imutils pytesseract  
pillow
```

6.3 Konfigurasi RFID pada Raspberry

Install library MFRC522 pada Raspberry Pi menggunakan *command* berikut :

```
sudo pip3 install mfrc522
```


Buatlah direktori tempat penyimpanan script untuk membaca id dari e-KTP. Dalam sistem ini, *developer* membuat direktori id.

```
mkdir pi
```

Untuk membaca id yang tersimpan dalam e-KTP, *script code* disimpan pada `readktp.py`

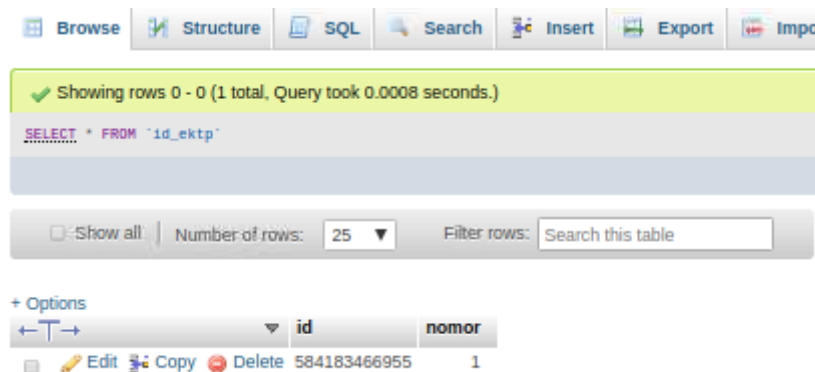
```
root@raspberrypi:~# cd id
root@raspberrypi:~/id# ls
readktp.py readtag.py
root@raspberrypi:~/id# python readktp.py
Scan e-KTP
AUTH ERROR!!
AUTH ERROR(status2reg & 0x08) != 0
584183466955
```

Id yang tersimpan di dalam e-ktp ialah 584183466955, pada hasil tersebut terjadi autentikasi error sebab ada informasi di dalam chip e-ktp yang telah dienkripsi dan tidak dapat diakses oleh sembarang orang. Namun untuk sistem ini, *developer* hanya membutuhkan id tersebut.

Setelah e-ktp di scan dan memperoleh id pada terminal, maka id tersebut ditambahkan kedalam *database*, seperti pada script dibawah

```
print ("Scan e-KTP")
try:
    id, text = reader.read()
    print(id)
    print(text)
    sql = "INSERT INTO id_ektp(id) VALUES ('%s')" % (id)
    cursor.execute(sql)
    db.commit()
```

Untuk melakukan pengujian apakah id e-KTP yang telah di scan sudah tersimpan atau tidak, bisa dilihat berdasarkan isi tabel `id_ektp`



6.4 Konfigurasi *Camera* modul pada Raspberry:

Untuk menghubungkan kamera modul dengan raspberry terlebih dahulu cari port kamera dan menghubungkan kamera dengan port yang tersedia pada raspberry. Kemudian lakukan *start up* pi, buka Configuration Tool Raspberry Pi dari menu utama atau boleh dengan perintah `sudo raspi config` lalu, aktifkan kamera *software*. Selanjutnya, *reboot* raspberry.

6.5 Kode Program E-KTP Scanner using Raspberry

1. Kode program RFID

```
#!/usr/bin/env python
import time
import RPi.GPIO as GPIO
from mfrc522 import SimpleMFRC522
import mysql.connector

db = mysql.connector.connect(
    host="127.0.0.1",
    user="root",
    passwd="kelompok9",
    database="TA09"
)

cursor = db.cursor()
reader = SimpleMFRC522()
print ("Scan e-KTP")
try:
    id, text = reader.read()
    print(id)
    print(text)
    sql = "INSERT INTO id(idktp) VALUES ('%s')" % (id)
    cursor.execute(sql)
    db.commit()
finally:
    GPIO.cleanup()
```

2. Kode program kamera modul

```
import cv2

import pytesseract

from picamera.array import PiRGBArray

from picamera import PiCamera

import mysql.connector


camera = PiCamera()

camera.resolution = (640, 480)

camera.framerate = 30


rawCapture = PiRGBArray(camera, size=(640, 480))


for frame in camera.capture_continuous(rawCapture,
format="bgr", use_video_port=True):

    image = frame.array

    cv2.imshow("Frame", image)

    key = cv2.waitKey(1) & 0xFF


    rawCapture.truncate(0)
```

```

        if key == ord("s"):

            text = pytesseract.image_to_string(image)

            print(text)

            cv2.imshow("Frame", image)

            cv2.waitKey(0)

            break

cv2.destroyAllWindows()

db = mysql.connector.connect(

    host="127.0.0.1",

    user="root",

    passwd="kelompok9",

    database="TA09"

)

cursor = db.cursor()

try:

    data = text.splitlines()

    ktp1 = data[0]

    ktp2 = data[1]

    ktp3 = data[2]

    ktp4 = data[3]

    ktp5 = data[4]

    ktp6 = data[5]

```

```

ktp7 = data[6]

ktp8 = data[7]

ktp9 = data[8]

ktp10 = data[9]

ktp11 = data[10]

ktp12 = data[11]

ktp13 = data[12]

sql = "INSERT INTO
ktp(nik,nama,ttl,jkl,alamat,rt,kel,kec,agama,status,pekerjaan,kewarganegaraan,berlaku) VALUES ('%s','%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s')" % (ktp1, ktp2, ktp3, ktp4, ktp5, ktp6, ktp7, ktp8, ktp9, ktp10, ktp11, ktp12, ktp13)

cursor.executemany(sql)

db.commit()

finally:

GPIO.cleanup()

```

3. Kode program untuk halaman E-KTP

```
<?php

use yii\helpers\Html;

use yii\grid\GridView;

/* @var $this yii\web\View */

/* @var $searchModel app\models\IdSearch */

/* @var $dataProvider yii\data\ActiveDataProvider */

$this->title = 'E-KTP';

$this->params['breadcrumbs'][] = $this->title;

?>

<div class="id-index">

    <h1><?= Html::encode($this->title) ?></h1>

    <?php // echo $this->render('_search', ['model' =>
$searchModel]); ?>

    <p>

        <?= Html::a('Create Id', ['create'], ['class' =>
'btn btn-success']) ?>

    </p>
```

```

<?= GridView::widget([
    'dataProvider' => $dataProvider,
    'filterModel' => $searchModel,
    'columns' => [
        // ['class' => 'yii\grid\SerialColumn'],

        'no',

        'idktp',

        [
            'attribute' => 'NIK',
            'value' => 'ktp.nik',
        ],

        [
            'attribute' => 'Nama',
            'value' => 'ktp.nama',
        ],

        [
            'attribute' => 'Tempat & Tanggal Lahir',
            'value' => 'ktp.ttl',
        ],
    ],

```

```

[
    'attribute' => 'Jenis Kelamin',
    'value' => 'ktp.jkl',
],
[
    'attribute' => 'Alamat',
    'value' => 'ktp.alamat',
],
    [
        'attribute' => 'RT',
        'value' => 'ktp.rt',
    ],
    [
        'attribute' => 'Kel',
        'value' => 'ktp.kel',
    ],
    [
        'attribute' => 'Kecamatan',
        'value' => 'ktp.kec',
    ],
    [
        'attribute' => 'Agama',
        'value' => 'ktp.agama',
    ]
]

```



```

],

[
    'attribute' => 'Status',
    'value' => 'ktp.status',
],

[
    'attribute' => 'Pekerjaan',
    'value' => 'ktp.pekerjaan',
],

[
    'attribute' => 'Kewarganegaraan',
    'value' => 'ktp.kewarganegaraan',
],

[
    'attribute' => 'berlaku',
    'value' => 'ktp.berlaku',
],

['class' => 'yii\grid\ActionColumn'],

],

]); ?>

</div>

```

4. Kode program untuk id.php

```
<?php

namespace app\models;

use Yii;

/**
 * This is the model class for table "id".
 *
 * @property int $no
 * @property string $idktp
 */
class Id extends \yii\db\ActiveRecord
{
    /**
     * @inheritdoc
     */
    public static function tableName()
    {
        return 'id';
    }

    /**
     * @inheritdoc
     */
}
```

```

public function rules()

{

    return [

        [['idktp'], 'required'],

        [['idktp'], 'string', 'max' => 25],

        [['idktp'], 'unique'],

    ];

}


/**
 * @inheritdoc
 */

public function attributeLabels()

{

    return [

        'idktp' => 'Idktp',

        'no' => 'No',

    ];

}

public function getKtp(){

    return $this->hasOne(Ktp::className(), ['nomor'
=>'no']);

}

```