

**RANCANG BANGUN ALAT DETEKSI NOMINAL  
MATA UANG RUPIAH DENGAN PREDIKSI  
*MACHINE LEARNING* UNTUK TUNANETRA**



**Disusun Oleh :**

<b>Bagja Bagus Adhitama</b>	<b>(3.34.19.2.04)</b>
<b>Gaudentio Shandie Mikolus</b>	<b>(3.34.19.2.11)</b>

**PROGRAM STUDI D-III TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI SEMARANG  
2022**

**RANCANG BANGUN ALAT DETEKSI NOMINAL  
MATA UANG RUPIAH DENGAN PREDIKSI  
*MACHINE LEARNING* UNTUK TUNANETRA**



Tugas Akhir ini disusun untuk melengkapi sebagian persyaratan  
menjadi Ahli Madya

**Disusun Oleh :**

**Bagja Bagus Adhitama (3.34.19.2.04)**

**Gaudentio Shandie Mikolus (3.34.19.2.11)**

**PROGRAM STUDI D-III TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI SEMARANG**

**2022**

## PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini, saya :

Nama/NIM/Kelas : Bagja Bagus Adhitama/3.34.19.2.04/IK-3C  
Nama/NIM/Kelas : Gaudentio Shandie Mikolus /3.34.19.2.11/IK-3C  
Jurusan : Teknik Elektro  
Program Studi : Diploma III Teknik Informatika

Kami menyatakan dengan sesungguhnya bahwa Tugas Akhir dengan judul **“Rancang Bangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi *Machine Learning* untuk Tunanetra”**, yang dibuat untuk melengkapi sebagian persyaratan menjadi Ahli Madya pada Program Studi Diploma III Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Semarang, sejauh yang kami ketahui bukan merupakan tiruan atau duplikasi dari tugas akhir yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar Ahli Madya di lingkungan Politeknik Negeri Semarang maupun di perguruan tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Semarang, 24 Agustus 2022

**Bagja Bagus Adhitama**  
NIM 3.34.19.2.04

**Gaudentio Shandie Mikolus**  
NIM 3.34.19.2.11

## HALAMAN PERSETUJUAN

Tugas Akhir dengan judul “**Rancang Bangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi *Machine Learning* untuk Tunanetra**” disusun untuk melengkapi sebagian persyaratan menjadi Ahli Madya pada Program Studi Diploma III Teknik Informatika, Jurusan Teknik Elektro Politeknik Negeri Semarang dan disetujui untuk diajukan dalam sidang ujian Tugas Akhir.

Semarang, 24 Agustus 2022

Pembimbing I

Pembimbing II

**Idhawati Hestiningsih, S.Kom., M.Kom.**  
NIP 197711192008012013

**Lilie Triyono, S.T., M.Kom.**  
NIP 198404202015041003

Mengetahui,

Ketua Program Studi Diploma III Teknik Informatika

**Idhawati Hestiningsih, S.Kom., M.Kom.**  
NIP 197711192008012013

## HALAMAN PENGESAHAN

Tugas Akhir dengan judul “**Rancang Bangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi *Machine Learning* untuk Tunanetra**”. Telah dipertahankan dalam ujian wawancara dan diterima sebagai syarat untuk menjadi Ahli Madya pada Program Studi Diploma III Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Semarang pada tanggal 25 Agustus 2022.

Tim Penguji

Penguji I

Penguji II

Penguji III

**Sukanto, S.T., M.Kom.**

NIP 197101172003121001

**Drs. Parsumo Rahardjo, M.Kom.**

NIP 196008221988031001

**Wiktasari, S.T., M.Kom.**

NIP 198703272019032012

Ketua,

Sekretaris,

**Idhawati Hestiningsih, S.Kom., M.Kom.**

NIP 197711192008012013

**Sirli Fahriah, S.Kom., M.Kom.**

NIP 199401272019032036

Mengesahkan,

Ketua Jurusan Teknik Elektro

**Yusnan Badruzzaman, S.T., M.Eng.**

NIP 197503132006041001

## KATA PENGANTAR

Puji dan Syukur kita panjatkan kepada Tuhan Yang Maha Esa karena dengan rahmat-Nya penulis dapat menyelesaikan Laporan Tugas Akhir ini dengan baik dan lancar tanpa kendala apapun.

Laporan Tugas Akhir dengan judul “**Rancang Bangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi *Machine Learning* untuk Tunanetra**” ini disusun dengan tujuan memenuhi salah satu syarat untuk lulus dan mendapatkan gelar Diploma III Teknik Elektro Program Studi Teknik Informatika di Politeknik Negeri Semarang.

Dalam menyusun Laporan Tugas Akhir ini, penulis menyadari bahwa tanpa bimbingan dan doa dari berbagai pihak, Laporan ini tidak dapat selesai dengan tepat waktu. Oleh karena itu, penulis mengucapkan terimakasih sebesar-besarnya kepada:

1. Allah SWT, atas rahmat dan karunia-Nya.
2. Kedua orang tua atas doa restunya dan selalu memberikan dukungan semangat, cinta, dan kasih sayang kepada penulis hingga terselesaikannya laporan ini.
3. Prof. Dr. Totok Prasetyo, B,Eng (Hons), MT, IPU, ACPE. selaku Direktur Politeknik Negeri Semarang.
4. Yusnan Badruzzaman, S.T., M.Eng. selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Semarang.
5. Idhawati Hestningsih, S.Kom., M.Kom. selaku Kepala Program Studi Teknik Informatika dan Dosen Pembimbing I.
6. Liliek Triyono, S.T., M.Kom. selaku Dosen Pembimbing II.

Akhir kata, penulis menyadari bahwa masih terdapat kekurangan dan kesalahan dalam penulisan laporan magang ini. Oleh karena itu, kritik dan saran dari pembaca sangat diharapkan bagi penulis. Atas perhatiannya, penulis mengucapkan terimakasih sebesar-besarnya dan semoga Tugas Akhir ini bermanfaat.

Semarang, 24 Agustus 2022

**Bagja Bagus Adhitama**  
NIM 3.34.19.2.04

**Gaudentio Shandie Mikolus**  
NIM 3.34.19.2.11

## ABSTRAK

*Bagja Bagus Adhitama dan Gaudentio Shandie Mikolus “Rancang Bangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi Machine Learning untuk Tunanetra”, Tugas Akhir Diploma III Jurusan Teknik Elektro Politeknik Negeri Semarang, di bawah bimbingan Idhawati Hestningsih, S.Kom., M.Kom. dan Liliek Triyono, S.T., M.Kom., 24 Agustus 2022, 85 halaman.*

Pada saat ini, uang merupakan alat transaksi yang paling sering digunakan tak terkecuali bagi para penyandang tunanetra. Namun, karena keterbatasan mereka pada penglihatan membuat mereka kesulitan dalam mengetahui berapa nominal uang yang sedang digunakan jika hanya diraba. Permasalahan ini muncul karena ukuran fisik uang kertas antar nominal yang sangat mirip, sehingga sulit dibedakan jika hanya diraba dengan tangan. Untuk mengatasi permasalahan tersebut dibuatlah alat yang dapat membaca nominal suatu uang dengan prediksi *machine learning*. Dalam hal ini digunakan sensor warna TCS230 untuk mendeteksi nilai frekuensi warna, speaker untuk mengeluarkan suara hasil prediksi, algoritma *K-Nearest Neighbor* (KNN) untuk prediksi, dan board ESP32 yang tersambung koneksi jaringan untuk pengiriman dan pengambilan data dari *firebase*. Dengan bantuan prediksi *machine learning* yang akurat, alat ini diharapkan dapat mengatasi permasalahan yang dihadapi para penyandang tunanetra. Skor akurasi yang didapat untuk algoritma yang digunakan adalah 92 persen dengan hasil prediksi pada setiap sampel pengujian adalah 100 persen akurat. Dengan hasil pengujian tersebut, diyakini alat ini siap digunakan kepada user (tunanetra). Namun untuk penggunaan lebih lanjut, alat ini masih dalam tahap pengembangan agar dapat digunakan lebih efektif dan efisien. Karena kekurangan dari alat ini adalah pada tombol reset, untuk memulai ulang pemindaian frekuensi warna uang oleh sensor warna. Tombol tersebut dirasa masih sulit digunakan bagi para penyandang tunanetra karena ukurannya yang kecil. Diharapkan juga untuk pengembangan selanjutnya dapat ditambahkan sumber daya misal seperti baterai, agar dapat digunakan secara portable.

Kata kunci: *Mata Uang, Tunanetra, Alat Deteksi Uang, Keaslian Mata Uang, Mikrokontroler, Arduino, Machine Learning*



## ABSTRACT

*Bagja Bagus Adhitama and Gaudentio Shandie Mikolus " Design and Build a Rupiah Currency Nominal Detection Tool with Machine Learning Predictions for the Blind ", Diploma III Final Project of the Department of Electrical Engineering, Semarang State Polytechnic, under the guidance of Idhawati Hestiningsih, S.Kom., M.Kom. and Liliek Triyono, S.T., M.Kom., 24 August 2022, 85 pages.*

*At this time, money is the most frequently used transaction tool, including for people with visual impairments. However, due to their limited vision, it is difficult for them to know how much money is being used if they are only touched. This problem arises because the physical size of banknotes between denominations are very similar, so it is difficult to distinguish if only touched by hand. To overcome this problem, a tool is made that can read the nominal of a money with machine learning predictions. In this case, the TCS230 color sensor is used to detect the color frequency value, a speaker to output the predicted sound, the K-Nearest Neighbor (KNN) algorithm for prediction, and an ESP32 board connected to a network connection for sending and retrieving data from firebase. With the help of accurate machine learning predictions, this tool is expected to be able to overcome the problems faced by blind people. The accuracy score obtained for the algorithm used is 92 percent with the prediction results in each test sample being 100 percent accurate. With the results of these tests, it is believed that this tool is ready to be used by users (blind). However, for further use, this tool is still in the development stage so that it can be used more effectively and efficiently. Because the lack of this tool is on the reset button, to restart the scanning of the color frequency by the color sensor. The button is still difficult to use for people with visual impairments because of its small size. It is also hoped that further development can add resources, such as batteries, so that they can be used portable.*

*Keyword: Currency, Blind, Money Detector, Currency Authenticity, Microcontroller, Arduino, Machine Learning*

## DAFTAR ISI

HALAMAN JUDUL.....	i
PERNYATAAN KEASLIAN TUGAS AKHIR.....	ii
HALAMAN PERSETUJUAN.....	iii
PERNYATAAN KEASLIAN TUGAS AKHIR.....	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR .....	v
ABSTRAK .....	vii
ABSTRACT.....	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR .....	xi
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN .....	1
1.1    Latar Belakang.....	1
1.1.1    Rumusan Masalah.....	1
1.1.2    Manfaat .....	2
1.1.3    Tujuan.....	2
1.2    Batasan Masalah .....	2
1.3    Sistematika Penulisan .....	2
3.3.4    BAB I : PENDAHULUAN.....	3
3.3.4    BAB II : TINJAUAN PUSTAKA .....	3
3.3.4    BAB III : KEGIATAN PELAKSANAAN .....	3
3.3.4    BAB IV : ANALISIS DAN PEMBAHASAN .....	3
3.3.4    BAB V KESIMPULAN.....	3
3.3.4    DAFTAR PUSTAKA .....	3
3.3.4    LAMPIRAN.....	3

BAB II TINJAUAN PUSTAKA.....	4
2.1    Landasan Teori.....	4
2.1.1    Pengertian Mata Uang Rupiah .....	4
2.1.2    Sensor TCS230.....	5
2.1.3    DF Player Mini .....	7
2.1.4    Mikrokontroler.....	9
2.1.5    ESP32 .....	11
2.1.6 <i>Integrated Development Environment</i> (IDE) Arduino.....	12
2.1.7 <i>Python</i> .....	13
2.1.8 <i>Machine Learning</i> .....	14
2.1.9    Speaker.....	15
2.1.10 <i>Firebase</i> .....	16
BAB III KEGIATAN PELAKSANAAN .....	18
3.1    Metodologi .....	18
3.2    Analisis Kebutuhan.....	19
3.2.1    Perangkat Keras yang digunakan.....	19
3.2.2    Perangkat Lunak yang digunakan.....	20
3.3    Perancangan Sistem.....	20
3.3.1    Block Diagram.....	20
3.3.2 <i>Flowchart</i> .....	22
3.3.3 <i>Wiring</i> Diagram.....	23
3.3.4    Perancangan Pengujian <i>User</i> .....	25
BAB IV ANALISIS DAN PEMBAHASAN .....	27
4.1    Pengujian.....	27
4.1.1    Deteksi dan pengiriman nilai <i>RGB</i> .....	27
4.1.2    Pengujian Web <i>Python</i> .....	30
4.1.3    Pengujian <i>Firebase</i> .....	32
4.1.4    Pengujian <i>DFPlayer</i> dan <i>Speaker</i> .....	34
4.2    Pembahasan Sistem .....	34
BAB V KESIMPULAN .....	37
5.1    Kesimpulan.....	37

5.2	Saran.....	37
DAFTAR PUSTAKA .....		38
LAMPIRAN.....		40
Lampiran 1 : Dokumentasi .....		40
Lampiran 2 : Kode Program Arduino .....		41
Lampiran 3 : Kode Program <i>Machine Learning</i> .....		44
Lampiran 4 : Kode Program <i>Web</i> .....		45
Lampiran 5 : Dataset .....		49

## DAFTAR GAMBAR

Gambar 2.1 Mata Uang Rupiah .....	4
Gambar 2.2 Sensor TCS230.....	5
Gambar 2.3 Pin TCS230 .....	5
Gambar 2.4 Skema Kerja TCS230.....	6
Gambar 2.5 DF <i>Player</i> Mini .....	7
Gambar 2.6 <i>Wiring</i> DF <i>Player</i> Mini .....	9
Gambar 2.7 Mikrokontroler .....	9
Gambar 2.8 Diagram Blok dan Struktur Mikrokontroler.....	10
Gambar 2.9 ESP32 .....	11
Gambar 2.10 IDE Arduino.....	12
Gambar 2.11 Bahasa <i>Python</i> .....	13
Gambar 2.12 <i>Machine Learning</i> .....	15
Gambar 2.13 <i>Speaker</i> .....	15
Gambar 2.14 <i>Firebase</i> .....	16
Gambar 3.1 Tahapan Metode <i>Prototype</i> .....	18
Gambar 3.2 <i>Block Diagram System</i> .....	21
Gambar 3.3 <i>Flowchart</i> Alur Kerja <i>System</i> .....	22
Gambar 3.4 <i>Flowchart</i> algoritma KNN .....	23
Gambar 3.5 <i>Wiring</i> Keseluruhan Alat.....	23
Gambar 3.6 <i>Wiring</i> DF <i>Player</i> Mini dan <i>Speaker</i> .....	23
Gambar 3.7 <i>Wiring</i> TCS230 .....	24
Gambar 4.1 Proses <i>Scanning</i> Mata Uang Rupiah .....	30
Gambar 4.2 Pengiriman Nilai RGB ke <i>Firebase</i> .....	30
Gambar 4.3 Kode untuk Menyimpan Model .....	31
Gambar 4.4 Kode <i>Framework Web</i> .....	31
Gambar 4.5 Kode <i>output_lable()</i> .....	32
Gambar 4.6 Tampilan <i>Web</i> .....	32
Gambar 4.7 Tampilan Data <i>Sensor</i> pada <i>Firebase</i> .....	33
Gambar 4.8 Tampilan Data <i>Output</i> pada <i>Firebase</i> .....	33
Gambar 4.9 Tampilan Koneksi DF <i>Player</i> .....	34



## DAFTAR TABEL

Tabel 2.1 Output <i>Frequency Scaling</i> .....	6
Tabel 2.2 Tipe Fotodioda .....	7
Tabel 2.3 Fungsi Pin DF <i>Player</i> Mini.....	8
Tabel 2.4 Perbandingan Teknologi .....	16
Tabel 3.1 Koneksi antar Pin DF <i>Player</i> dan ESP32.....	24
Tabel 3.2 Koneksi antar Pin TCS230 dan ESP32 .....	25
Tabel 3.3 Daftar Pertanyaan.....	25
Tabel 3.4 Indikator Penilaian .....	26
Tabel 3.5 Indikator Kepuasan Pengguna .....	26
Tabel 4.1 Nilai RGB Mata Uang Rp. 1000.....	27
Tabel 4.2 Nilai RGB Mata Uang Rp. 2000.....	28
Tabel 4.3 Nilai RGB Mata Uang Rp. 5000.....	28
Tabel 4.4 Nilai RGB Mata Uang Rp. 10.000.....	28
Tabel 4.5 Nilai RGB Mata Uang Rp. 20.000.....	29
Tabel 4.6 Nilai RGB Mata Uang Rp. 50.000.....	29
Tabel 4.7 Nilai RGB Mata Uang Rp. 100.000.....	29
Tabel 4.8 Hasil Pengujian Speaker .....	34
Tabel 4.9 Perbandingan antar Algoritma dan Akurasi tiap Mata Uang.....	35
Tabel 4.10 Hasil Kuesioner Responden .....	35





## DAFTAR LAMPIRAN

Lampiran 1 : Dokumentasi.....	40
Lampiran 2 : Kode Program Arduino .....	41
Lampiran 3 : Kode Program <i>Machine Learning</i> .....	44
Lampiran 4 : Kode Program Web .....	45
Lampiran 5 : Dataset .....	49

# **BAB I PENDAHULUAN**

## **1.1 Latar Belakang**

Uang dalam ilmu ekonomi tradisional diartikan sebagai alat tukar yang dapat diterima secara umum. Alat tukar itu dapat berupa benda apapun yang dapat diterima oleh masyarakat dalam pertukaran barang dan jasa. Sedangkan dalam ilmu ekonomi modern uang diartikan sebagai sesuatu yang tersedia secara umum dan diterima sebagai alat pembayaran bagi pembelian barang dan jasa serta kekayaan berharga lainnya dan untuk pembayaran utang. Jadi uang merupakan suatu benda yang diterima secara umum oleh masyarakat untuk mengukur nilai, menukar, dan melakukan pembayaran atas pembelian barang dan jasa.

Di era modern ini uang merupakan alat yang penting bagi manusia untuk kehidupan tidak terlepas bagi mereka penyandang disabilitas salah satunya seperti tunanetra. Dengan ukuran uang kertas rupiah yang saat ini beredar, penyandang tunanetra kesulitan untuk mengenali nominal uang dengan cara meraba uang yang akan digunakan.

Dengan demikian dibutuhkan alat yang dapat membantu penyandang disabilitas tunetra untuk mengecek nominal mata uang rupiah untuk menghindari agar tidak terjadinya mendapatkan uang palsu dan juga memudahkan mereka dalam bertransaksi dengan aman dan nyaman.

### **1.1.1 Rumusan Masalah**

Berdasarkan latar belakang tersebut, dapat diidentifikasi masalah-masalah sebagai berikut :

- a. Bagaimana membangun alat sistem pendeteksi nominal uang dengan prediksi *machine learning*?
- b. Bagaimana agar alat tersebut dapat dibawa kemanapun (*portable*)?
- c. Bagaimana pengguna agar dapat mengerti dalam mengoperasikan alat tersebut?

### **1.1.2 Manfaat**

Tugas Akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

- a. Memudahkan penyandang disabilitas tunanetra dalam mendeteksi nominal mata uang rupiah.
- b. Meminimalisir kesalahan dalam membedakan uang.

### **1.1.3 Tujuan**

Pembuatan Tugas Akhir ini bertujuan untuk merancang dan membangun sistem pembaca nominal uang kertas rupiah menggunakan prediksi *machine learning* untuk membangun teknologi yang belum ada sebelumnya sehingga menghasilkan suatu alat yang memudahkan penyandang disabilitas (tunanetra) untuk terhindar dari kesalahan dalam membedakan uang.

## **1.2 Batasan Masalah**

Dalam pembuatan Tugas Akhir ini, masalah yang akan dibahas terbatas pada beberapa faktor diantaranya :

- a. Mesin dirancang hanya untuk mengenali uang kertas rupiah yaitu Rp 100.000 (Seratus Ribu Rupiah), Rp 50.000 (Lima Puluh Ribu Rupiah), Rp 20.000 (Dua Puluh Ribu Rupiah) dan Rp 10.000 (Sepuluh Ribu Rupiah), Rp 5.000 (Lima Ribu Rupiah), Rp 2.000 (Dua Ribu Rupiah), Rp 1.000 (Seribu Rupiah)
- b. Mesin membaca uang satu-persatu.
- c. Keadaan uang dalam kondisi baru dan tidak terlipat.

## **1.3 Sistematika Penulisan**

Dalam penyusunan tugas akhir ini sistematika penulisan dibagi menjadi beberapa bab, meliputi :

**BAB I PENDAHULUAN,** bab pendahuluan berisi latar belakang masalah dibuatnya tugas akhir, perumusan masalah yang menjadi kajian dari sistem, manfaat maupun tujuan pembuatan yang hendak dicapai dalam tugas akhir, batasan masalah, dan sistematika penulisan tugas akhir.

**BAB II TINJAUAN PUSTAKA,** bab tinjauan pustaka berisi beberapa penelitian yang telah dilakukan oleh ahli dan berkaitan dengan tugas akhir serta membantu dalam pembuatan tugas akhir. selain itu, terdapat perbandingan sistem yang telah ada dengan tugas akhir yang dikerjakan dengan adanya beberapa indikator.

**BAB III KEGIATAN PELAKSANAAN,** bab kegiatan pelaksanaan berisi uraian tahap kegiatan pelaksanaan dan penyelesaian tugas akhir.

**BAB IV ANALISIS DAN PEMBAHASAN,** bab analisis dan pembahasan berisi analisis rancangan sistem dan pembahasan sistem yang dibuat dalam tugas akhir.

**BAB V KESIMPULAN,** bab kesimpulan berisi pernyataan singkat dan tepat yang dijabarkan dari analisis dan pembahasan dari tugas akhir.

**DAFTAR PUSTAKA,** daftar pustaka berisi judul-judul beserta pengarang dari buku dan jurnal yang digunakan sebagai referensi dalam pembuatan tugas akhir ini.

**LAMPIRAN,** lampiran memuat informasi tambahan yang tidak esensial dan bertujuan.

## BAB II TINJAUAN PUSTAKA

Tinjauan Pustaka membahas mengenai penelitian terkait yang pernah dilakukan dan berkaitan dengan tugas akhir Rancang Bangun Alat Deteksi Keaslian dan Nominal Mata Uang Rupiah untuk Tunenetra. Selain itu terdapat perbandingan fitur-fitur dan teknologi antara sistem yang dibuat dengan sistem lain yang telah dibuat penelitian.

### 2.1 Landasan Teori

#### 2.1.1 Pengertian Mata Uang Rupiah

Uang dalam ilmu ekonomi tradisional di artikan sebagai alat tukar yang dapat diterima secara umum. Alat tukar itu dapat berupa benda apapun yang dapat diterima oleh masyarakat dalam pertukaran barang dan jasa. Sedangkan dalam ilmu ekonomi modern uang diartikan sebagai sesuatu yang tersedia secara umum dan diterima sebagai alat pembayaran bagi pembelian barang dan jasa serta kekayaan berharga lainnya dan untuk pembayaran utang. Jadi uang merupakan suatu benda yang diterima secara umum oleh masyarakat untuk mengukur nilai, menukar, dan melakukan pembayaran atas pembelian barang dan jasa[1].



**Gambar 2.1 Mata Uang Rupiah**

(Sumber:

[https://upload.wikimedia.org/wikipedia/id/thumb/6/68/Uang\\_Kertas\\_Rupiah\\_Indonesia\\_2020.png/252px-Uang\\_Kertas\\_Rupiah\\_Indonesia\\_2020.png](https://upload.wikimedia.org/wikipedia/id/thumb/6/68/Uang_Kertas_Rupiah_Indonesia_2020.png/252px-Uang_Kertas_Rupiah_Indonesia_2020.png))

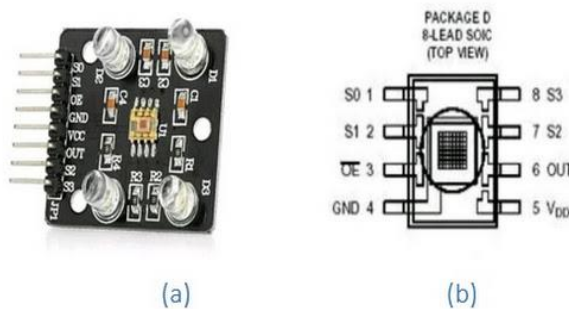
### 2.1.2 Sensor TCS230



**Gambar 2.2 Sensor TCS230**

(Sumber : <https://beetrona.com/wp-content/uploads/2019/08/Color-Recognition-Sensor-TCS230-600x600.jpg>)

Sensor warna yang digunakan untuk pendeteksi suatu objek benda atau warna dari objek yang di monitor yaitu Sensor warna TCS230. Sensor warna TCS230 juga dapat digunakan sebagai sensor gerak, dimana sensor mendeteksi gerakan suatu objek berdasarkan perubahan warna yang diterima oleh 11 sensor. Pada dasarnya sensor warna TCS230 merupakan konverter yang diprogram untuk mengubah warna menjadi frekuensi yang tersusun atas konfigurasi *silicon photodiode* dan konverter arus ke frekuensi dalam IC CMOS monolithic yang tunggal. Keluaran dari sensor ini adalah gelombang kotak (*duty cycle 50%*) frekuensi yang berbanding lurus dengan intensitas cahaya (*irradiance*)[2].



**Gambar 2.3 Pin TCS230**

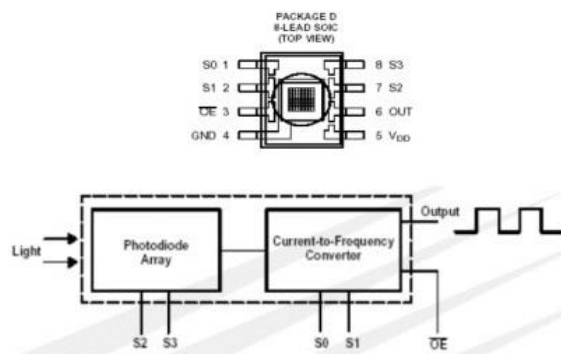
(Sumber : [https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcT7wkZkJjOPdbLDO\\_o-O-9axyR7NRKdPcVMTQ&usqp=CAU](https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcT7wkZkJjOPdbLDO_o-O-9axyR7NRKdPcVMTQ&usqp=CAU))

Keluaran frekuensi skala penuh dapat diskalakan oleh satu dari tiga nilai-nilai yang ditetapkan via dua kontrol pin input. Masukan digital dan keluaran digital memungkinkan antarmuka langsung ke mikrokontroler atau sirkuit logika lainnya.

Konverter cahaya ke frekuensi membaca sebuah *array* 8x8 dengan 16 buah konfigurasi fotodiode yang berfungsi sebagai filter warna merah, 16 *photodiode* sebagai filter warna biru dan 16 photo dioda lagi tanpa filter warna. Sensor warna TCS230 merupakan sensor yang dikemas dalam chip DIP 8 pin dengan bagian muka transparan sebagai tempat menerima intensitas cahaya yang berwarna. Berikut bentuk fisik dan skema pin dari sensor warna TCS230.

### 2.1.2.1 Prinsip Kerja Sensor Warna TCS230

Prinsip kerja dari sensor warna TCS230 yaitu *Photodiode* akan mengeluarkan arus yang besarnya sebanding dengan kadar warna dasar cahaya yang menyimpannya.



**Gambar 2.4 Skema kerja TCS230**

(Sumber : <http://elektronika-dasar.web.id/wp-content/uploads/2012/07/Konstruksi-Sensor-Warna-TCS230.jpg>)

Arus ini kemudian dikonversikan menjadi sinyal kotak atau pulsa digital dengan frekuensi sebanding dengan besarnya arus. Frekuensi Output ini bisa diskala dengan mengatur kaki selektor S0 dan S1. Sehingga dapat 15 diskripsikan juga, bahwa modul TCS ini bekerja sebagai konverter, dimana modul TCS ini mengubah sinyal analog inputan berupa arus menjadi bentuk frekuensi.

**Tabel 2.1 Output Frequency Scaling**

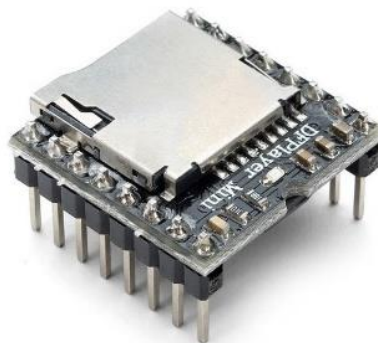
S0	S1	OUTPUT FREQUENCY SCALING ( $f_o$ )
L	L	Power down
L	H	2%
H	L	20%
H	H	100%

Pada sensor warna TCS230 terdapat selektor S2 dan S3 yang berfungsi untuk memilih kelompok konfigurasi *photodiode* yang akan digunakan atau dipakai. Kombinasi fungsi S2 dan S3 dalam pemilihan kelompok *photodiode* adalah sebagai berikut :

Tabel 2.2 Tipe fotodiode

S2	S3	PHOTODIODE TYPE
L	L	Red
L	H	Blue
H	L	Clear (no filter)
H	H	Green

### 2.1.3 DF Player Mini



**Gambar 2.5. DFPlayer Mini**

(Sumber : <https://cdn.shopify.com/s/files/1/2822/2674/products/voltaat-dfplayer-mini-mp3-player-module-3592238334054.jpg?v=1628492951>)

DF Player Mini adalah modul *sound player* yang dapat mendukung beberapa file salah satunya adalah file mp3 yang umumnya digunakan sebagai format *sound* file. DF Player mini ini mempunyai 16 pin *interface* yaitu berupa pin standar DIP dan pin *header* pada kedua sisinya.

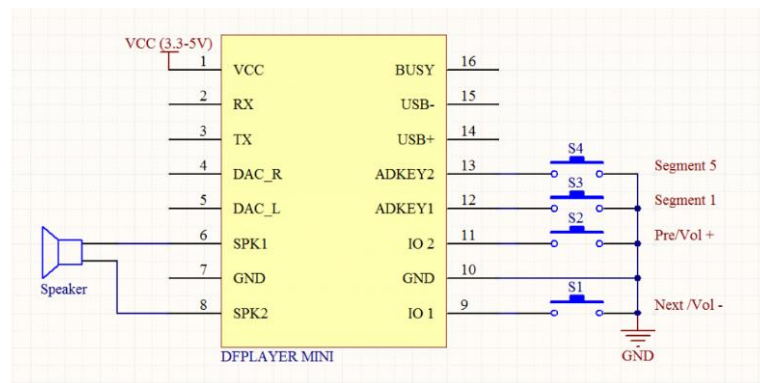
DF Player mini tersebut dapat bekerja sendiri ataupun bekerja bersama dengan mikrokontroler melalui koneksi serial[3].



Tabel 2.3. Fungsi Pin DF Player Mini

<b>Nama</b>	<b>Deskripsi</b>	<b>Catatan</b>
VCC	<i>Input Tegangan</i>	DC 3,2-5.0V
RX	<i>UART input serial</i>	
TX	<i>UART output serial</i>	
DAC_R	<i>Output audio saluran kanan</i>	<i>Earphone drive dan amplifier</i>
DAC_L	<i>Output audio saluran kiri</i>	
SPK2	<i>Speaker</i>	<i>Speaker power (&lt;3W)</i>
GND	<i>Ground</i>	<i>Power ground</i>
SPK1	<i>Speaker</i>	<i>Speaker power (&lt;3w)</i>
I0 1	<i>Trigger port 1</i>	Tekan sebentar untuk memainkan lagu berikutnya (tahan lama untuk mengurangi volume)
GND	<i>Ground</i>	<i>Power ground</i>
I02	<i>Trigger port 2</i>	Tekan lama untuk memainkan lagu berikutnya (tahan lama untuk meningkatkan volume)
ADKEY1	<i>AD port 1</i>	Memicu memainkan segmen pertama
ADKEY2	<i>AD port 2</i>	Memicu memainkan segmen kelima
UBS +	<i>USB + DP</i>	<i>Port USB</i>
USB -	<i>USB - DM</i>	<i>Port USB</i>
<i>Busy</i>	<i>Memainkan status</i>	Rendah Memainkan musik
		Tinggi tidak memainkan musik

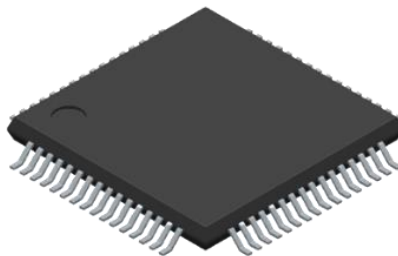
Berikut adalah *wiring* yang sangat sederhana dari penggunaan modul DF *Player* mini, bahkan dapat mengabaikan *push button* S3 dan S4 yang terhubung di pin ADKey. Hal ini hanya memerlukan 2 buah *push button* dan 1 mini *speaker* yaitu menekan S1 dan S2 dengan cepat untuk *next* atau *previous* dan tekan S1 dan S2 secara ditahan untuk atur *volume*.



**Gambar 2.6 Wiring DF Player Mini**

(Sumber : <https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSC972xMCbrx2QZ3ufl56b-PE6L9Tc4sayppA&usqp=CAU>)

#### 2.1.4 Mikrokontroler



**Gambar 2.7 Mikrokontroler**

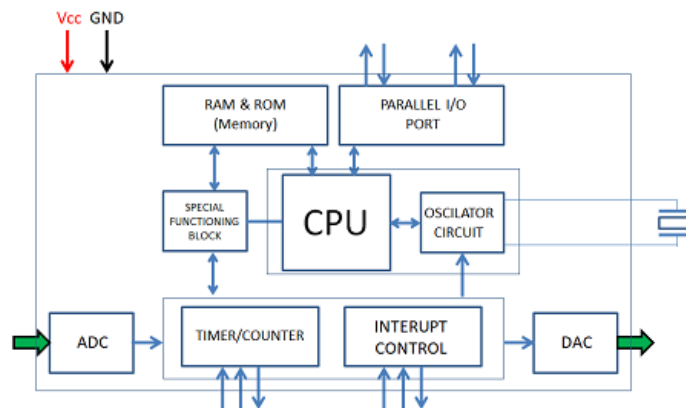
(Sumber : [https://res.cloudinary.com/rsc/image/upload/b\\_rgb:FFFFFF,c\\_pad,dpr\\_1.0,f\\_auto,h\\_843,q\\_auto,w\\_1500/c\\_pad,h\\_843,w\\_1500/R8996928-01?pgw=1&pgwact=1](https://res.cloudinary.com/rsc/image/upload/b_rgb:FFFFFF,c_pad,dpr_1.0,f_auto,h_843,q_auto,w_1500/c_pad,h_843,w_1500/R8996928-01?pgw=1&pgwact=1))

Sebuah komputer kecil yang dikemas dalam bentuk *chip IC (Integrated Circuit)* dan dirancang untuk melakukan tugas atau operasi tertentu di sebut Mikrokontroler. Pada dasarnya, sebuah IC Mikrokontroler terdiri dari satu atau lebih Inti Prosesor (CPU), Memori (RAM dan ROM) serta perangkat *input* dan *output* yang dapat diprogram. Dalam pengaplikasiannya, Pengendali Mikro yang dalam bahasa Inggris disebut dengan *microcontroller* ini digunakan dalam produk ataupun perangkat yang dikendalikan secara otomatis seperti sistem kontrol mesin mobil, perangkat medis, pengendali jarak jauh, mesin, peralatan listrik, mainan dan perangkat-perangkat yang menggunakan sistem tertanam lainnya. Penggunaan Mikrokontroler ini semakin populer karena kemampuannya yang dapat mengurangi ukuran dan biaya pada suatu produk atau desain apabila dibandingkan dengan

desain yang dibangun dengan menggunakan mikroprosesor dengan memori dan perangkat input dan output secara terpisah[4].

#### 2.1.4.1 Diagram Blok dan Struktur Mikrokontroler

Diagram Blok dan Struktur Mikrokontroler beserta penjelasan singkat tentang bagian-bagian utamanya adalah sebagai berikut:



**Gambar 2.8 Diagram Blok dan Struktur Mikrokontroler**

(Sumber : <https://teknikelektronika.com/wp-content/uploads/2020/03/Pengertian-Mikrokontroler-Microcontroller.png>)

- CPU adalah otak mikrokontroler. CPU bertanggung jawab untuk mengambil instruksi (*fetch*), menerjemahkannya (*decode*), lalu akhirnya dieksekusi (*execute*). CPU menghubungkan setiap bagian dari mikrokontroler ke dalam satu sistem. Fungsi utama CPU adalah mengambil dan mendekode instruksi. Instruksi yang diambil dari memori program harus diterjemahkan atau melakukan *decode* oleh CPU tersebut.
- Fungsi memori dalam mikrokontroler sama dengan mikroprosesor. Memori ini digunakan untuk menyimpan data dan program. Sebuah mikrokontroler biasanya memiliki sejumlah RAM dan ROM (EEPROM, EPROM dan lain-lainnya) atau memori *flash* untuk menyimpan kode sumber program (*source code program*).
- Port Input /Output* paralel digunakan untuk mendorong atau menghubungkan berbagai perangkat seperti LCD, LED, *printer*, memori dan perangkat *input/output* lainnya ke mikrokontroler.
- Port serial menyediakan berbagai antarmuka serial antara mikrokontroler dan periferal lain seperti port paralel.

- e) *Timer* dan *Counter* adalah salah satu fungsi yang sangat berguna dari Mikrokontroler. Mikrokontroler mungkin memiliki lebih dari satu timer dan counter. Pengatur waktu (*Timer*) dan Penghitung (*Counter*) menyediakan semua fungsi pengatur waktu dan penghitungan di dalam mikrokontroler. Operasi utama yang dilakukan di bagian ini adalah fungsi jam, modulasi, pembangkitan pulsa, pengukuran frekuensi, osilasi, dan lain sebagainya. Bagian ini juga dapat digunakan untuk menghitung pulsa eksternal.
- f) Konverter ADC digunakan untuk mengubah sinyal analog ke bentuk digital. Sinyal input dalam konverter ini harus dalam bentuk analog (misalnya Output dari Sensor) sedangkan Outputnya dalam bentuk digital. Output digital dapat digunakan untuk berbagai aplikasi digital seperti layar digital pada Perangkat pengukuran.
- g) DAC melakukan operasi pembalikan konversi ADC. DAC mengubah sinyal digital menjadi format analog. Ini biasanya digunakan untuk mengendalikan perangkat analog seperti motor DC dan lain sebagainya.
- h) Kontrol interupsi atau *Interrupt Control* digunakan untuk menyediakan interupsi (penundaan) untuk program kerja. Interrupt dapat berupa eksternal (diaktifkan dengan menggunakan pin interrupt) atau internal (dengan menggunakan instruksi interupsi selama pemrograman).
- i) Beberapa Mikrokontroler yang hanya dapat digunakan untuk beberapa aplikasi khusus (misalnya sistem Robotik), pengontrol ini memiliki beberapa port tambahan untuk melakukan operasi khusus tersebut yang umumnya dinamakan dengan Blok Fungsi Khusus.

### 2.1.5 ESP32



**Gambar 2.9 ESP32**

(Sumber : [https://lzd-img-global.slatic.net/g/p/31e2c5afee2cb72fc5c3d0b22cfba331.jpg\\_720x720q80.jpg\\_.webp](https://lzd-img-global.slatic.net/g/p/31e2c5afee2cb72fc5c3d0b22cfba331.jpg_720x720q80.jpg_.webp))

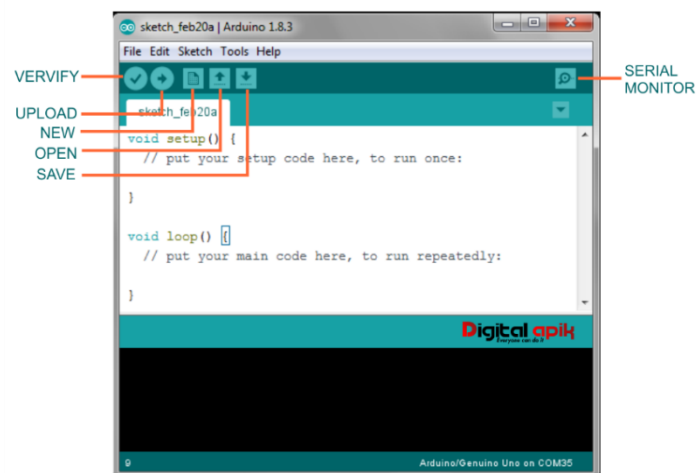
ESP32 adalah mikrokontroler yang dikenalkan oleh Espressif System merupakan penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul WiFi dalam chip sehingga sangat mendukung untuk membuat sistem aplikasi *Internet of Things*. Terlihat pada gambar di atas merupakan pin out dari ESP32. Pin tersebut dapat dijadikan input atau output untuk menyalakan LCD, lampu, bahkan untuk menggerakkan motor DC[5].

Pin out ESP32 terdiri dari:

- 18 ADC (*Analog Digital Converter*, berfungsi untuk merubah sinyal analog ke digital).
- 2 DAC (*Digital Analog Converter*, kebalikan dari ADC).
- 16 PWM (*Pulse Width Modulation*).
- 10 Sensor sentuh.
- 2 jalur antarmuka UART.
- Pin antarmuka I2C, I2S, dan SPI.

### 2.1.6 Integrated Development Environment (IDE) Arduino

Sebuah software yang digunakan untuk mengembangkan aplikasi mikrokontroler yang dapat melakukan kinerja mulai dari menuliskan *source* program, kompilasi, upload hasil kompilasi dan uji coba secara terminal serial disebut IDE (*Integrated Development Environment*)[6].



**Gambar 2.10 IDE Arduino**

(Sumber : <https://4.bp.blogspot.com/-YLOLqjxHbpc/XGxzMBliuXI/AAAAAAAAADQ/pQJM8vC2rpEIsXtKtZ0QvtmPyYnq7G429QCLcBGAs/s1600/bgian.png>)

Keterangan dari gambar yaitu :

- a) *Icon* menu *verify* yang bergambar ceklis berfungsi untuk mengecek program yang ditulis apakah ada yang salah atau error.
- b) *Icon* menu *upload* yang bergambar panah ke arah kanan berfungsi untuk memuat atau transfer program yang dibuat di software arduino ke hardware arduino.
- c) *Icon* menu *New* yang bergambar sehelai kertas berfungsi untuk membuat halaman baru dalam pemrograman.
- d) *Icon* menu *Open* yang bergambar panah ke arah atas berfungsi untuk membuka program yang disimpan atau membuka program yang sudah dibuat dari pabrikan software arduino.
- e) *Icon* menu *Save* yang bergambar panah ke arah bawah berfungsi untuk menyimpan program yang telah dibuat atau dimodifikasi.
- f) *Icon* menu serial monitor yang bergambar kaca pembesar berfungsi untuk mengirim atau menampilkan serial komunikasi data saat dikirim dari hardware arduino.

### 2.1.7 Python

*Python* merupakan bahasa pemrograman *interpretative* multiguna. *Python* memiliki struktur data tingkat tinggi yang efisien dan pendekatan yang sederhana, namun efektif untuk pemrograman berorientasi objek. *Python* lebih menekankan pada keterbacaan kode agar lebih mudah untuk memahami sintaks.



**Gambar 2.11 Bahasa *python***

(Sumber : <https://www.petanikode.com/img/python/python.avif>)

Library yang digunakan di dalam *python* :

- a) *Pandas*, merupakan paket *python* yang *open source* berguna untuk menganalisis data serta membangun *machine learning*. Dibuat berdasarkan satu paket lain bernama *numpy* untuk mendukung *arrays* multi dimensi. Cara kerja yaitu data berupa CSV diubah menjadi objek *python* dengan *rows* dan *column* diubah menjadi data *frame*. Data yang sudah diolah akan mudah untuk dianalisis karena berbentuk daftar *library*[7].
- b) *Sklearn*, merupakan module dari Bahasa *python* dibangun berdasarkan *numpy* dan *matplotlib*. Berguna untuk membantu melakukan *processing* data ataupun melakukan *training* data untuk *machine learning*[8].
- c) *Flask*, merupakan sebuah *framework* atau *library* keluaran *python*, berguna untuk alat bantu pengembangan *web development*. *Library* ini juga memiliki fitur bawaan yang menarik seperti pengembangan server dan pewaktuan kemudian dukungan integrasi pada pengujian unit[9].
- d) *Firebase* admin, merupakan *library server* yang dapat digunakan untuk interaksi dengan *firebase*. Berguna untuk membaca dan menulis *realtime* database.

### 2.1.8 Machine Learning

*Machine Learning* menurut IBM adalah cabang dari kecerdasan buatan AI dan ilmu computer yang berfokus pada penggunaan data dan algoritma untuk meniru cara belajar manusia secara bertahap dan meningkatkan akurasi. *Machine learning* sangat penting dari bidang perkembangan data melalui penggunaan statistik. *Machine learning* juga dilatih untuk mengklasifikasikan dan memprediksi dalam pengembangan data[10]. Algoritma K-Nearest Neighbor (KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Data pembelajaran diproyeksikan ke ruang berdimensi banyak, dimana masing masing dimensi merepresentasikan label dari data. [11].



**Gambar 2.12 Machine Learning**

(Sumber : <https://www.mas-software.com/wp-content/uploads/2021/07/Machine-Learning.jpg>)

Metode yang digunakan dalam *machine learning* :

- a) *Supervised Learning*, metode yang dilakukan dengan pemberian label pada dataset yang digunakan oleh *machine learning*. Cara kerja yaitu informasi yang masuk sebagai input dan data berlabel sebagai hasil atau output.
- b) *Unsupervised Learning*, metode yang dilakukan tanpa pengawasan, prosesnya dilakukan pada dataset mentah yang tidak berlabel dan algoritma akan mencoba mengidentifikasi pola dan relasinya.
- c) *Reinforcement Learning*, metode ini dijalankan menggunakan dataset bersistem “*rewards/punishment*” yaitu menawarkan umpan balik ke algoritma untuk belajar dari pengalamannya secara random atau coba coba sama seperti pemahaman pola yang dilakukan manusia yaitu belajar dari percobaan.

### 2.1.9 Speaker



**Gambar 2.13 Speaker**

(Sumber : [https://gd.image-gmkt.com/li/458/322/1000322458.g\\_520-w-st\\_g.jpg](https://gd.image-gmkt.com/li/458/322/1000322458.g_520-w-st_g.jpg))

Speaker memiliki peran yang penting dalam mengeluarkan hasil pemrosesan berupa suara. Tentunya, pada penelitian kali ini sangat membutuhkan speaker untuk membantu seorang disabilitas tunanetra dalam mengetahui keaslian mata uang rupiah. Speaker dapat berfungsi apabila didukung perangkat keras berupa



sound card atau pemroses audio/suara. Sementara untuk modelnya, speaker memiliki beragam bentuk, dan nilai ohm speaker. Lantas apa yang dimaksud nilai ohm speaker, yaitu nilai resistansi atau besar tahanan speaker yang juga dapat disebut impedansi. Impedansi sendiri yaitu sebuah parameter yang digunakan untuk menentukan nilai resistansi pada arus AC ( bolak balik )[12].

#### 2.1.10 Firebase



**Gambar 2.14 Firebase**

(Sumber : <https://firebase.google.com/images/social.png>)

Salah satu database dengan basis NoSQL dan merupakan tools yang banyak digunakan developer aplikasi dan web juga terkenal sederhana dan powerful sehingga dapat mempercepat pekerjaan adalah *firebase*. *Firebase* juga merupakan BaaS (*Backend as a service*) yang dapat mempercepat pekerjaan developer sehingga para developer dapat focus pada pengembangan aplikasi tanpa effort besar untuk *back-end*[13].

Tabel 2.4 Perbandingan Teknologi

No.	Penulis	Fitur				
		<i>Machine Learning</i>	Suara	Cahaya UV	Deteksi Nominal	Otomatis Sistem
1.	(Ribka, J. S. (2021).)	-	-	-	√	√
2.	(Dihan Retno, M., Auliq, an S., & Brahma Nugroho S.Si.)	-	-	√	√	√
3.	(Indriani, T., Elektro, J. T., Teknik, F., & Belitung, U. B. (2017).)	-	-	√	-	√
4.	(Nurjanah, M. S. (2022).)	-	-	√	√	√

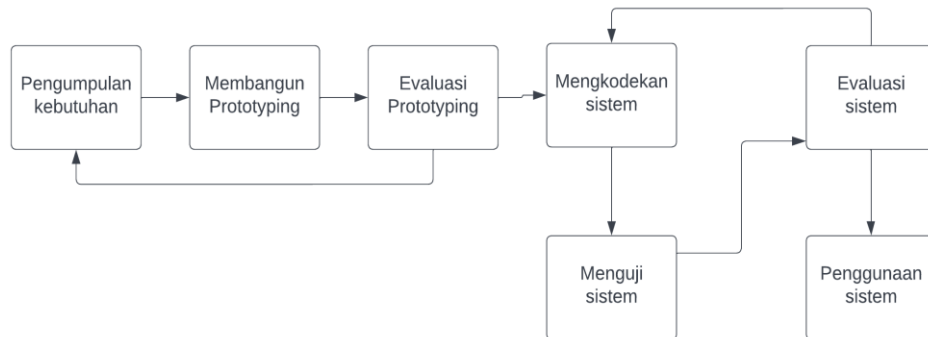
Tabel 2.4 Perbandingan Teknologi (lanjutan)

5.	(Dwi, A. P. (2014).)	-	√	√	√	√
6.	(Gunawan, Indra Ahmadi, Hamzan. (2021).)	-	-	-	√	√
7.	(Hidayat, Isfanul. (2017).)	-	-	-	√	-
8.	(Hinestroza, Denniye. (2018).)	-	-	-	√	√

## BAB III KEGIATAN PELAKSANAAN

### 3.1 Metodologi

Metodologi penelitian adalah sekumpulan kegiatan dan peraturan, serta prosedur yang digunakan oleh peneliti. Metodologi yang digunakan adalah metode *prototype*. Metode ini sering dinamakan siklus hidup (*life cycle using prototyping*) yang dimana memiliki metode dengan pengembangan yang sangat cepat dan pengujian model kerja baru sehingga dapat digunakan dengan baik. Metode ini didasarkan pada konsep model bekerja (*working model*) dan memiliki tujuan untuk mengembangkan model menjadi sistem *final*. Yang menjadi ciri khas metode ini adalah pengembang *system* dapat melihat dan melakukan eksperimen sejak awal proses pengembangan. Tahapan metode *prototype* dapat dilihat pada Gambar 3.1. di bawah ini.



**Gambar 3.1 Tahapan Metode *Prototype***

#### 1. Pengumpulan kebutuhan

Langkah pertama yang dilakukan dalam metode *prototype* adalah mengidentifikasi seluruh perangkat dan permasalahan. Pengumpulan kebutuhan sangat penting dalam proses ini karena kita akan menemukan kebutuhan garis besar dari sistem dan kita dapat mengerti langkah apa yang kita perbuat setelah mengetahui permasalahannya di dalam analisa dan identifikasi.

## 2. Membangun *prototype*

Langkah selanjutnya membangun *prototype* yang berfokus pada penyajian pengguna.

## 3. Evaluasi *prototype*

Sebelum ke langkah selanjutnya, kita harus memastikan langkah 1 dan 2 berhasil dikarenakan ini merupakan langkah untuk mengevaluasi langkah yang sudah kita lakukan benar atau ada yang kurang dan sebagai faktor penentu keberhasilan di langkah selanjutnya.

## 4. Mengkodekan *system*

Langkah selanjutnya yaitu pengkodean, di sini kita perlu mengetahui program yang akan kita buat menggunakan bahasa pemrograman apa saja dan mengaplikasikan kebutuhan dalam bentuk kode program.

## 5. Menguji *system*

Langkah selanjutnya yaitu menguji *system*, pada tahap ini *system* akan di-*testing* dengan cara *white box* yang artinya menguji kodingan program yang telah dibuat.

## 6. Evaluasi *system*

Langkah selanjutnya yaitu evaluasi *system*, pada tahap ini kita perlu mengevaluasi keseluruhan langkah yang telah kita lakukan apakah sudah sesuai dengan kebutuhan kita dan berjalan dengan baik.

## 7. Menggunakan *system*

Langkah terakhir *system prototyping* sudah selesai dan siap untuk digunakan.

## 3.2 Analisis Kebutuhan

Bagian ini akan membahas mengenai analisis kebutuhan berupa kebutuhan perangkat keras (*hardware*) dan kebutuhan perangkat lunak (*software*) yang diperlukan dalam pembuatan Tugas Akhir.

### 3.2.1 Perangkat Keras yang digunakan

Perangkat keras yang digunakan untuk membangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi *Machine Learning* untuk Tunanetra adalah sebagai berikut :

- a. ESP32 DevKits
- b. TCS230 sensor warna
- c. *DFPlayer mini*
- d. Speaker
- e. Kabel jumper
- f. Box

### 3.2.2 Perangkat Lunak yang digunakan

Perangkat lunak yang digunakan untuk membangun Alat Deteksi Keaslian dan Nominal Mata Uang Rupiah untuk Tunanetra adalah sebagai berikut :

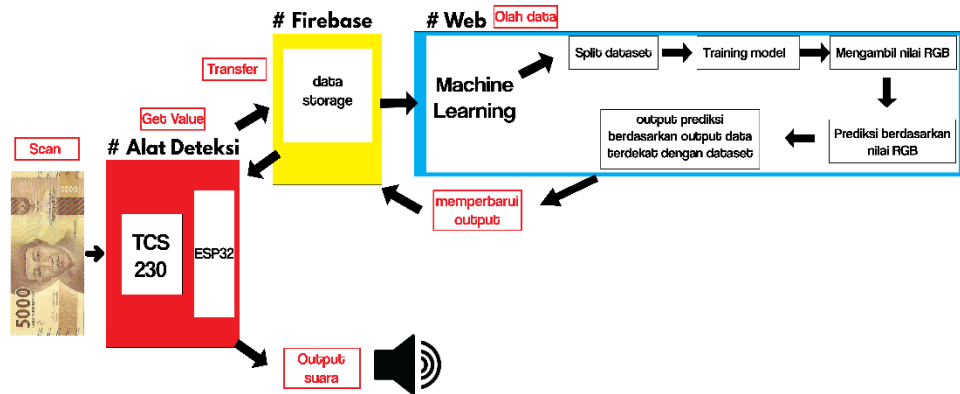
- a. *Arduino IDE*
- b. *Google Chrome*
- c. *Firebase*
- d. *Visual Studio Code*
- e. *Python*
- f. *Windows 10*

## 3.3 Perancangan Sistem

Pada bagian ini membahas tentang perancangan desain sistem meliputi *block diagram*, *flowchart* dan *wiring* yang digunakan dalam sistem. Pada pemilihan yang digunakan pada *machine learning* dilakukan beberapa proses sehingga mendapatkan hasil yang terbaik. Sampel yang digunakan adalah seluruh nominal mata uang kertas rupiah dengan masing-masing nominal ada empat lembar. Sampel tersebut nantinya akan dipindai nilai RGB-nya dan dijadikan dataset untuk dilakukan proses *training* dan testing oleh algoritma yang akan digunakan. Algoritma yang akan diuji adalah Random Forest Classifier (RFC), Gradient Boost Classifier (GBC), dan K-Nearest Neighbor (KNN). Kemudian algoritma tersebut di-*deploy* ke dalam web yang menggunakan *library flask* dan diuji ketepatan output dari algoritma yang diuji.

### 3.3.1 Block Diagram

Block diagram digunakan untuk mengetahui bagaimana cara kerja sistem dalam bentuk blok – blok diagram yang sudah digambarkan pada gambar 3.2



**Gambar 3.2 Block Diagram System**

Keterangan mengenai alur block diagram :

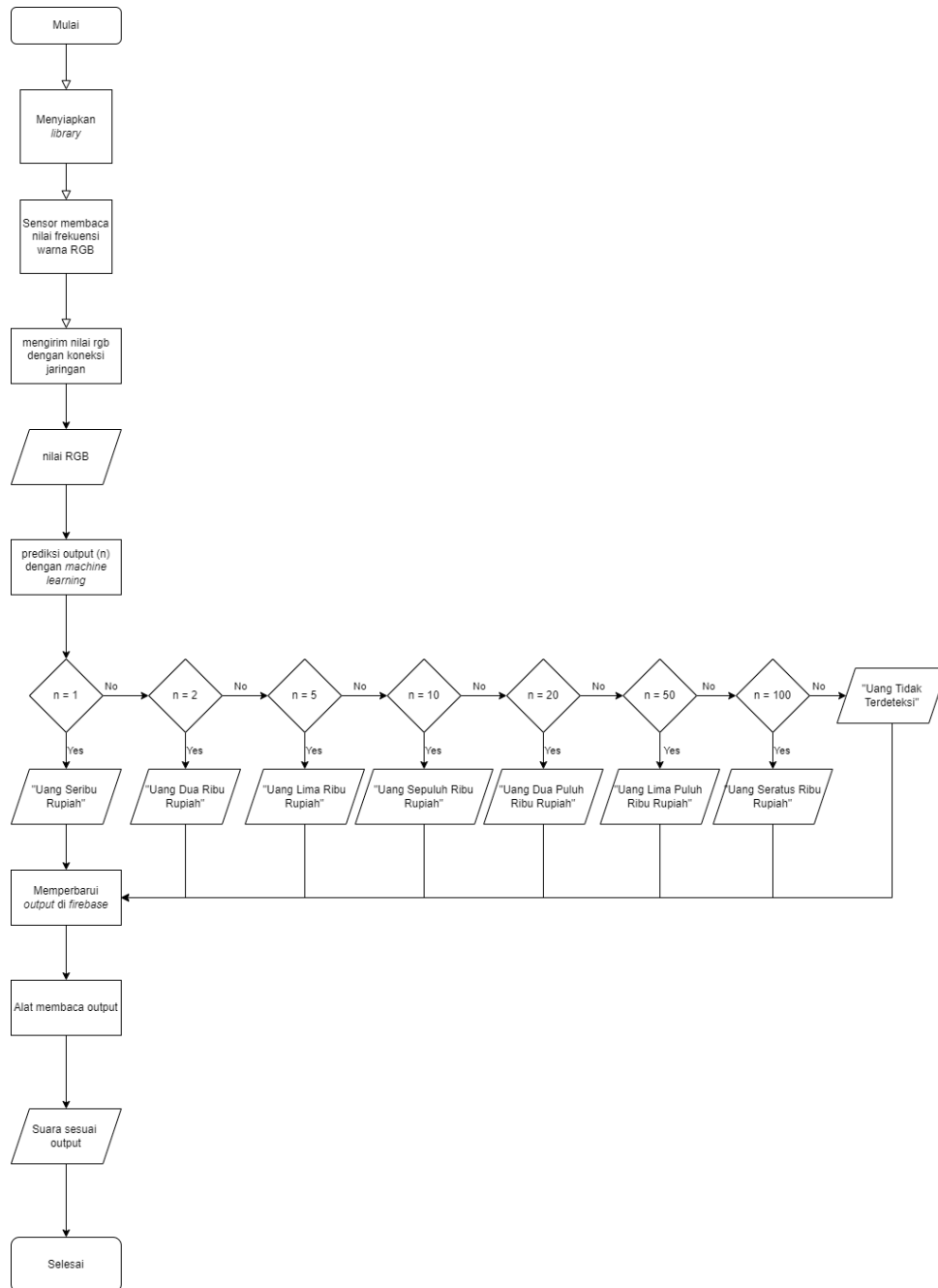
- Scan* RGB yaitu proses mendapatkan parameter warna dari setiap mata uang yang di dipindai.
- Alat deteksi berfungsi sebagai menyimpan atau meneruskan nilai yang sudah didapat dari proses scan menggunakan tcs230 yang berupa nilai RGB.
- Firestore* berfungsi sebagai *data storage* yang terkoneksi dengan Arduino dan juga python sehingga data dapat saling transfer untuk proses selanjutnya.
- Python* web berfungsi sebagai *machine learning* dan untuk memprediksi nilai RGB yang sudah di dapat dari *firebase* dan akan menghasilkan *output* nominal.
- Firestore* selanjutnya *firebase* berfungsi untuk mengambil nominal output yang sudah di proses di *machine learning* python web.

Proses didalam *machine learning* :

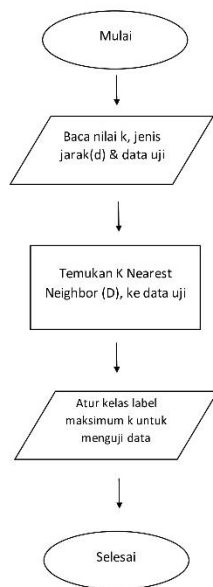
- Split dataset, membagi ke dalam data *training* dan data *testing*.
  - *Training model*, melatih model dengan dataset yang dimiliki.
  - Mengambil nilai *RGB*, mengambil nilai yang masuk dari *firebase*.
  - Prediksi berdasarkan nilai RGB, memprediksi nilai rgb yang didapatkan menggunakan algoritma KNN.
  - *Output* prediksi, mendapatkan hasil output dari nilai rgb yang telah di prediksi.
- Alat berfungsi sebagai mendapatkan nilai terakhir yang sudah di olah berupa *output* nominal.
  - Output* berfungsi sebagai perubah dari data *string* menjadi suara.

### 3.3.2 Flowchart

Rancang Bangun Alat Deteksi Nominal Mata Uang Rupiah dengan Prediksi *Machine Learning* untuk Tunanetra dibangun menggunakan mikrokontroler yaitu ESP32, TCS230, *DFPlayer mini* dan *Speaker*. Dalam pengimplementasiannya sendiri, *flowchart* digunakan untuk melihat bagaimana alur cara kerja dari sistem yang dibuat bisa bekerja dan *flowchart* algoritma KNN.

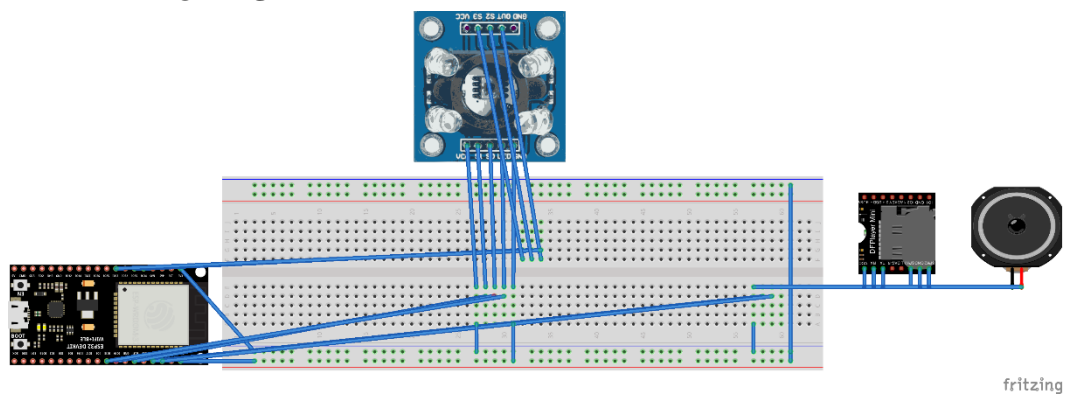


**Gambar 3.3 Flowchart Alur Kerja Sistem**



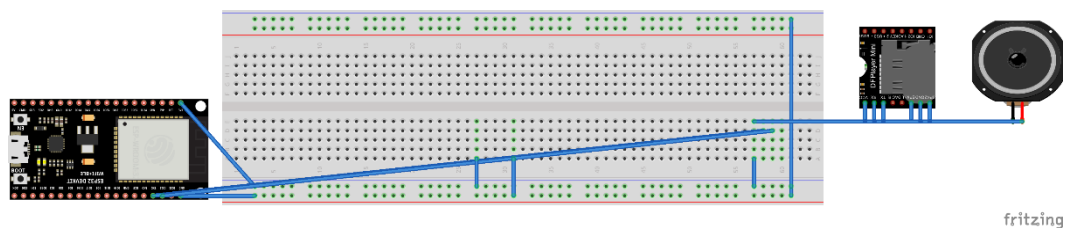
**Gambar 3.4 Flowchart Algoritma KNN**

### 3.3.3 Wiring Diagram



**Gambar 3.5 Wiring Keseluruhan Alat**

Pada Gambar 3.5 ESP 32 dihubungkan dengan sensor TCS230 untuk dapat mengambil data mata uang secara langsung dan dihubungkan dengan df player agar dapat memutar suara setelah keluar *output* dari Arduino IDE.



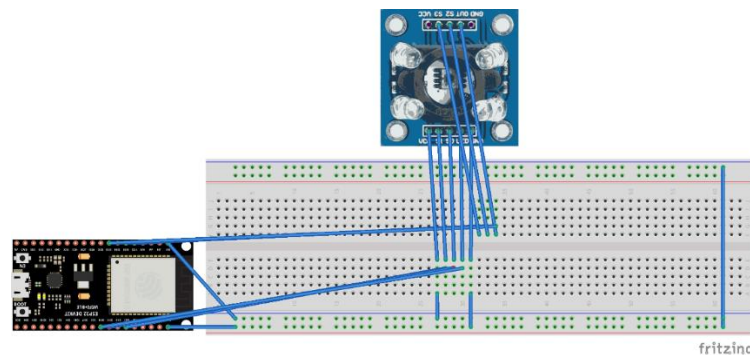
**Gambar 3.6 Wiring DFPlayer Mini dan Speaker**



Pada Gambar 3.6 merupakan *wiring* dari *DFPlayer mini* dan *speaker* yang kita *jumper* dari *bread board* ke *Arduino* dikarenakan ada beberapa pin yang jalannya sama untuk lebih jelasnya pin VCC kita hubungkan ke pin VIN di dalam ESP32 kemudian pin RX dan TX kita hubungkan ke dalam pin RX0 dan TX0 dan selanjutnya *speaker* memiliki dua kabel + dan – kita hubungkan langsung ke dalam *df player* agar terdapat output suaranya yaitu untuk pin + dan – kita hubungkan ke dalam pin *spk\_1* dan *spk\_2* setelah semua pin terhubung kita membutuhkan daya melalui pin GND.

Tabel 3.1 Koneksi antar pin *DFPlayer* dan ESP32

DF Player	ESP32	Speaker	DF Player
VCC	VIN	kabel +	spk_1
RX	RX0	kabel -	spk_2
TX	TX0		
GND	GND		



**Gambar 3.7 Wiring TCS230**

Pada Gambar 3.7 merupakan *wiring* dari sensor TCS230 yaitu sensor warna pada sensor ini sangat berperan penting dikarenakan untuk mendeteksi warna keaslian mata uang. Lebih jelasnya untuk pin yang terhubung terdapat pin S0 dan S1 yang terhubung ke dalam ESP32 yaitu pada pin D4 dan D5, untuk pin S2 dan S3 terhubung ke dalam pin D26 dan D25, untuk pin OUT dan VCC terhubung ke dalam D27 dan 3v3, terakhir membutuhkan daya pin GND terhubung ke dalam GND.

Tabel 3.2 Koneksi antar pin TCS230 dan ESP32

TCS230	ESP32
S0	D4
S1	D5
S2	D26
S3	D25
OUT	D27
VCC	3v3
GND	GND

### 3.3.4 Perancangan Pengujian User

Pada tahap ini dilakukan proses pengujian sistem yang sudah jadi kepada *user*. Dalam hal ini penulis menggunakan 2 sampel orang untuk melakukan pengujian sistem. Dalam pengujian sistem ini didapatkan data dengan cara memberikan beberapa poin pertanyaan seputar sistem ini serta mengharapkan kritik dan saran demi perkembangan selanjutnya.

Dalam pengujian Alat Deteksi Mata Uang Rupiah dengan prediksi *Machine Learning* untuk Tunanetra ini diberikan beberapa pertanyaan untuk mendapatkan hasil perhitungan yang digunakan sebagai acuan untuk mengetahui tingkat kepuasan pengguna dalam penggunaannya. Untuk daftar pertanyaan dapat dilihat pada Tabel 3.3, sedangkan untuk penilaian pengukuran tingkat kepuasan pengguna didasarkan pada Tabel 3.4 dan Tabel 3.5.

Tabel 3.3 Daftar Pertanyaan

No	Kriteria Pertanyaan
1	apakah anda setuju bahwa alat ini mempermudah tunanetra membedakan nominal uang?
2	apakah anda setuju alat ini mudah digunakan?
3	apakah anda setuju alat ini dapat mendeteksi uang dengan akurat?

Tabel 3.4 Indikator Penilaian

<b>Predikat</b>	<b>Nilai</b>
Sangat Setuju	5
Setuju	4
Cukup	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Tabel 3.5 Indikator Kepuasan Pengguna

<b>Kategori</b>	<b>Presentase</b>
Sangat Memuaskan	81%-100%
Memuaskan	61%-80%
Cukup Memuaskan	41%-60%
Tidak Memuaskan	21%-40%
Sangat Tidak Memuaskan	0-20%

Untuk menghitung kepuasan pengguna menggunakan rumus sebagai berikut:

$$\text{Presentase kepuasan pengguna (\%)} = \frac{\text{Total Kepuasan Pengguna}}{\text{Total Kepuasan Maksimum Pengguna}} \times 100\%$$

## BAB IV ANALISIS DAN PEMBAHASAN

### 4.1 Pengujian

Pengujian merupakan tahapan yang perlu dilakukan untuk menganalisis kinerja sistem. Dalam tahap pengujian dilakukan beberapa langkah, seperti berikut:

1. Deteksi dan pengiriman nilai *RGB*.
2. Pengujian *web python*.
3. Pengujian *Firebase*.
4. Pengujian *DFPlayer* dan speaker.

#### 4.1.1 Deteksi dan pengiriman nilai *RGB*

Tujuan dari pengujian ini untuk mengetahui keluaran frekuensi sensor warna dari objek uang kertas yang diujikan. Peralatan yang diperlukan dalam pengujian ini antara lain, sensor warna TCS 230, Mikrokontroller ESP32, uang kertas 1.000, 2.000, 5.000, 10.000, 20.000, 50.000, dan 100.000 masing-masing 4 buah.

- a. Nilai *RGB* setiap mata uang rupiah

Nilai dari setiap mata uang rupiah yang telah dilakukan proses *scanning* untuk mendapatkan nilai untuk dataset dapat dilihat sebagai berikut:

Tabel 4.1 Nilai *RGB* mata uang Rp. 1000

Mata Uang Rp. 1000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	89	98	95	87	90	73	Lusuh
2	108	117	108	93	104	104	Baik
3	108	118	108	92	103	103	Baik
4	108	118	107	93	104	101	Baik

Pada Tabel 4.1 menjelaskan nilai *RGB* pada mata uang seribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang berbeda sehingga didapatkan nilai *RGB* yang berbeda pada setiap kondisi fisik uang.

Tabel 4.2 Nilai RGB mata uang Rp. 2000

Mata Uang Rp. 2000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	86	90	76	120	125	109	Baik
2	87	90	76	118	123	107	Baik
3	85	90	76	89	92	76	Lusuh
4	86	91	76	108	111	99	Baik Tertekuk

Pada Tabel 4.2 menjelaskan nilai RGB pada mata uang dua ribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang berbeda sehingga didapatkan nilai RGB yang berbeda pada setiap kondisi fisik uang.

Tabel 4.3 Nilai RGB mata uang Rp. 5000

Mata Uang Rp. 5000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	88	117	98	80	103	97	Baik Tertekuk
2	80	102	86	81	104	98	Baik
3	70	82	76	79	102	96	Baik
4	79	102	96	71	89	87	Lusuh

Pada Tabel 4.3 menjelaskan nilai RGB pada mata uang lima ribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang berbeda sehingga didapatkan nilai RGB yang berbeda pada setiap kondisi fisik uang.

Tabel 4.4 Nilai RGB mata uang Rp. 10.000

Mata Uang Rp. 10.000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	109	126	82	93	100	67	Baik
2	109	126	81	95	104	70	Baik
3	103	122	92	95	104	69	Baik Terlipat
4	122	153	113	95	103	69	Lusuh

Pada Tabel 4.4 menjelaskan nilai RGB pada mata uang sepuluh ribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang

berbeda sehingga didapatkan nilai RGB yang berbeda pada setiap kondisi fisik uang.

Tabel 4.5 Nilai RGB mata uang Rp. 20.000

Mata Uang Rp. 20.000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	111	93	84	131	129	129	Baik
2	113	95	86	132	130	130	Baik
3	102	88	83	130	128	128	Baik Terlipat
4	103	89	84	113	95	85	Lusuh

Pada Tabel 4.5 menjelaskan nilai RGB pada mata uang dua puluh ribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang berbeda sehingga didapatkan nilai RGB yang berbeda pada setiap kondisi fisik uang.

Tabel 4.6 Nilai RGB mata uang Rp. 50.000

Mata Uang Rp. 50.000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	94	86	67	146	124	87	Lusuh
2	120	101	73	145	124	87	Baik
3	121	102	74	153	130	90	Baik
4	119	100	72	106	100	75	Baik Terlipat

Pada Tabel 4.6 menjelaskan nilai RGB pada mata uang lima puluh ribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang berbeda sehingga didapatkan nilai RGB yang berbeda pada setiap kondisi fisik uang.

Tabel 4.7 Nilai RGB mata uang Rp. 100.000

Mata Uang Rp. 100.000	Depan			Belakang			Kondisi Uang
	R	G	B	R	G	B	
1	72	118	93	70	80	68	Baik
2	71	118	93	70	78	68	Baik
3	73	118	93	69	79	67	Baik
4	72	119	92	69	80	67	Baik

Pada Tabel 4.7 menjelaskan nilai RGB pada mata uang seratus ribu rupiah, dan dilakukan empat kali percobaan dengan nominal yang sama dengan kondisi yang berbeda sehingga didapatkan nilai RGB yang berbeda pada setiap kondisi fisik uang.

b. Pengiriman Nilai RGB

Pada proses ini tcs32 berfungsi mengambil data nilai RGB dan kemudian dilakukan pengiriman oleh ESP32 melalui jaringan wifi yang terkoneksi dengan *firebase*.



**Gambar 4.1 Proses *Scanning* Mata Uang Rupiah**

```
Connecting to Wi-Fi.
Connected with IP: 192.168.175.122
-----
Connected...
R = 85 G = 79 B = 64
```

**Gambar 4.2 Pengiriman Nilai RGB ke *Firestore***

#### 4.1.2 Pengujian Web Python

Tujuan pengujian ini adalah untuk memprediksi data yang masuk melalui *firebase* dan memperbarui data *output* nominal. Pada pengujian ini dilakukan *training* dan testing pada *dataset* dengan metode klasifikasi *K-Nearest Neighbor*. Perangkat yang diperlukan yaitu *visual studio code*, bahasa pemrograman *python*, *browser* dan koneksi jaringan.

a. Menyimpan *model* ke dalam *disk*

Pada proses ini dataset dibagi menjadi *data training* dan *data testing* dengan perbandingan 70 : 30. Kemudian dilakukan *training* pada *model* dengan algoritma KNN dan disimpan ke dalam file *model.pkl* dengan *library* *Pickle*.

```

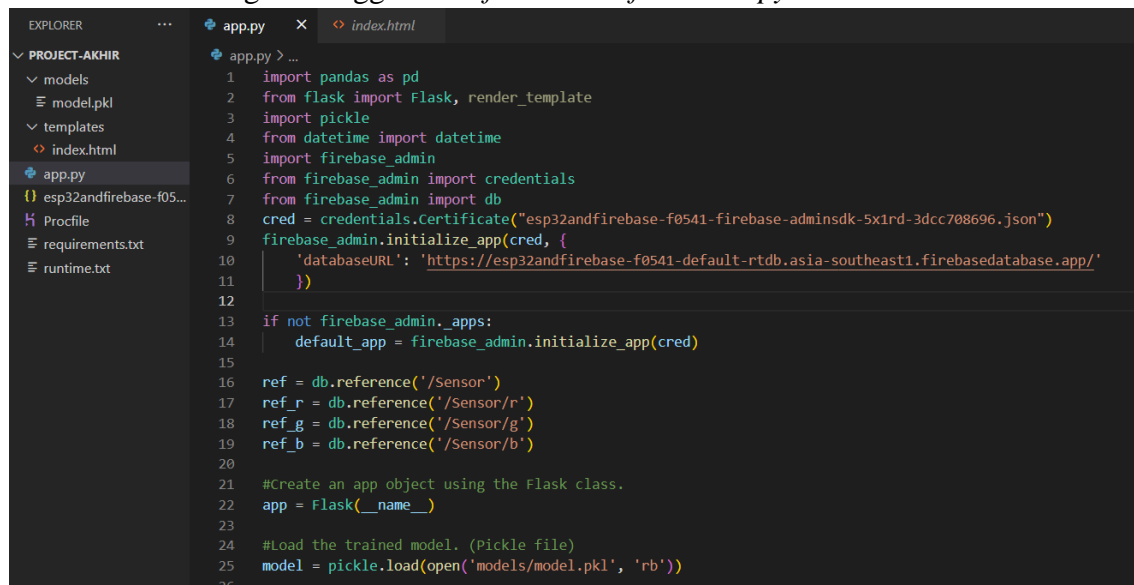
app.py > ...
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.neighbors import KNeighborsClassifier
4 import pickle
5 df = pd.read_csv('dataset3.csv')
6 df = df.sample(frac=1)
7 for col in ['label']:
8     df[col] = df[col].astype('category')
9 x = df[['R', 'G', 'B']]
10 y = df['label']
11 x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=42)
12 # Fit model on training set
13 model = KNeighborsClassifier(n_neighbors=1)
14 model.fit(x_train, y_train)
15 # Save the model to disk
16 filename = 'model.pkl'
17 pickle.dump(model, open(filename, 'wb'))
18
19 # ...
20
21 # load the model from disk
22 loaded_model = pickle.load(open(filename, 'rb'))
23 result = loaded_model.score(x_test, y_test)
24 print(result)

```

**Gambar 4.3 Kode Untuk Menyimpan Model**

b. *Deployment model* ke dalam *web*

Pada proses ini dilakukan *deployment model* yang sudah di-*training* agar *web* mampu melakukan proses klasifikasi secara otomatis. Pembuatan *web* dilakukan dengan menggunakan *framework flask* dari *python*.



```

EXPLORER
PROJECT-AKHIR
  models
    model.pkl
  templates
    index.html
  app.py
  esp32andfirebase-f05...
  Profile
  requirements.txt
  runtime.txt

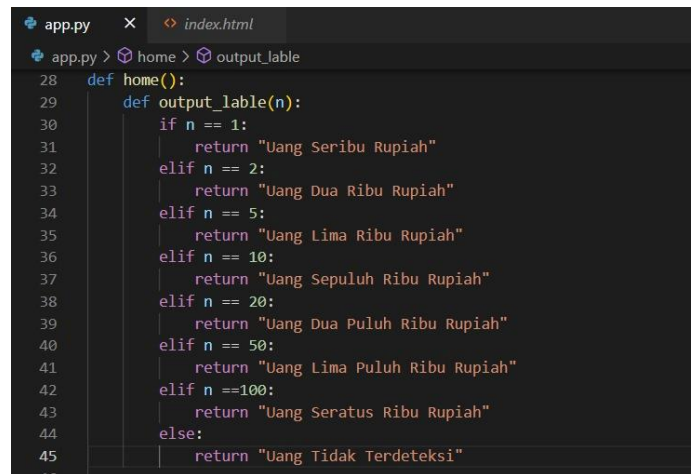
app.py > ...
1 import pandas as pd
2 from flask import Flask, render_template
3 import pickle
4 from datetime import datetime
5 import firebase_admin
6 from firebase_admin import credentials
7 from firebase_admin import db
8 cred = credentials.Certificate("esp32andfirebase-f0541-firebase-adminsdk-5x1rd-3dcc708696.json")
9 firebase_admin.initialize_app(cred, {
10     'databaseURL': 'https://esp32andfirebase-f0541-default-rtdb.asia-southeast1.firebaseio.com/'
11 })
12
13 if not firebase_admin._apps:
14     default_app = firebase_admin.initialize_app(cred)
15
16 ref = db.reference('/Sensor')
17 ref_r = db.reference('/Sensor/r')
18 ref_g = db.reference('/Sensor/g')
19 ref_b = db.reference('/Sensor/b')
20
21 #Create an app object using the Flask class.
22 app = Flask(__name__)
23
24 #Load the trained model. (Pickle file)
25 model = pickle.load(open('models/model.pkl', 'rb'))
26

```

**Gambar 4.4 Kode Framework Web**



Di dalam web dilakukan eksekusi pada file `model.pkl` yang berisi *training*, *testing*, dan *modeling* pada *dataset*. Kemudian dilakukan klasifikasi pada fungsi `output_lable()`.



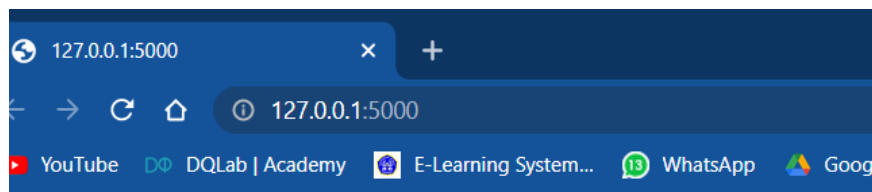
```

app.py x index.html
app.py > home > output_lable
28 def home():
29     def output_lable(n):
30         if n == 1:
31             return "Uang Seribu Rupiah"
32         elif n == 2:
33             return "Uang Dua Ribu Rupiah"
34         elif n == 5:
35             return "Uang Lima Ribu Rupiah"
36         elif n == 10:
37             return "Uang Sepuluh Ribu Rupiah"
38         elif n == 20:
39             return "Uang Dua Puluh Ribu Rupiah"
40         elif n == 50:
41             return "Uang Lima Puluh Ribu Rupiah"
42         elif n == 100:
43             return "Uang Seratus Ribu Rupiah"
44         else:
45             return "Uang Tidak Terdeteksi"

```

**Gambar 4.5 Kode `output_lable()`**

Selanjutnya, *web* akan melakukan *reload* setiap satu detik untuk menjalankan ulang program agar setiap data nilai RGB terbaru yang masuk dapat dilakukan proses klasifikasi secara otomatis.



he current time is 21 August 2022 16:26:12.

**Gambar 4.6 Tampilan Web**

#### 4.1.3 Pengujian Firebase

Tujuan dari pengujian untuk mengetahui bahwa *realtime database* dari *firebase* telah menerima data terbaru. Pada pengujian ini diperlukan koneksi jaringan dan *browser*.

##### a. Pengujian pada data *Sensor*

Pada database di *firebase* terdapat semaca *folder* yang bernama *Sensor*, di dalamnya terdapat data *r*, *g*, dan *b* dengan tipe data *integer*. Pada folder ini

menerima data nilai RGB dari sensor warna TCS230 yang akan diperbarui jika ada data yang dikirim.

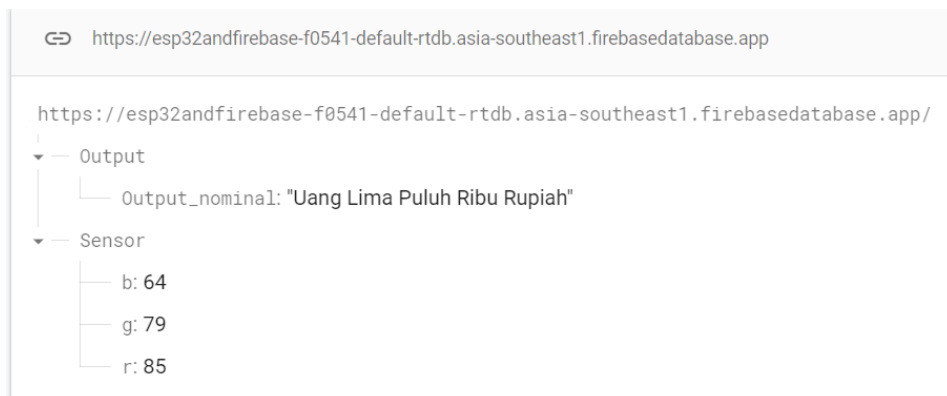


**Gambar 4.7 Tampilan Data *Sensor* pada *Firebase***

Pada gambar 4.7 terdapat nilai r, g, b yang sudah diperbarui dengan data nilai RGB terbaru yaitu 85, 79, 64.

b. Pengujian pada data *Output*

Di dalam folder *Output* terdapat data *Output\_nominal* yang akan diperbarui oleh web dengan keluaran berupa *string*.



**Gambar 4.8 Tampilan data *Output* pada *Firebase***

Berdasarkan gambar 4.8 dengan nilai  $r = 85$ ,  $g = 79$ ,  $b = 64$  mendapatkan nilai *Output\_nominal* = “Uang Seribu Rupiah”.

#### 4.1.4 Pengujian *DFPlayer* dan *Speaker*

Pengujian ini dilakukan untuk mengetahui apakah *speaker* dapat berfungsi dan *DFPlayer* memutar suara sesuai dengan *output*. Perangkat yang diperlukan yaitu, *DFPlayer*, *sd card*, dan *speaker mini*.

```
DFRobot DFPlayer Mini Demo
Initializing DFPlayer ... (May take 3-5 seconds)
DFPlayer Mini online.
```

**Gambar 4.9 Tampilan Koneksi *DFPlayer***

**Tabel 4.8 Hasil Pengujian *Speaker***

No	Nominal <i>Output</i>	<i>DFPlayer</i> ( <i>Connected</i> )
1	Uang Seribu Rupiah	Memainkan suara
2	Uang Dua Ribu Rupiah	Memainkan suara
3	Uang Lima Ribu Rupiah	Memainkan suara
4	Uang Sepuluh Ribu Rupiah	Memainkan suara
5	Uang Dua Puluh Ribu Rupiah	Memainkan suara
6	Uang Lima Puluh Ribu Rupiah	Memainkan suara
7	Uang Seratus Ribu Rupiah	Memainkan suara
8	Uang Tidak Terdeteksi	Memainkan suara

## 4.2 Pembahasan Sistem

Hasil dari pengujian yang telah dilakukan pada pengambilan data nilai RGB mata uang rupiah yang dikirimkan ke dalam *firebase* kemudian web akan mengambil data RGB tersebut dan dilakukan proses prediksi untuk mengetahui nominal mata uang yang diuji. Selanjutnya, hasil prediksi tersebut akan dikirim ke *firebase* untuk memperbarui data *Output\_nominal* dalam bentuk *string*. Tetapi didapatkan hasil nilai RGB yang berbeda pada setiap mata uang karena gradasi warna yang berbeda-beda. Maka dari itu, dengan menggunakan algoritma KNN (*K-Nearest Neighbor*) dapat menentukan secara akurat mata uang yang dibaca berdasarkan hasil pembelajaran objek terhadap tetangga terdekat. Pengujian ini dilakukan pada uang Rp. 100.000 (seratus ribu rupiah), Rp. 50.000 (lima puluh ribu rupiah), Rp. 20.000 (dua puluh ribu rupiah) dan Rp. 10.000 (sepuluh ribu rupiah), Rp. 5.000 (lima ribu rupiah), Rp. 2.000 (dua ribu rupiah), dan Rp. 1.000 (seribu

rupiah) dengan menggunakan 4 lembar setiap nominal mendapatkan nilai algoritma sebesar 90%.

Tabel 4.9 Perbandingan antar Algoritma dan Akurasi tiap Mata Uang

Uang	Nilai			Algoritma ( terbaca )		
	R	G	B	Random Forest	Gradient Boost	K-Nearest Neighbour
Rp 1,000	106	117	108	seribu rupiah (83%)	seribu rupiah (90%)	seribu rupiah (90%)
Rp 2,000	91	94	78	dua ribu rupiah (82%)	dua ribu rupiah (80%)	dua ribu rupiah (87%)
Rp 5,000	86	113	94	seratus ribu rupiah (56%)	lima ribu rupiah (85%)	lima ribu rupiah (92%)
Rp 10,000	115	141	103	seratus ribu rupiah (50%)	lima ribu rupiah (50%)	sepuluh ribu rupiah (91%)
Rp 20,000	126	107	109	dua ribu rupiah (62%)	dua ribu rupiah (68%)	dua puluh ribu rupiah (91%)
Rp 50,000	128	103	74	lima puluh ribu rupiah (78%)	lima puluh ribu rupiah (84%)	lima puluh ribu rupiah (90%)
Rp 100,000	72	105	81	sepuluh ribu rupiah (51%)	seratus ribu rupiah (87%)	seratus ribu rupiah (92%)

Selanjutnya alat akan membaca nilai *Output\_nominal* di firebase, kemudian speaker dan DFPlayer akan memutar suara sesuai dengan data yang ada pada *Output\_nominal* di firebase

Setelah seluruh sistem berhasil dibuat, maka dilakukan pengujian sistem kepada pengguna. Pengujian ini dilakukan pada tanggal 28 Agustus 2022 kepada penyandang tunanetra di komunitas pertuni (persatuan tunanetra Indonesia) cabang semarang. Pertanyaan yang diajukan kepada 2 responden dapat dilihat pada Tabel 3.3 Daftar Pertanyaan. Panduan perhitungan skor kuesioner Alat Deteksi Mata Uang Rupiah dengan prediksi *Machine Learning* untuk Tunanetra dari 2 responden yang berbeda, seperti yang ditunjukkan pada Tabel 3.4. Untuk hasil perhitungan total kepuasan pengguna didasarkan pada tabel indikator penilaian kepuasan pengguna pada Tabel 3.5. Berikut adalah hasil penilaian responden mengenai Alat Deteksi Mata Uang Rupiah dengan prediksi *Machine Learning* untuk Tunanetra dari 2 responden yang berbeda, seperti ditunjukkan pada Tabel 4.6.

Tabel 4.10 Hasil Kuesioner Responden

No	Nama	Daftar Pertanyaan			Total Nilai
		1	2	3	
1	Triyani	4	2	5	10
2	Bapak Sakur	4	3	4	11
<b>TOTAL</b>					21

Berikut adalah perhitungan tingkat kepuasan pengguna terhadap Alat Deteksi Mata Uang Rupiah dengan prediksi *Machine Learning* untuk Tunanetra:

Jumlah nilai maksimal untuk indikator penilaian =  $5 \times 3 = 15$

Jumlah responden yang mengisi kuesioner = 2

Jumlah kepuasan maksimal =  $2 \times 15 = 30$

Presntase kepuasan pengguna (%) = Total kepuasan pengguna / Total kepuasan maksimum pengguna

$$\text{Presntase kepuasan pengguna (\%)} = \frac{\text{Total kepuasan pengguna}}{\text{Total kepuasan maksimum pengguna}} \times 100\%$$

$$= \frac{21}{30} \times 100\%$$

$$= 70\%$$

Berdasarkan hasil data Tabel 4.10 dapat dilihat bahwa tingkat kepuasan pengguna diperoleh nilai 70%. Kategori nilai yang didapat cukup memuaskan, namun perlu pengembangan yang lebih jauh agar mendapat tingkat kepuasan yang lebih tinggi sehingga alat ini dapat digunakan sesuai dengan kebutuhan *user*.

## **BAB V KESIMPULAN**

### **5.1 Kesimpulan**

Berdasarkan pengambilan data, hasil pengujian dan analisis yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

1. Mesin dapat bekerja secara baik dan optimal.
2. Mesin dapat membaca nominal setiap mata uang.
3. Mesin dapat mengeluarkan suara untuk mempermudah tunanetra.
4. Terciptanya sistem baru menggunakan *machine learning*

### **5.2 Saran**

Perancangan ini masih sangat perlu untuk dikembangkan, untuk mendapat hasil yang lebih baik, perlu dilakukan penelitian lebih lanjut pada alat ini, baik dari segi cara kerja sistem maupun struktur alat sehingga mampu mengatasi batasan penelitian alat ini, berikut adalah saran untuk pengembangan lebih lanjut pada alat ini:

1. Sumber daya lebih baik menggunakan baterai agar dapat bekerja secara *portable*.
2. Dari segi dimensi untuk alat ini bisa lebih di perkecil agar mudah dibawa kemana mana dan tidak memakan tempat bagi para penyandang tuna netra.
3. Struktur alat dapat dimodifikasi sehingga alat lebih mudah digunakan bagi penyandang tunanetra.
4. Menggunakan komponen yang lebih kuat agar tidak mudah rusak.
5. Data harus selalu terupdate jika ada keluaran emisi uang baru.

## DAFTAR PUSTAKA

- [1] Wikipedia.2022."Uang",<https://id.wikipedia.org/wiki/Uang>, diakses pada 14 Agustus 2022 pukul 16.38
- [2] Undip, E-Jurnal.2018."BAB\_II.pdf"
- [3] Erza Q-POP.2018."BAB II.pdf"
- [4] Dickson Kho.2020."Pengertian Mikrokontroler (Microcontroller) dan Strukturnya", <https://teknikelektronika.com/pengertian-mikrokontroler-microcontroller-struktur-mikrokontroler/>, diakses pada 14 Agustus 2022 pukul 16.53
- [5] Pradana, Restu Adi.2019."Mikrokontroler ESP32, apa itu? (bagian 1) #Microcontrollers101", <https://timur.ilearning.me/2019/04/19/mikrokontroler-esp32-apa-itu/>, diakses pada 14 Agustus 2022 pukul 16.55
- [6] Zahra.2017."3\_133310002\_BAB II.pdf"
- [7] Algoritma.2022."MENGENAL PANDAS DALAM PYTHON",<https://algorit.ma/blog/library-pandas-python-2022/>, diakses pada 14 Agustus 2022 pukul 16.59
- [8] Academy.2021."Machine Learning dengan Scikit learn python",<https://www.bisa.ai/course/detail/MzU3/1>, diakses pada 14 Agustus 2022 pukul 17.02
- [9] DQLab.2021."Mengenal Flask, Library Machine Learning Python Idaman Developer",<https://dqlab.id/mengenal-flask-library-machine-learning-python-idaman-developer> diakses pada 14 Agustus pukul 17.04
- [10] Algonz D.B. Raharja.2022."Machine Learning: Pengertian, Cara Kerja, dan 3 Metodenya!",<https://www.ekrut.com/media/apa-itu-machine-learning>, diakses pada 14 Agustus 2022 pukul 17.06
- [11] Yustanti, Wiyli.2012."Algoritma K-Nearest Neighbour untuk Memprediksi Harga Jual Tanah"
- [12] Bliaudio.2020."Fungsi Speaker",[https://www.bliaudio.com/index.php?route=information/blogger&logger\\_id=70](https://www.bliaudio.com/index.php?route=information/blogger&logger_id=70), diakses pada 14 Agustus pukul 17.09

- [13] Icloudhost.2021."Apa itu Firebase? Pengertian dan Cara Kerjanya",<https://idcloudhost.com/panduan/apa-itu-firebase/>, diakses pada 14 Agustus pukul 17.10



## LAMPIRAN

### Lampiran 1 : Dokumentasi

#### a. Pengujian user



## Lampiran 2 : Kode Program Arduino

### a. A\_package

```
#include <WiFi.h>
#include <FirebaseESP32.h>

#define FIREBASE_HOST "https://esp32andfirebase-f0541-default-rtdb.asia-southeast1.firebaseio.com/"
#define FIREBASE_AUTH "sHHVED50JYjHhV3mwJUg6Y3nWS2D34veVVUHpzMQ"
#define WIFI_SSID "Sen"
#define WIFI_PASSWORD "okeokeoke"

//Define FirebaseESP32 data object
FirebaseData firebaseData;
FirebaseData nomData;
FirebaseJson json;

// TCS230 or TCS3200 pins wiring to Arduino
#define S0 4
#define S1 5
#define S2 18
#define S3 19
#define sensorOut 21

// Stores frequency read by the photodiodes
int redFrequency = 0;
int greenFrequency = 0;
int blueFrequency = 0;
```

### b. B\_main

```
#include <WiFi.h>
#include <FirebaseESP32.h>

#define FIREBASE_HOST "https://esp32andfirebase-f0541-default-rtdb.asia-southeast1.firebaseio.com/"
#define FIREBASE_AUTH "sHHVED50JYjHhV3mwJUg6Y3nWS2D34veVVUHpzMQ"
#define WIFI_SSID "Sen"
#define WIFI_PASSWORD "okeokeoke"

//Define FirebaseESP32 data object
FirebaseData firebaseData;
FirebaseData nomData;
FirebaseJson json;

// TCS230 or TCS3200 pins wiring to Arduino
#define S0 4
#define S1 5
#define S2 18
#define S3 19
#define sensorOut 21

// Stores frequency read by the photodiodes
int redFrequency = 0;
int greenFrequency = 0;
int blueFrequency = 0;
```

## c. C\_warna\_data

```

void setWarna() {
  // Setting the outputs
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);

  // Setting the sensorOut as an input
  pinMode(sensorOut, INPUT);

  // Setting frequency scaling to 20%
  digitalWrite(S0,HIGH);
  digitalWrite(S1,LOW);

  // Begins serial communication
  Serial.begin(115200);
}
void loopWarna() {
  // Setting RED (R) filtered photodiodes to be read
  digitalWrite(S2,LOW);
  digitalWrite(S3,LOW);

  // Reading the output frequency
  redFrequency = pulseIn(sensorOut, LOW);

  // Printing the RED (R) value
  Serial.print("R = ");
  Serial.print(redFrequency);
  delay(1000);

  // Setting GREEN (G) filtered photodiodes to be read
  digitalWrite(S2,HIGH);
  digitalWrite(S3,HIGH);

  // Reading the output frequency
  greenFrequency = pulseIn(sensorOut, LOW);

  // Printing the GREEN (G) value
  Serial.print(" G = ");
  Serial.print(greenFrequency);
  delay(1000);

  // Setting BLUE (B) filtered photodiodes to be read
  digitalWrite(S2,LOW);
  digitalWrite(S3,HIGH);

  // Reading the output frequency
  blueFrequency = pulseIn(sensorOut, LOW);

  // Printing the BLUE (B) value
  Serial.print(" B = ");
  Serial.println(blueFrequency);
  delay(1000);
}

```

## d. D\_mp3\_play

```

void setSuara()
{
  Serial.begin(115200);
  Serial2.begin(9600);
  delay(100);

  Serial.println();
  Serial.println(F("DFRobot DFPlayer Mini Demo"));
  Serial.println(F("Initializing DFPlayer ... (May take 3~5 seconds)"));

  if (!mp3.begin(Serial2)) { //Use softwareSerial to communicate with mp3.
    Serial.println(F("Unable to begin:"));
    Serial.println(F("1.Please recheck the connection!"));
    Serial.println(F("2.Please insert the SD card!"));
    while (true);
  }
  Serial.println(F("DFPlayer Mini online."));
  mp3.setTimeout(500); //Set serial communication time out 500ms
  mp3.volume(30); //Set volume value (0~30).
  // delay(1000);
  // mp3.play(98); //Play the first mp3
  delay(1000);
}

void loopSuara()
{
  mp3.play ();
  delay (6000);
}

```

### Lampiran 3 : Kode Program *Machine Learning*

```
a. app.py
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
import pickle
df = pd.read_csv('dataset3.csv')
df = df.sample(frac=1)
for col in ['label']:
    df[col] = df[col].astype('category')
x = df[["R", "G", "B"]]
y = df["label"]
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3,
random_state=42)
# Fit model on training set
model = KNeighborsClassifier(n_neighbors=2)
model.fit(x_train,y_train)
# Save the model to disk
filename = 'model.pkl'
pickle.dump(model, open(filename, 'wb'))

# ...
# load the model from disk
loaded_model = pickle.load(open(filename, 'rb'))
result = loaded_model.score(x_test, y_test)
print(result)
```

#### Lampiran 4 : Kode Program Web

```
a. app.py
import pandas as pd
from flask import Flask, render_template
import pickle
from datetime import datetime
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
cred = credentials.Certificate("esp32andfirebase-f0541-firebase-adminsdk-
5x1rd-3dcc708696.json")
firebase_admin.initialize_app(cred, {
    'databaseURL': 'https://esp32andfirebase-f0541-default-rtdb.asia-
southeast1.firebaseio.com/app/'
})

if not firebase_admin._apps:
    default_app = firebase_admin.initialize_app(cred)

ref = db.reference('/Sensor')
ref_r = db.reference('/Sensor/r')
ref_g = db.reference('/Sensor/g')
ref_b = db.reference('/Sensor/b')

#Create an app object using the Flask class.
app = Flask(__name__)

#Load the trained model. (Pickle file)
model = pickle.load(open('models/model.pkl', 'rb'))

@app.route('/')
def home():
    def output_label(n):
        if n == 1:
            return "Uang Seribu Rupiah"
        elif n == 2:
            return "Uang Dua Ribu Rupiah"
        elif n == 5:
            return "Uang Lima Ribu Rupiah"
        elif n == 10:
            return "Uang Sepuluh Ribu Rupiah"
        elif n == 20:
            return "Uang Dua Puluh Ribu Rupiah"
        elif n == 50:
```

```

        return "Uang Lima Puluh Ribu Rupiah"
    elif n == 100:
        return "Uang Seratus Ribu Rupiah"
    else:
        return "Uang tidak terdeteksi"

ref_new = db.reference('Output')
nom_ref_new = ref_new.child('Output_nominal')

def testing(r, g, b):

    features = pd.DataFrame([[r,g,b]], columns=['R', 'G', 'B'])
    prediction = model.predict(features)

    output = str(output_lable(prediction[0]))
    ref_new.update({
        'Output_nominal': output
    })
    print(output)

testing(ref_r.get(), ref_g.get(), ref_b.get())
return ""
<meta http-equiv="refresh" content="1" />
<br>The current time is {}.format(datetime.strftime(datetime.now(), "%d
%B %Y %X"))

@app.route('/predict', methods=['POST'])
def predict():
    def output_lable(n):
        if n == 1:
            return "Uang Seribu Rupiah"
        elif n == 2:
            return "Uang Dua Ribu Rupiah"
        elif n == 5:
            return "Uang Lima Ribu Rupiah"
        elif n == 10:
            return "Uang Sepuluh Ribu Rupiah"
        elif n == 20:
            return "Uang Dua Puluh Ribu Rupiah"
        elif n == 50:
            return "Uang Lima Puluh Ribu Rupiah"
        elif n == 1000 :
            return "Uang Seratus Ribu Rupiah"

    ref_new = db.reference('Output')
    nom_ref_new = ref_new.child('Output_nominal')

```

```

def testing(r, g, b):

    features = pd.DataFrame([[r,g,b]], columns=['R', 'G', 'B'])
    prediction = model.predict(features)

    output = str(output_label(prediction[0]))
    ref_new.update({
        'Output_nominal': output
    })
    print(output)

testing(ref_r.get(), ref_g.get(), ref_b.get())

if __name__ == "__main__":
    app.run()

```

b. index.html

```

<!DOCTYPE html>
<html >
<head>
    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>MONEY NOMINAL PREDICITON</title>

</head>

<body style="background: rgb(36, 179, 153);">
    <div class="login">
        <h1 style="color:black;">money nominal prediction</h1>

        <br>
        <br>
        <h4 style="color:black;">
            {{ prediction_text }} </h4>

    </div>

</body>
</html>

```



c. profile  
web: gunicorn app:app

d. requirement.txt

Flask==1.1.2

pandas==1.1.3

numpy==1.19.5

requests==2.24.0

pickle4

gunicorn

scikit-learn==0.23.0

e. runtime.txt

python-3.7.10

**Lampiran 5 : Dataset**

R	G	B	label
106	117	108	1
115	124	112	1
91	94	78	2
83	101	94	5
108	111	98	2
90	92	74	1
86	113	94	5
115	138	98	10
94	96	75	1
94	97	75	1
26	28	25	2
26	28	24	2
26	27	25	2
26	27	24	2
25	28	25	2
25	28	24	2
111	94	71	50
25	27	25	2
25	27	24	2
24	28	25	2
24	28	24	2
24	27	25	2
24	27	24	2
32	36	32	2
32	36	33	2
32	35	32	2
32	35	33	2
31	36	32	2
31	36	33	2
31	35	32	2
31	35	33	2
24	27	25	2
34	33	27	2
35	33	27	2
22	24	24	2
23	24	24	2

33	35	30	2
33	35	28	2
33	34	30	2
33	34	28	2
33	33	30	2
33	33	28	2
33	32	30	2
33	32	28	2
32	35	30	2
32	35	28	2
32	34	30	2
32	34	28	2
32	33	30	2
32	33	28	2
32	32	30	2
32	32	28	2
31	35	30	2
31	35	28	2
31	34	30	2
31	34	28	2
31	33	30	2
31	33	28	2
31	32	30	2
31	32	28	2
30	35	30	2
30	35	28	2
30	34	30	2
30	34	28	2
30	33	30	2
30	33	28	2
30	32	30	2
30	32	28	2
35	36	30	2
35	37	30	2
32	37	34	2
20	23	22	1
20	22	22	1
27	32	30	1
27	32	30	1

27	31	30	1
90	87	72	50
91	88	73	50
89	86	71	50
27	31	31	1
28	32	30	1
28	32	30	1
104	92	85	20
94	91	101	20
95	92	102	20
93	90	100	20
97	89	85	20
96	88	84	20
98	90	86	20
28	31	30	1
28	31	31	1
28	30	26	1
28	30	27	1
86	79	90	20
87	80	91	20
85	78	89	20
121	100	90	20
125	158	116	10
124	157	115	10
126	159	117	10
80	103	97	5
81	104	98	5
79	102	96	5
71	89	87	5
72	90	88	5
70	88	86	5
71	83	79	5
70	82	78	5
72	84	80	5
85	111	93	5
86	112	94	5
84	110	92	5
128	103	74	50
129	104	75	50

127	102	73	50
122	101	91	20
115	141	103	10
116	142	104	10
114	140	102	10
120	99	89	20
124	108	110	20
127	108	110	20
128	109	111	20
126	107	109	20
28	31	26	1
109	92	82	20
110	93	83	20
108	91	81	20
28	31	27	1
29	30	26	1
29	30	27	1
29	31	26	1
29	31	27	1
26	30	29	1
116	114	108	20
115	113	107	20
117	115	109	20
72	104	80	100
72	105	80	100
27	30	29	1
31	36	34	1
70	79	68	100
70	80	68	100
70	78	68	100
69	79	67	100
69	80	67	100
69	78	67	100
71	79	69	100
71	80	69	100
71	78	69	100
31	36	33	1
31	35	34	1
95	84	77	20

108	119	110	1
85	77	81	20
86	78	82	20
84	76	80	20
105	109	97	2
104	108	98	2
106	110	96	2
85	92	78	2
94	96	74	1
96	85	78	20
94	76	76	20
72	100	78	100
72	106	81	100
95	87	68	50
96	88	69	50
94	86	67	50
120	101	73	50
121	102	74	50
119	100	72	50
72	105	81	100
72	107	81	100
71	106	80	100
71	105	80	100
131	129	129	20
132	130	130	20
130	128	128	20
113	95	85	20
111	93	84	20
113	95	86	20
102	88	83	20
103	89	84	20
101	87	82	20
71	107	80	100
114	98	95	20
113	97	94	20
115	99	96	20
73	106	82	100
109	92	69	50
110	93	70	50

108	91	68	50
73	105	82	100
73	107	82	100
104	110	118	20
105	111	119	20
103	109	120	20
104	111	119	20
104	109	120	20
80	75	80	20
81	76	81	20
96	84	78	20
95	85	79	20
126	120	112	20
127	121	113	20
125	119	111	20
97	83	77	20
79	74	79	20
85	76	80	20
136	128	129	20
137	128	129	20
135	127	128	20
135	129	130	20
84	76	80	20
86	77	79	20
31	35	33	1
106	107	116	20
105	106	115	20
107	108	117	20
32	36	34	1
121	103	99	20
120	102	100	20
122	104	98	20
111	94	83	20
112	93	84	20
113	95	82	20
32	36	33	1
32	35	34	1
91	126	121	5
90	125	120	5

92	127	122	5
32	35	33	1
116	119	118	20
117	120	119	20
115	118	117	20
32	37	35	1
33	35	27	1
126	124	120	20
126	123	120	20
125	124	121	20
127	122	119	20
125	123	120	20
33	35	28	1
33	34	27	1
33	34	28	1
21	22	18	1
27	29	23	1
28	29	23	1
27	28	22	1
22	23	19	1
27	29	23	1
69	100	84	100
80	134	106	100
80	135	106	100
80	136	106	100
79	134	105	100
79	135	105	100
79	136	105	100
81	134	107	100
81	135	107	100
81	136	107	100
27	29	22	1
28	29	23	1
28	29	22	1
61	83	70	100
61	82	70	100
61	84	70	100
62	83	71	100
62	82	71	100



62	84	71	100
63	83	69	100
63	82	69	100
63	84	69	100
18	22	19	5
17	22	19	5
30	38	37	5
30	37	37	5
30	37	36	5
29	37	36	5
77	106	85	100
20	27	23	5
72	87	77	100
71	87	77	100
73	87	77	100
72	86	76	100
71	86	76	100
73	86	76	100
72	88	78	100
71	88	78	100
73	88	78	100
21	26	23	5
20	26	24	5
20	26	23	5
21	27	23	5
70	86	84	5
87	90	73	1
93	104	104	1
92	103	103	1
93	104	101	1
93	104	102	1
78	99	84	5
82	84	68	1
81	84	67	1
86	90	76	2
87	90	76	2
85	90	76	2
86	91	76	2
119	100	73	50

118	99	72	50
120	101	74	50
87	91	76	2
85	91	76	2
86	89	76	2
87	89	76	2
85	89	78	2
86	90	78	2
87	90	77	2
72	118	93	100
71	118	93	100
73	118	93	100
72	119	92	100
71	119	92	100
73	119	92	100
72	117	94	100
71	117	94	100
73	117	94	100
85	90	77	2
85	92	82	2
85	86	68	1
83	92	82	2
97	107	97	2
97	107	96	2
75	91	78	100
74	90	77	100
76	92	79	100
83	86	71	2
84	86	71	2
85	86	71	2
89	99	91	2
88	91	77	2
88	91	77	2
103	119	79	10
103	118	79	10
93	100	67	10
95	104	70	10
95	104	69	10
109	126	82	10

109	126	81	10
103	122	92	10
122	153	113	10
121	153	113	10
112	103	107	20
113	104	108	20
111	102	106	20
108	126	81	10
95	103	69	10
112	136	96	10
129	103	75	50
63	90	75	100
62	90	75	100
64	90	75	100
63	89	74	100
62	89	74	100
64	89	74	100
110	146	114	10
111	147	115	10
109	145	113	10
63	91	76	100
96	131	115	5
95	130	114	5
97	132	116	5
84	107	103	5
83	106	102	5
85	108	104	5
62	91	76	100
64	91	76	100
104	127	101	10
105	128	102	10
74	84	79	5
75	85	80	5
73	83	78	5
103	126	100	10
89	112	85	10
89	111	85	10
104	122	86	10
106	122	86	10

107	122	87	10
107	123	87	10
115	134	92	10
110	98	72	50
146	124	87	50
145	124	87	50
153	130	90	50
106	100	75	50
117	98	72	50
117	98	71	50
94	86	68	50
117	99	72	50
115	92	65	50
85	78	63	50
117	93	64	50
116	92	64	50
92	78	58	50
92	77	58	50
122	104	79	50
122	105	79	50
96	90	74	50
97	90	74	50
96	90	73	50
89	98	95	1
108	117	108	1
108	118	108	1
108	118	107	1
89	91	73	1
86	89	71	1
99	92	74	50
86	89	70	1
98	100	86	2
98	99	85	2
98	99	86	2
69	82	76	5
69	79	75	5
108	111	99	2
88	117	98	5
80	102	86	5

70	82	76	5
79	102	96	5
74	92	89	5
120	125	109	2
118	123	107	2
89	92	76	2
123	105	79	50
80	108	105	5
79	107	104	5
81	109	106	5