

## Implementasi Algoritma Genetika Pada Perancangan Aplikasi Android Untuk Memprediksi Buta Warna

Meirina Suci Ridha<sup>1</sup>, Hani Harafani<sup>2</sup>

<sup>1</sup>Teknik Informatika, STMIK Nusa Mandiri Jakarta  
e-mail: meirinasuciridha@gmail.com

<sup>2</sup> Teknik Informatika, STMIK Nusa Mandiri Jakarta  
e-mail: hani.hhf@nusamandiri.ac.id

---

**Cara Sitasi:** Ridha, M. S., & Harafani, H. (2019). Implementasi Algoritma Genetika Pada Perancangan Aplikasi Android Untuk Memprediksi Buta Warna. *Jurnal Teknik Komputer*, 77-86. doi:10.31294/jtk.v4i2

---

**Abstract** *Color blindness is one of the decreasing diseases which is very difficult to determine its deterioration, whether a family member will suffer from color blindness or not, especially with the prediction of color blindness which has been using manual methods by calculating the inheritance formula. Genetic Algorithms have advantages over other traditional optimization algorithms. To implement a computerized method in predicting color blindness that can be used by many people, it is necessary to have a user-friendly operating system like the Android operating system. The test results show that the implementation of genetic algorithms in applications to predict color blindness produces more optimal predictions and their application to Android makes the application a user friendly application.*

**Keywords:** *Heredity, Color Blindness, Genetic Algorithms, Android Application, Color Blindness Predictions*

### PENDAHULUAN

Buta warna merupakan salah satu penyakit menurun yang menjadi masalah yang cukup serius dalam silsilah keluarga (Ervan dan Mulyanto, 2015:145).

Sementara itu, teknik yang dilakukan secara manual sudah tidak lagi efektif digunakan untuk mendiagnosa, maka menjadi keputusan yang tepat untuk mengembangkan penelitian dengan metode komputerisasi dalam mendiagnosa penyakit (Neshat, Mehdi & Yaghoobi dalam Fatmawati, 2016).

Beberapa peneliti telah melakukan penelitian untuk memprediksi penyakit dengan metode komputerisasi seperti CART untuk memprediksi Kanker Serviks (Praningki, & Budi, 2017), algoritma C4.5 untuk memprediksi Diabetes Mellitus (Andriani, 2013), dan Algoritma Genetika untuk memprediksi buta warna (Ervan, & Mulyanto, 2015).

Algoritma C4.5 diperkenalkan oleh J. Ross Quinlan yang merupakan perkembangan dari algoritma ID3 sehingga memiliki beberapa kelebihan meliputi metode untuk menangani *numeric attributes*, *missing values*, *noisy data*, dan aturan yang menghasilkan rules dari *trees* namun algoritma C4.5 memiliki kelemahan dalam pembacaan data yang cukup besar (Andriani, 2013).

Algoritma CART memiliki kelebihan yaitu

menentukan prediksi yang mampu menangani himpunan data yang mempunyai jumlah besar dan variabel yang sangat banyak (Puteri, Maharani, & Suliyo, 2013).

Sedangkan kelemahan dari Algoritma CART adalah hasil akhirnya tidak didasarkan model probabilistik. Tidak ada tingkat probabilitas dari selang kepercayaan yang berhubungan dengan dugaan yang didapat dari pohon Algoritma CART untuk pengelompokan data baru. (Darmawan, Yuniarti, & Nasution, 2017)

Algoritma Genetika memiliki kelebihan dibandingkan algoritma optimasi tradisional lainnya, dua diantaranya yaitu kemampuan untuk menangani permasalahan kompleks dan parallel. (Harafani, 2018).

Untuk menerapkan metode komputerisasi dalam melakukan prediksi resiko buta warna yang dapat digunakan oleh banyak orang, maka diperlukan adanya pemilihan sistem operasi yang *user friendly* seperti sistem operasi Android.

Sistem operasi Android yang merupakan salah satu sistem operasi yang sedang booming saat ini menawarkan kemudahan bagi penggunaanya dalam memperoleh informasi (Nurajizah & Saputra, 2018). Selain informasi, banyak juga aplikasi yang sengaja dibuat berhubungan dengan kesehatan seperti untuk mendiagnosa penyakit kulit dan kelamin yang

dilakukan oleh Kemala, Irawan, dan Nasrun pada tahun 2015, penyakit mata oleh Aditiawarman, dkk pada tahun 2017 dan pneumonia pada anak balita oleh Mujahidin dan Pribadi pada tahun 2017.

Berdasarkan penelitian-penelitian yang sudah dilakukan oleh peneliti-peneliti sebelumnya, maka aplikasi yang berjalan diatas platform android kami rasa dapat menjadi terobosan untuk melakukan prediksi terhadap buta warna dengan mengimplementasikan algoritma genetika agar dapat menghasilkan aplikasi yang *user friendly* dengan hasil prediksi yang lebih optimal.

## METODOLOGI PENELITIAN

### 1. Metode Pengumpulan Data

Pada penelitian ini data yang digunakan adalah data percobaan yang diambil dari penelitian (Ervan dan Mulyanto, 2015:145) dalam penelitiannya yang berjudul Deteksi Risiko Penderita Buta Warna Menurun Berbasis Pohon Keluarga dengan Algoritma Genetika. Data yang digunakan merupakan data percobaan dan hasil percobaan pada 21 sample orang tua yang normal, buta warna, buta warna sebagian, dan pembawa buta warna.

Tabel 1. Data percobaan

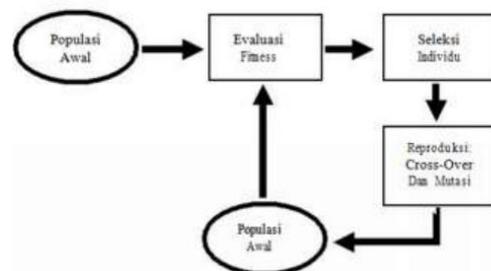
No	Jumlah Anak		Induk		Risiko		Selisish	
	P	L	I	A	$Y_i$	$P_r$	Angka	Dlm%
1	1	2	4	2	0	0	0	0%
2	0	4	4	2	0	0	0	0%
3	2	1	3	5	0	0	0	0%
4	0	5	2	4	0	0	0	0%
5	4	4	4	2	0	1	1	100%
6	5	4	4	2	4	2	2	50%
7	1	3	4	2	1	0	1	100%
8	1	3	4	2	1	0	1	100%
9	0	3	4	2	0	0	0	0%
10	1	2	4	2	1	0	1	100%
11	2	1	4	2	0	0	0	0%
12	2	2	4	2	1	0	1	100%
13	2	1	4	2	0	0	0	0%
14	2	2	4	2	1	0	1	100%
15	1	1	4	2	1	0	1	100%
16	0	2	4	2	0	0	0	0%
17	0	1	4	2	0	0	0	0%
18	0	4	3	5	0	0	0	0%
19	2	2	3	5	0	0	0	0%
20	0	4	1	2	0	0	0	0%
21	3	3	1	2	0	0	0	0%
<b>Rata-Rata Kesalahan</b>								35,7%

Sumber : (Ervan dan Mulyanto, 2015:145)

## 2. Algoritma Genetika

Algoritma genetika menurut Gorunescu dalam (Harafani & Wahono, 2015) memiliki tiga operator genetic utama yaitu *crossover* (proses penukaran kromosom), mutasi (proses penggantian salah satu solusi untuk meningkatkan keragaman populasi), seleksi (penggunaan solusi dengan nilai fitness yang tinggi untuk lulus ke generasi berikutnya). Terdapat langkah-langkah yang sering dilakukan untuk menyelesaikan permasalahan-permasalahan dalam optimasi diantaranya:

- a) Inisialisasi Populasi awal
- b) Evaluasi
- c) Seleksi
- d) Proses Penyilangan kromosom (*Crossover*)
- e) Evaluasi populasi baru (Mutasi)
- f) Selama syarat belum terpenuhi ulangi langkah 3.



Sumber : (Evan & Mulyanto, 2015)

Gambar 2. Siklus Algoritma Genetika

Terdapat beberapa kelebihan dari algoritma genetika dibandingkan algoritma lainnya, dua diantaranya yaitu kemampuan untuk menangani permasalahan kompleks dan paralel.

## 3. Metode Pengembangan Aplikasi

### a. Perencanaan

Pada tahap ini penulis mendefinisikan tujuan dan ruang lingkup pengembangan aplikasi.

### b. Analisis

Pada tahap ini penulis melakukan studi literatur untuk menemukan suatu kasus yang bisa ditangani oleh sistem, mengklasifikasikan masalah, peluang, dan solusi yang mungkin diterapkan untuk kasus tersebut, menganalisa kebutuhan pada sistem dan membuat batasannya serta mendefinisikan kebutuhan dari aplikasi.

### c. Perancangan

Pada tahap ini penulis menganalisa interaksi obyek dan fungsi pada sistem serta merancang *user interface* dengan menggunakan UML dan menggunakan IDE Android Studio.

### d. Implementasi

Pada tahap ini penulis melakukan instalasi dari aplikasi yang sudah dibuat untuk kemudian digunakan oleh *user* dengan menggunakan bahasa pemrograman Java dan Algoritma Genetika.

### e. Evaluasi

Pada tahap ini dilakukan evaluasi terhadap kinerja dari aplikasi yang sudah dibuat dengan menggunakan *black box testing*.

4. Layar : 12”

## 2. Desain

Dibawah ini akan dijelaskan tahap penggunaan algoritma genetika untuk menyelesaikan masalah prediksi buta warna.

### 1. Inisialisasi

Pada tahap ini dilakukan inisialisasi terhadap kemungkinan solusi yang merupakan sifat dari individu sebagai berikut:

Tabel 2. Inisialisasi Individu

No	Sifat Gen	Inisial
1	Wanita normal	3
2	Wanita carrier	4
3	Wanita penderita	1
4	Pria normal	2
5	Pria penderita	3

### 2. Pembangkitan populasi awal

Populasi awal yang dibangkitkan diambil dari jumlah anak berdasarkan jenis kelamin dan sifat gen orang tua.

Tabel 3. Populasi Awal

Kromosom ke-	Inisialisasi
1	1 2 4 2
2	0 4 4 2
3	2 1 3 5
4	0 3 4 2
5	2 1 4 2
6	0 1 4 2
7	0 2 4 2
8	2 2 3 5
9	0 4 1 2
10	3 3 1 2

### 3. Evaluasi Fitness

Fungsi fitness untuk setiap kemungkinan solusi adalah sebagai berikut:

$$f_a = -1,267 + 0,348.P + 0,193.L + 0,285.I - 0,095.A \quad (1)$$

$$f_b = -3,662 - 0,013.P + 0,0547.L + 0,459.I + 0,684.A \quad (2)$$

nilai yang diperoleh dari evaluasi fitness pada setiap kromosom/individu kemudian dibulatkan.

## HASIL DAN PEMBAHASAN

### 1. Analisa Kebutuhan Software

Hasil dari studi pustaka, wawancara dan observasi yang penulis lakukan dalam mengumpulkan data, dapat diidentifikasi beberapa masalah sebagai berikut:

1. Prediksi hereditas buta warna masih dilakukan secara manual dengan menggunakan hukum mendel.
2. Hasil prediksi buta warna belum memberikan hasil yang optimal.
3. Dibutuhkan aplikasi yang mampu diakses oleh banyak orang dengan sifat yang *user friendly* untuk memprediksi buta warna.

Setelah melakukan identifikasi dan mendefinisikan masalah penulis merumuskan kebutuhan untuk menyelesaikan permasalahan yang ditemukan pada identifikasi masalah, sebagai berikut:

#### a. Analisa Kebutuhan Perangkat Lunak

Untuk menyelesaikan permasalahan yang ditemukan pada penelitian ini, digunakan beberapa perangkat lunak, antara lain:

1. Bahasa pemrograman: Java 8 dan XML 1.0
2. IDE : Android Studio 3.1.2
3. Perancangan UML : StarUML versi 2.5.0
4. Sistem Operasi : Microsoft Windows 10

#### b. Analisa Kebutuhan Perangkat Keras

Untuk menyelesaikan permasalahan yang ditemukan maka dibutuhkan perangkat keras sebagai berikut:

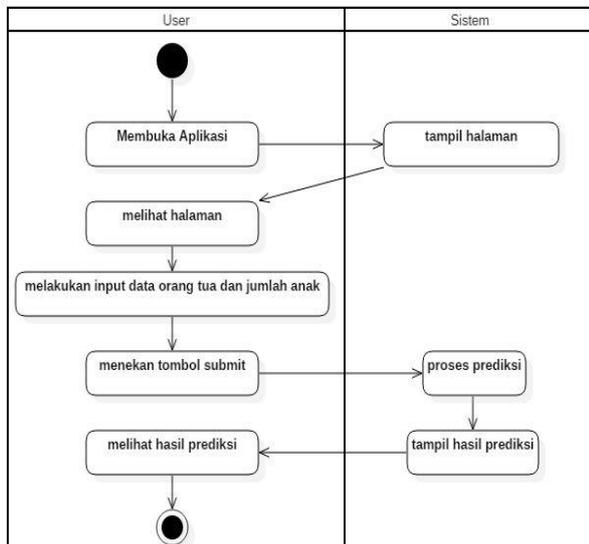
1. Processor : Intel® Core™2 Duo CPU L9400 @1.86GHz 1.87GHz
2. RAM : 5.00 GB
3. Hardisk : 150 GB

Tabel 4. Nilai Fitness

Kromosom ke	Jumlah Anak		Orang Tua		Nilai Fitness	
	Perempuan (P)	Laki-laki (L)	Ibu (I)	Ayah (A)	$f_a$	$f_b$
1	1	2	4	2	0	0
2	0	4	4	2	0	0
3	2	1	3	5	0	1
4	0	3	4	2	0	0
5	2	1	4	2	0	0
6	0	1	4	2	0	0
7	0	2	4	2	0	0
8	2	2	3	5	0	1
9	0	4	1	2	0	2
10	3	3	1	2	0	2

4. Seleksi individu

Pada tahap ini, individu dengan nilai fitness terbaik akan diambil untuk dijadikan sebagai induk, yaitu: Kromosom 5 = 2 1 4 2, dan Kromosom 10 = 3 3 1 2



Gambar 3. Activity Diagram Prediksi

5. Crossover

Pada tahap ini terjadi persilangan antara Kromosom 5 dengan kromosom 10, yang digambarkan sebagai berikut:

2	1	4	2
---	---	---	---

3	3	1	2
---	---	---	---

Hasil/offspring:

2	3	1	2
---	---	---	---

3	1	1	2
---	---	---	---

6. Crossover

Pada tahap ini terjadi persilangan antara Kromosom 5 dengan kromosom 10, yang digambarkan sebagai berikut:

2	1	4	2
---	---	---	---

3	3	1	2
---	---	---	---

Hasil/offspring:

2	3	1	2
---	---	---	---

3	1	1	2
---	---	---	---

7. Populasi baru terbentuk

Setelah terjadi crossover maka populasi baru terbentuk sebagai berikut:

Tabel 5. Populasi Baru

Kromosom ke	Jumlah Anak		Orang Tua	
	Perempuan (P)	Laki-laki (L)	Ibu (I)	Ayah (A)
1	1	2	4	2
2	0	4	4	2
3	2	1	3	5
4	0	3	4	2
5	2	3	4	2
6	0	1	4	2
7	0	2	4	2
8	2	2	3	5
9	0	4	1	2
10	3	1	1	2

Berikut adalah pseudocode dari Algoritma Genetika: *Procedure Genetic Algorithms*

```

begin
  t <- 0;
  inisialisasi P(t);
  evaluasi P(t);
  while (not syarat berhenti) do
    begin
      kombinasi-kembali P(t) untuk membentuk
      C(t);
      evaluasi C(t);
      pilih P(t + 1) dari P(t) dan C(t);
      t <- t + 1;
    end
  End;
  
```

Gambar 5. Activity Diagram Info

Adapun untuk desain user interface digunakan UML sebagai berikut:

1. Activity Diagram

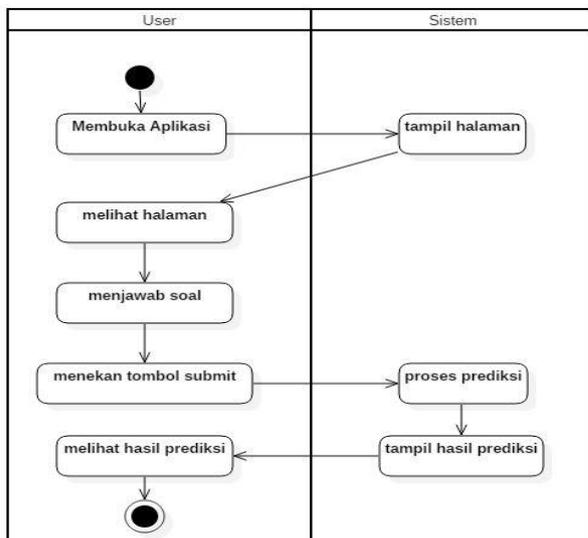
Berikut adalah activity diagram yang ada pada aplikasi prediksi buta warna yang penulis buat:

a. Activity Diagram Prediksi

Berikut adalah activity diagram untuk Prediksi:

b. Activity Diagram Test

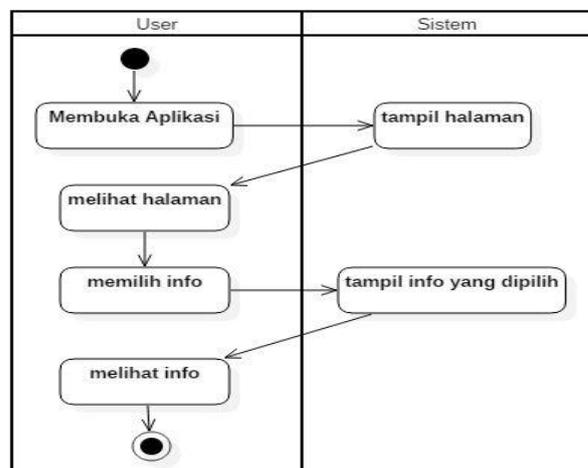
Berikut adalah activity diagram untuk Test:



Gambar 4. Activity Diagram Test

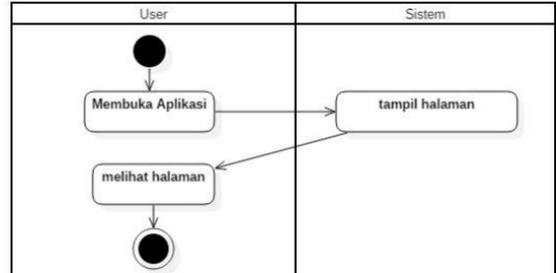
c. Activity Diagram Info

Berikut adalah activity diagram untuk Info:



d. Activity Diagram Tentang

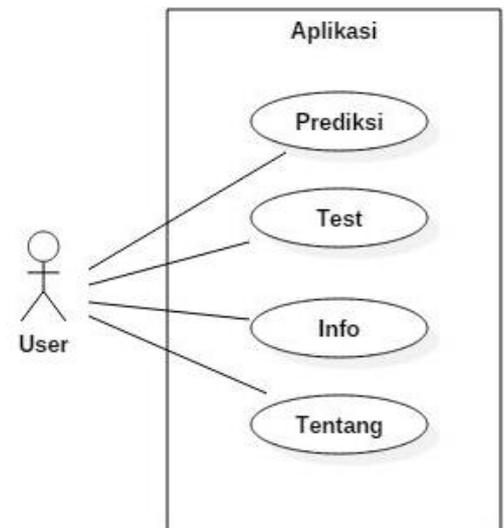
Berikut adalah activity diagram untuk Tentang:



Gambar 6. Activity Diagram Tentang

2. Use Case Diagram

Diagram di bawah ini memperlihatkan bagaimana peran user dalam menggunakan fungsi yang berbeda serta menu pada sistem.



Gambar 7. Usecase Diagram

3. Sequence Diagram

Sequence Diagram pada aplikasi prediksi buta warna ini mengacu kepada setiap use case dari Use Case Diagram diatas, berikut ini adalah beberapa Sequence Diagram-nya:



### 3. Testing

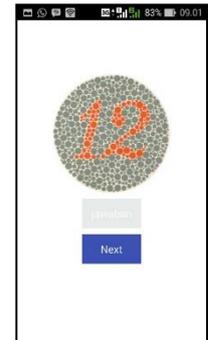
Testing yang penulis lakukan menggunakan Blackbox Testing memberikan hasil sebagai berikut:

Tabel 6. Blackbox Testing

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Hasil Uji
1.	Menampilkan menu utama	Halaman menu utama menampilkan 4 tombol pilihan menu, yaitu menu Prediksi, Test, Info dan Tentang		Sesuai
	Menampilkan Halaman Prediksi	Halaman Prediksi menampilkan radiobutton untuk pilihan sifat orang tua dan editText untuk input jumlah anak berdasarkan jenis kelamin dan tombol submit untuk melakukan prediksi		Sesuai
	Input data sifat orang tua dan jumlah anak berdasarkan jenis kelaminnya	Setelah user mengklik tombol submit, maka halaman prediksi menampilkan hasil dari prediksi		Sesuai
	Menampilkan Halaman Info	Halaman Info menampilkan 5 tombol pilihan informasi mengenai buta warna		Sesuai
	Menampilkan Halaman Tentang	Halaman Tentang menampilkan informasi singkat mengenai Aplikasi Prediksi Buta Warna		Sesuai

Menampilkan Halaman Test

Halaman Test menampilkan imageview untuk menampilkan plate soal tes buta warna, editText untuk input jawaban dan tombol next untuk melakukan prediksi



Sesuai

Input jawaban dari soal-test buta warna

Setelah user mengklik tombol submit, maka halaman Test menampilkan hasil dari test buta warna



Sesuai

Menampilkan Halaman Info

Halaman Info menampilkan 5 tombol pilihan informasi mengenai buta warna



Sesuai

Menampilkan Halaman Tentang

Halaman Tentang menampilkan informasi singkat mengenai Aplikasi Prediksi Buta Warna

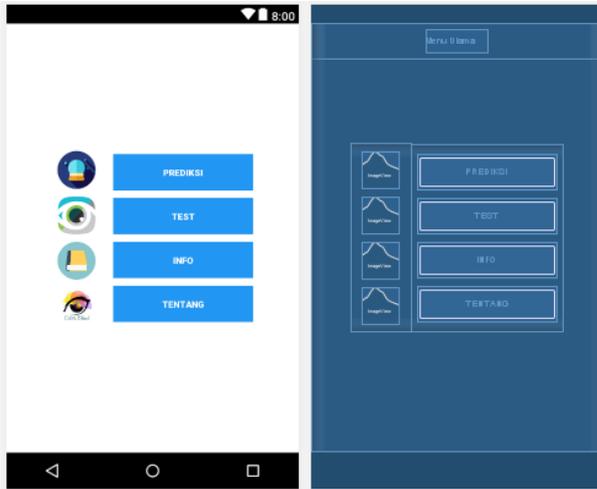


Sesuai

#### 4. Implementasi

##### 1. Tampilan Menu Utama

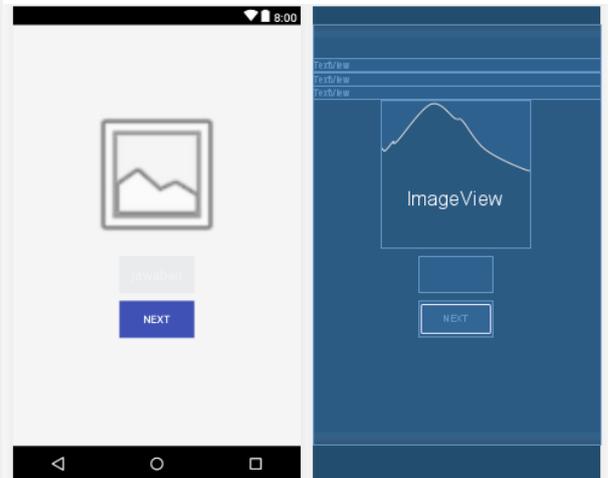
Berikut adalah tampilan menu Utama:



Gambar 14. Layout Menu Utama

##### 3. Tampilan Menu Test

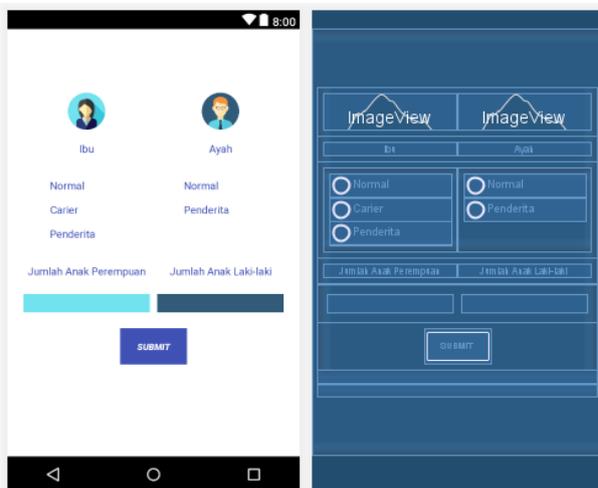
Berikut adalah tampilan menu Test:



Gambar 16. Layout Menu Test

##### 2. Tampilan Menu Prediksi

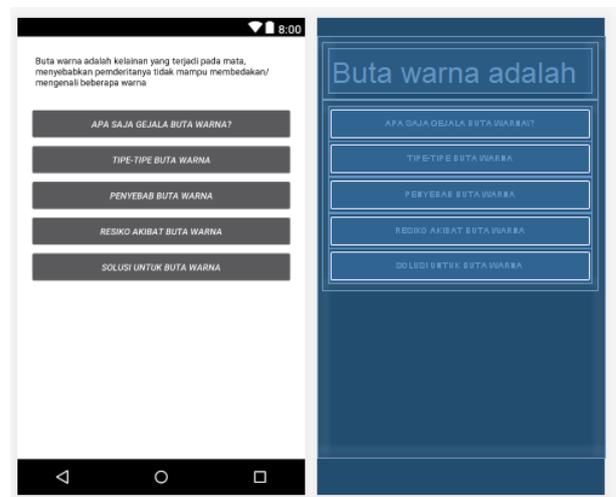
Berikut adalah tampilan menu Prediksi:



Gambar 15. Layout Menu Prediksi

##### 4. Tampilan Menu Info

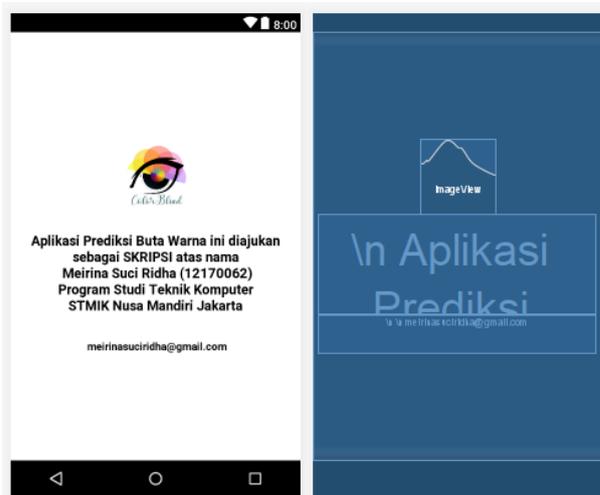
Berikut adalah tampilan menu Info:



Gambar 17. Layout Menu Info

## 5. Tampilan Menu Tentang

Berikut adalah tampilan menu Tentang:



Gambar 18. Layout Menu Tentang

## 5. Support

Spesifikasi perangkat keras yang diusulkan pada aplikasi Prediksi Buta Warna ini adalah sebagai berikut:

- Layar : 4,5"
- Processor : Intel Atom, Dual core 1.2 Ghz
- RAM : 512 MB
- Storage : 8 GB

Adapun spesifikasi perangkat lunak yang diusulkan pada aplikasi Prediksi Buta Warna ini adalah sebagai berikut:

- Sistem Operasi : *Android 6.0 (Marshmallow)*

## KESIMPULAN

Berdasarkan penelitian yang telah penulis lakukan dan berdasarkan penjelasan pada bab-bab sebelumnya, maka penulis menarik beberapa kesimpulan sebagai berikut:

- Aplikasi Prediksi Buta Warna berbasis mampu memberikan kemudahan bagi masyarakat untuk melakukan prediksi buta warna karena hanya dengan memberikan *input* berupa sifat orang tua dan jumlah anak, maka masyarakat dapat mengetahui hasilnya.
- Pemilihan smartphone berbasis Android menjadikan aplikasi prediksi buta warna menjadi sebuah aplikasi yang *userfriendly* dan dapat menjangkau banyak kalangan masyarakat.
- Implementasi Algoritma Genetika untuk prediksi buta warna mampu memberikan hasil prediksi yang lebih optimal.

## REFERENSI

Aditiawarman, Nasution, H., & Tursina. (2017). Sistem Pakar Pendeteksi Penyakit Mata

Berbasis Android. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, 1(2), 57–61. Retrieved from [jurnal.untan.ac.id/index.php/justin/article/view/18695](http://jurnal.untan.ac.id/index.php/justin/article/view/18695)

Andriani, A. (2013). Sistem prediksi penyakit diabetes berbasis decision tree. *Jurnal Bianglala Informatika*, 1(1), 1–10.

Darmawan, H. D., Yuniarti, D., & Nasution, N. (2017). Klasifikasi Lama Masa Studi Mahasiswa Menggunakan Perbandingan Metode Algoritma C . 45 dan Algoritma Classification and Regression Tree Comparison of C4 . 5 Algorithm and Classification and Regression Tree Algorithm In The Classification of Study Period o. *Jurnal Eksponensial*, 8(2), 151–160.

Ervan, D. S., & Mulyanto, E. (2015). DETEKSI RISIKO PENDERITA BUTA WARNA MENURUN. *Techno.COM*, 14(2), 145–150.

Fatmawati. (2016). PERBANDINGAN ALGORITMA KLASIFIKASI DATA MINING MODEL C4 . 5 DAN NAIVE BAYES UNTUK PREDIKSI PENYAKIT DIABETES. *Jurnal Techno Nusa Mandiri*, XIII(1), 50–59.

Harafani, H. (2018). OPTIMASI ALGORITMA GENETIKA PADA K-NN UNTUK MEMPREDIKSI KECENDERUNGAN “BLOG POSTING.” *Jurnal Pendidikan Teknologi Dan Kejuruan*, 15(1). <https://doi.org/10.23887/jptk-undiksha.v15i1.12873>

Harafani, H., & Wahono, R. S. (2015). Optimasi Parameter pada Support Vector Machine Berbasis Algoritma Genetika untuk Estimasi Kebakaran Hutan. *Journal of Intelligent Systems*, 1(2).

Kemala, V., Irawan, B., & Nasrun, M. (2015). Rancang Bangun Aplikasi Sistem Pakar Untuk Diagnosis Penyakit Kulit Dan Kelamin Berbasis Smartphone Android. *Program Studi Sistem Komputer, Fakultas Elektro Dan Komunikasi, Institut Teknologi Telkom Bandung*, 2(2), 3568–3574.

Mujahidin, A., & Pribadi, D. (2017). Penerapan Algoritma C4 . 5 Untuk Diagnosa Penyakit Pneumonia Pada Anak Balita Berbasis Mobile. *Jurnal Swabumi*, 5(2), 155–161.

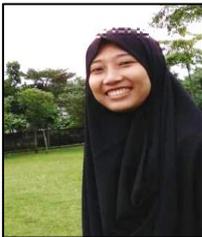
Praningki, T., & Budi, I. (2017). Sistem Prediksi Penyakit Kanker Serviks Menggunakan CART , Naive Bayes , dan k-NN. *Citec Journal*, 4(2), 83–93.

Puteri, N. A., Maharani, W., & Suliiyo, M. D. (2013). PREDIKSI PENYAKIT JANTUNG DENGAN ALGORITMA CLASSIFICATION AND REGRESSION TREE. *Tugas Akhir*.

### UCAPAN TERIMA KASIH

Akhir kata penulis mengucapkan terimakasih pada para peneliti sebelumnya, pada grup penelitian Intelligent Systems dan kepada semua tokoh pada referensi kami atas kontribusi tidak langsung pada penelitian ini, namun tanpa referensi dari para tokoh kami tidak mungkin memiliki pengetahuan dan ide mengenai penelitian ini.

### DAFTAR PROFIL PENULIS



**Meirina Suci Ridha.**  
Memperoleh gelar S.Kom dari Sekolah Tinggi Manajemen Ilmu Komputer Nusa Mandiri, Jakarta.



**Hani Harafani, M.Kom.**  
Dosen di STMIK Nusa Mandiri Jakarta. Minat penelitian saat ini pada bidang Data Mining