

Perancangan Aplikasi *Mobile* Kalkulator Pakan Ikan Berbasis Android untuk Optimalisasi Formulasi Pakan

Rian Prasetyo^{1*}, Ainur Komariah¹

¹Teknik Industri, Fakultas Teknik, Universitas Veteran Bangun Nusantara,
Jl. Letjend Sujono Humardani No.1, Gadingan, Jombor, Kabupaten Sukoharjo, Jawa Tengah 57521.

Penulis untuk Korespondensi/E-mail: rnprasetyo286@gmail.com

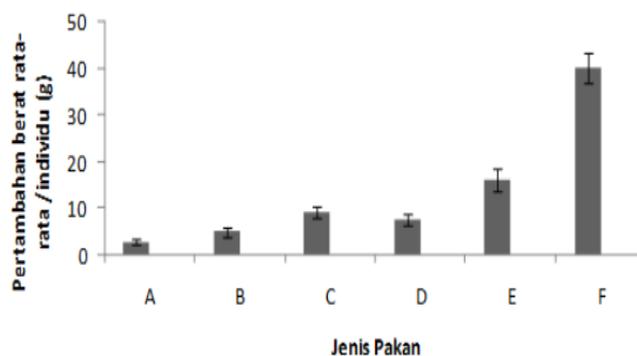
Abstract - Feed is a main factor influencing fish growth. However, feed formulation using traditional methods such as the Pearson Square is often considered complex and time-consuming for fish farmers. This study aims to design and development an Android-based application named “Hitung Pakan”, which can calculate fish feed formulations quickly and accurately. The research employed the waterfall method, including requirement analysis, system design, implementation, and testing. The application was developed using Java programming language and tested by 10 respondents from a fish farming group. The results showed no differences between the application’s calculations and manual calculations using Pearson Square method equations. This indicates that the developed application provides accurately results, simplifies the formulation process, and reduces human error. Therefore, the application is feasible to be used as a formulation tool in fish farming group.

Abstrak - Pakan merupakan faktor utama yang memengaruhi pertumbuhan ikan. Namun, proses formulasi pakan dengan metode manual, seperti *Pearson Square*, sering dianggap rumit dan memakan waktu oleh pembudidaya ikan. Penelitian ini bertujuan untuk membuat aplikasi “Hitung Pakan” berbasis Android yang mampu menghitung formulasi pakan ikan secara cepat dan akurat. Metode yang digunakan adalah *waterfall* dengan tahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Aplikasi dikembangkan menggunakan bahasa pemrograman Java dan diuji pada 10 responden dari kelompok budidaya ikan. Hasil pengujian menunjukkan tidak terdapat perbedaan antara hasil perhitungan aplikasi dengan perhitungan manual menggunakan persamaan dari metode *Pearson Square*. Hal ini membuktikan bahwa aplikasi yang dikembangkan mampu memberikan hasil yang presisi, memudahkan proses formulasi serta mengurangi risiko kesalahan perhitungan. Dengan demikian, aplikasi ini layak digunakan sebagai alat bantu praktis dalam penyusunan formulasi pakan ikan mandiri.

Keywords - *Android Application, Calculation Efficiency, Fish Feed Formulation, Pearson Square.*

PENDAHULUAN

Pakan merupakan faktor penting dalam pertumbuhan ikan. Komposisi dan jumlah pemberian pakan yang tepat dapat meningkatkan pertumbuhan secara optimal. Pakan yang berbeda menghasilkan pertambahan berat ikan lele yang bervariasi (gambar 1) [1]. frekuensi pemberian pakan memengaruhi berat ikan nila hingga 37% lebih tinggi bila diberikan dua kali sehari dibanding satu kali sehari [2], [3]. Hal ini menunjukkan pakan ikan merupakan hal yang sangat penting terhadap pertumbuhan ikan.



Gambar 1. Pertambahan Berat Ikan Lele pada 45 hari untuk 6 Jenis Pakan yang Berbeda [1]

Pakan ikan produksi dari pabrik semakin menunjukkan peningkatan harga [4], [5]. Kondisi tersebut berakibat pakan kurang bisa terjangkau oleh pembudidaya ikan. Hal ini salah satunya diakibatkan oleh fluktuasi nilai tukar rupiah terhadap dollar AS [6]. Berdasarkan kondisi tersebut, maka diperlukan solusi pakan alternatif atau pembuatan pakan dengan bahan sendiri.

Pakan ikan dapat berupa pakan alami maupun buatan. Pakan alami banyak terdapat pada alam, sedangkan pakan buatan didapatkan dengan formula tertentu sesuai dengan kebutuhan nutrisi ikan. Sejauh ini mayoritas budidaya ikan banyak menggunakan pakan buatan dari pabrik [7].

Formulasi pakan ikan adalah tahapan yang digunakan untuk mendapatkan campuran bahan pakan sesuai dengan nutrisi yang dibutuhkan ikan [8]. Salah satu metode formulasi pakan ikan yang cukup sederhana dan populer adalah *Pearson Square*. Metode ini didasarkan pada klasifikasi protein basal (bahan baku dengan kandungan protein kurang dari 20%) dan protein suplemen (bahan baku dengan kandungan protein lebih dari 20%) [9].

Berdasarkan observasi yang dilakukan di salah satu kelompok budidaya ikan, metode *Pearson Square* pernah diajarkan untuk melakukan formulasi pakan ikan, namun belum sepenuhnya dapat menghitung dengan cepat dan tepat. Kondisi tersebut karena kelompok menganggap formulasi pakan masih memerlukan perhitungan yang cukup kompleks.

Salah satu cara agar pembudidaya ikan tidak perlu melakukan perhitungan kompleks adalah dengan mengintegrasikan proses perhitungan pada *software* atau aplikasi. Integrasi formulasi pakan pada *software* atau aplikasi pernah dilakukan oleh beberapa peneliti. Aplikasi *mobile* untuk formulasi pakan sapi dan kambing telah dikembangkan [10]. Pengembangan aplikasi *mobile* untuk menghitung kebutuhan nutrisi ternak juga pernah dilakukan, pada ternak bebek [11] dan silase pakan ternak domba [12] serta aplikasi formulasi pakan ternak sapi potong [13]. Beberapa aplikasi pernah dikembangkan oleh peneliti sebelumnya, namun belum ada yang terkait dengan budidaya ikan.

Berdasarkan kondisi tersebut tujuan penelitian ini adalah untuk merancang dan membangun aplikasi kalkulator formulasi pakan ikan. Hal ini dilakukan agar pembudidaya ikan tidak perlu melakukan perhitungan kompleks untuk mendapatkan formulasi pakan ikan yang diinginkan. Aplikasi akan

dibuat dengan berbasis android. Hal ini karena sistem operasi tersebut merupakan yang terbanyak digunakan di Indonesia (pengguna mencapai 74,2%) per Januari 2016 [14].

METODE

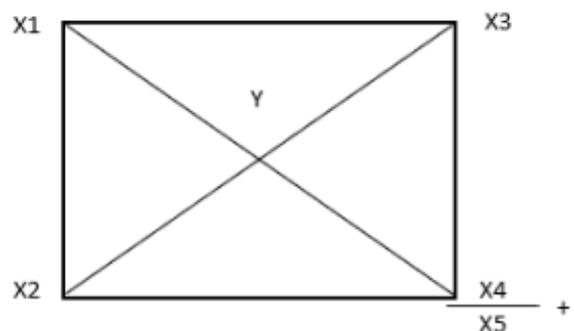
Desain penelitian

Penelitian ini mengambil studi kasus pada kelompok tani ikan GP, yang mengelola budidaya ikan dengan keramba jaring apung di Bulu Sukoharjo, dengan anggota 10 orang.

Metode *Waterfall*

Analisis Kebutuhan

Tahap ini bertujuan untuk mengidentifikasi kebutuhan pengguna dan spesifikasi sistem. Wawancara dilakukan dengan 10 responden yang merupakan anggota kelompok budidaya ikan, sekaligus sebagai *user* aplikasi yang akan dirancang. Hal ini untuk mengumpulkan data apa saja yang menjadi kebutuhan untuk dimunculkan di aplikasi yang akan dibuat. Aplikasi akan menggunakan Metode *Pearson Square* [9], dengan tahapan sebagai berikut, (1) Menentukan bahan baku yang akan digunakan, (2) Mengklasifikasikan bahan baku menjadi protein basal dan protein suplemen, (3) Menghitung rata-rata kandungan protein basal dan protein suplemen, (4) Menyusun kotak perhitungan berbentuk persegi panjang (gambar 2).



Gambar 2. Kotak Perhitungan [9]

Keterangan:

Y = % protein pakan yang diinginkan

X1 = % protein basal

X2 = % protein supplement

X3 = X2 - Y

X4 = Y - X1

X5 = X3 + X4

(5) Melakukan perhitungan komposisi pada setiap bahan baku dengan persamaan (1), (2).

$$\text{Protein basal} = \frac{x_3}{x_5} \times 100\% \quad (1)$$

$$\text{Protein supplement} = \frac{x_4}{x_5} \times 100\% \quad (2)$$

Perancangan Sistem (Design)

Tahap ini mencakup pembuatan desain teknis aplikasi. Penyusunan algoritma dan *flowchart* dari aplikasi yang akan disusun pada tahap ini. Penulisan *coding* untuk aplikasi kalkulator formulasi pakan dan *user interface* akan disusun dengan *software* android studio. File dengan format .apk akan dihasilkan pada tahap ini.

Pengujian

Pada tahap ini aplikasi diuji untuk memastikan fungsionalitas dan kinerja. Pengujian dilakukan oleh seluruh anggota kelompok tani ikan (10 orang). Masing-masing responden akan melakukan proses penghitungan formulasi pakan ikan sesuai dengan prosedur yang telah ditentukan. Urutan prosedur pengujian adalah sebagai berikut, (1) *User* menginstall Aplikasi Hitung Pakan, (2) *User* membuka aplikasi, (3) *User* memasukkan bahan pakan, kadar protein yang diinginkan, protein basal dan suplemen, (4) Aplikasi menghitung formulasi pakan, (5) *User* mendapatkan hasil formulasi pakan.

Analisis data dan pembahasan

Analisis data dan pembahasan yang dilakukan terkait dengan pengambilan data yang dilakukan. Pembahasan terkait gambaran berhasil atau tidaknya *user* melakukan perhitungan formulasi pakan ikan dengan aplikasi yang telah dibuat.

HASIL DAN PEMBAHASAN

Analisis kebutuhan

Hasil wawancara menunjukkan bahwa kelompok budidaya ikan sering mengalami kesulitan dalam menghitung komposisi pakan secara manual, terutama dalam menentukan perbandingan bahan pakan yang memiliki kadar protein berbeda. Selama ini, proses perhitungan dilakukan dengan persamaan (1) dan (2) dari Metode *Pearson Square* secara tertulis. Sejauh ini belum ada aplikasi khusus untuk menyelesaikan perhitungan formulasi pakan, sehingga perhitungan dilakukan dengan kalkulator biasa. Cara tersebut memakan waktu lebih lama dan rentan terjadi kesalahan hitung, karena perhitungan yang kompleks. Berdasarkan wawancara, pengguna menginginkan aplikasi dengan ketentuan seperti berikut.

Sederhana dan mudah digunakan

Tampilan antarmuka (*user interface*) harus intuitif, sehingga pengguna tanpa latar belakang teknologi tinggi tetap dapat mengoperasikannya. Navigasi antar halaman harus jelas dan tidak memerlukan banyak langkah.

Dapat Menggunakan Metode *Pearson Square*

Aplikasi harus dapat melakukan perhitungan formulasi pakan ikan berdasarkan Metode *Pearson Square* seperti yang telah dijelaskan [9], sesuai yang dipelajari kelompok budidaya ikan. Proses perhitungan dilakukan secara otomatis setelah pengguna memasukkan data.

Input Data Fleksibel

Pengguna dapat memasukkan jenis bahan pakan, kadar protein yang diinginkan serta nilai protein bahan pakan basal dan suplemen. Data bahan pakan dapat diubah sesuai kondisi dan ketersediaan bahan di lapangan.

Output yang Jelas dan Informatif

Hasil perhitungan ditampilkan secara langsung dalam bentuk berat komposisi bahan pakan basal dan suplemen. Menyediakan ringkasan hasil yang mudah dibaca dan sederhana.

Efisiensi Waktu dan Akurasi

Aplikasi diharapkan dapat mengurangi waktu perhitungan. Kesalahan perhitungan yang sering terjadi pada metode manual dapat diminimalkan.

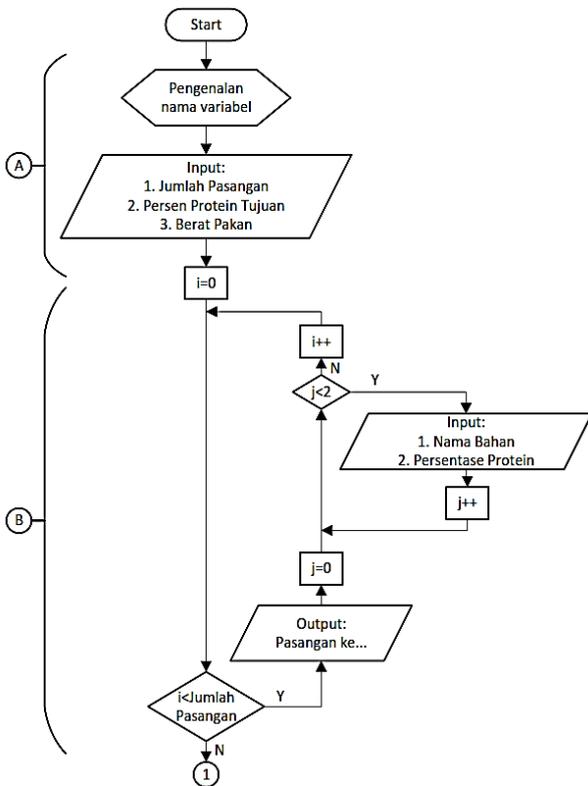
Berdasarkan kebutuhan tersebut, spesifikasi sistem yang akan dibangun meliputi, (1) Fungsi utama aplikasi formulasi pakan ikan berbasis rumus *Pearson Square*, (2) *Platform* berbasis Android dengan file instalasi berformat apk. (3) Komponen input berupa Jenis bahan pakan, kadar protein yang diinginkan, kadar protein bahan basal, kadar protein bahan suplemen dan berat pakan yang akan dibuat. (4) Komponen output berupa berat komposisi bahan pakan basal dan suplemen serta total formulasi.

Dengan adanya identifikasi kebutuhan yang jelas pada tahap ini, proses perancangan dan pengembangan aplikasi akan lebih terarah, sehingga hasil akhir diharapkan benar-benar sesuai dengan kebutuhan petani ikan dan dapat digunakan secara praktis di lapangan.

Penyusunan algoritma (Flowchart)

Algoritma dalam pembuatan aplikasi ini disusun berupa *flowchart* yang akan memberikan gambaran tahapan yang runtut. *Flowchart* akan dibahas pada 3

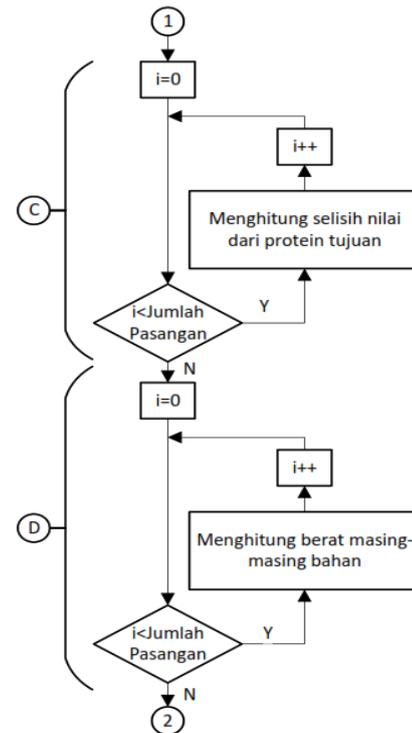
tahapan dari program yang akan dirancang. Tahap pertama adalah proses input. *Flowchart* proses input dapat dilihat pada gambar 3.



Gambar 3. *Flowchart* Proses Input

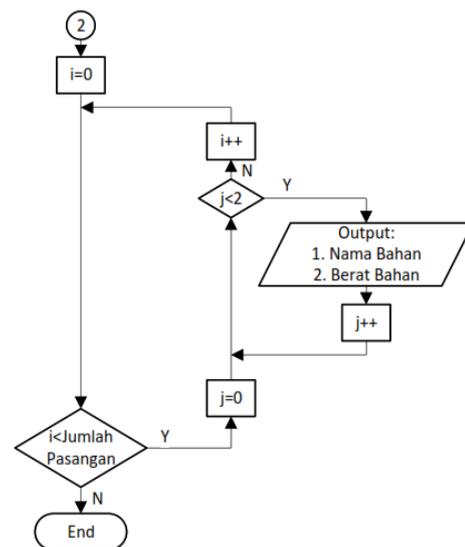
Berdasarkan *flowchart* pada gambar 3, bagian A, pengguna diminta memasukkan jumlah pasangan bahan pakan, target persentase protein dan total berat pakan yang ingin dibuat. Bagian B menunjukkan pengguna akan memasukkan setiap pasangan terdiri dari dua bahan. Masing-masing bahan diberi input berupa nama bahan dan persentase protein dengan cara perulangan (*looping*). Validasi dilakukan dengan syarat, bahan pertama harus memiliki protein lebih tinggi dari target, sedangkan bahan kedua harus memiliki protein lebih rendah dari target. Jika tidak sesuai, akan diminta ulang menginput. Hal ini sesuai dengan prinsip dasar *Pearson Square*, perhitungan membutuhkan satu bahan dengan protein lebih tinggi dan satu bahan dengan protein lebih rendah.

Tahap kedua adalah proses perhitungan. Tahap ini merupakan lanjutan dari tahap 1. Proses perhitungan dilakukan dalam dua tahap yaitu menghitung selisih kadar protein dan menentukan bobot tiap bahan. *Flowchart* proses perhitungan dapat dilihat pada Gambar 4.



Gambar 4. *Flowchart* Proses Perhitungan

Berdasarkan gambar 4 dapat dilihat pada bagian C merupakan tahap menghitung selisih kadar protein antara target dan bahan dengan cara perulangan (*looping*). Hal ini dikenal sebagai nilai bagian pada *Pearson Square Method*. Perhitungan selisih terdapat 2 jenis yaitu (1) Selisih antara target dan bahan rendah, untuk menentukan bobot bahan tinggi. (2) Selisih antara bahan tinggi dan target, untuk menentukan bobot bahan rendah. Bagian D menunjukkan tahapan perhitungan proporsi berat tiap bahan dengan cara perulangan (*looping*).



Gambar 5. *Flowchart* Proses Menampilkan Hasil

Perhitungan pada bagian ini adalah penjumlahan dari kedua selisih, pembagian total berat pakan secara merata ke setiap pasangan dan berat aktual masing-masing bahan dalam kilogram. Dengan perhitungan ini, setiap pasangan bahan akan menghasilkan campuran pakan yang kandungan proteinnya mendekati target. Tahap ketiga adalah menampilkan rincian hasil perhitungan. Tahap ini merupakan lanjutan dari tahap 2.

Berdasarkan gambar 5, untuk setiap pasangan ditampilkan nama bahan beserta bobot yang harus dicampurkan. Dengan demikian pengguna dapat langsung mengetahui berapa kilogram dari tiap bahan pakan yang harus digunakan agar kandungan protein pakan sesuai dengan persentase target. Program akan menampilkan hasil dengan format desimal dua angka di belakang koma, sehingga lebih mudah dibaca.

Penyusunan Source Code

Halaman Input Data Bahan Pakan

Halaman awal aplikasi digunakan untuk memasukkan data bahan pakan yang akan dihitung. Input pertama yang disediakan meliputi jumlah pasangan bahan, persentase protein target serta berat total pakan yang akan dibuat. Selain itu juga terdapat input kedua yang meliputi nama bahan pakan dan persentase protein (sesuai dengan jumlah pasangan pada input pertama). Data ini diisi melalui komponen “*EditText*” yang memungkinkan pengguna mengetikkan angka sesuai kebutuhan. *Source code* pilihan input disusun dengan Bahasa pemrograman Java dan dapat dilihat pada gambar 6. Pada Android Studio, halaman ini dibangun menggunakan XML *layout* seperti yang terlihat pada gambar 7.

```

EditText editJumlahPasangan, editProteinTarget, editTotalBerat;
Button btnProses;

editJumlahPasangan = findViewById(R.id.editJumlahPasangan);
editProteinTarget = findViewById(R.id.editProteinTarget);
editTotalBerat = findViewById(R.id.editTotalBerat);
btnProses = findViewById(R.id.btnProses);
    
```

Gambar 6. Source Code Pilihan Input

Source code pada gambar 6 menunjukkan inisialisasi komponen input pada aktivitas utama. Masing-masing pilihan pada “*EditText*” terhubung dengan *layout XML* pada gambar 7, sehingga pengguna dapat mengisi data langsung melalui antarmuka aplikasi. Hal ini menjadi fondasi utama sebelum proses perhitungan dilakukan. Hasil *layout* antarmuka pada aplikasi android yang dibuat dapat dilihat pada gambar 8 dan gambar 9.

```

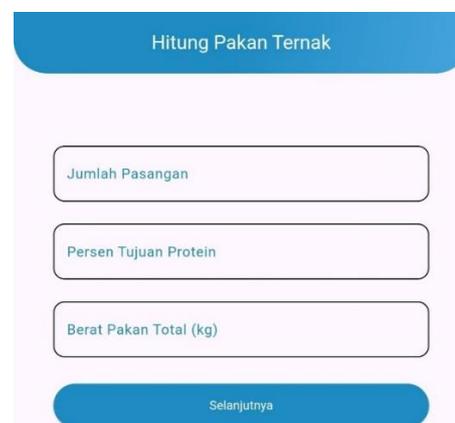
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <!-- Input kadar protein bahan 1 -->
    <EditText
        android:id="@+id/inputProtein1"
        android:hint="Kadar Protein Bahan 1 (%)"
        android:inputType="numberDecimal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

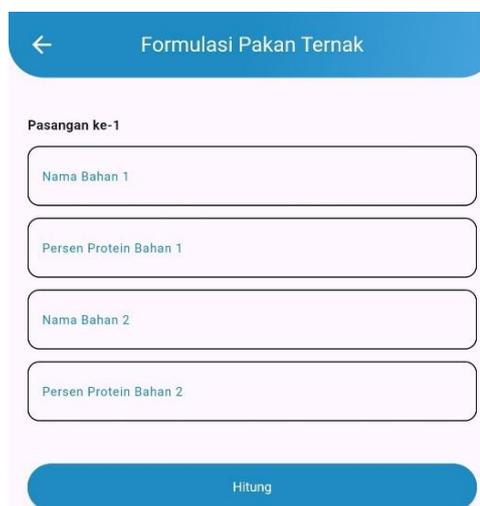
    <!-- Input kadar protein bahan 2 -->
    <EditText
        android:id="@+id/inputProtein2"
        android:hint="Kadar Protein Bahan 2 (%)"
        android:inputType="numberDecimal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <!-- Input target protein -->
    <EditText
        android:id="@+id/inputTargetProtein"
        android:hint="Target Protein (%)"
        android:inputType="numberDecimal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    
```

Gambar 7. Source Code Layout Input



Gambar 8. Antarmuka Input Pertama pada Aplikasi Hitung Pakan



Gambar 9. Antarmuka Input Kedua pada Aplikasi Hitung Pakan

Proses Perhitungan

Setelah data dimasukkan pada gambar 8 dan gambar 9, pengguna akan menekan tombol “hitung” untuk menjalankan Metode *Pearson Square*. Tombol tersebut disusun dengan *source code layout XML* yang dapat dilihat pada gambar 10. Setelah itu aplikasi yang dibuat akan menghitung formulasi pakan. Formulasi pakan dari Metode *Pearson Square* disusun dalam Bahasa pemrograman Java yang dapat dilihat pada gambar 11.

```
<!-- Tombol proses -->
<Button
    android:id="@+id/btnHitung"
    android:text="Hitung Formulasi"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"/>
</LinearLayout>
```

Gambar 10. *Source Code Layout XML*

```
Button btnHitung = findViewById(R.id.btnHitung);
btnHitung.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        double protein1 = Double.parseDouble(inputProtein1.getText().toString());
        double protein2 = Double.parseDouble(inputProtein2.getText().toString());
        double target = Double.parseDouble(inputTargetProtein.getText().toString());

        // Rumus Pearson Square
        double selisih1 = Math.abs(protein2 - target);
        double selisih2 = Math.abs(protein1 - target);
        double total = selisih1 + selisih2;

        double propors1 = (selisih1 / total) * 100;
        double propors2 = (selisih2 / total) * 100;

        // Kirim hasil ke halaman berikutnya
        Intent intent = new Intent(MainActivity.this, HasilActivity.class);
        intent.putExtra("hasil1", propors1);
        intent.putExtra("hasil2", propors2);
        startActivity(intent);
    }
});
```

Gambar 11. *Source Code* Perhitungan Formulasi Pakan

Saat tombol “hitung” ditekan, aplikasi mengambil nilai dari input pengguna, lalu menjalankan algoritma *Pearson Square* yang telah tersusun pada gambar 11. Proses perhitungan melibatkan pengurangan antara kadar protein bahan dengan target, lalu dihitung proporsinya sehingga diperoleh persentase campuran bahan pakan. Hasil ini kemudian dikirimkan ke halaman hasil menggunakan perintah “*Intent*” pada *source code*.

Halaman Hasil Perhitungan

Halaman terakhir berfungsi untuk menampilkan output berupa bobot masing-masing bahan pakan. *Source code* tampilan dibuat dengan XML, dapat dilihat pada gambar 12. *Source code* untuk mengisi tampilan tersebut dibuat dengan Bahasa pemrograman Java, dapat dilihat pada gambar 13.

```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <TextView
        android:id="@+id/txtHasil1"
        android:text="Hasil Bahan 1"
        android:textSize="18sp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

    <TextView
        android:id="@+id/txtHasil2"
        android:text="Hasil Bahan 2"
        android:textSize="18sp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

Gambar 12. *Source Code Layout* Tampilan Hasil Perhitungan

```
TextView txtHasil1 = findViewById(R.id.txtHasil1);
TextView txtHasil2 = findViewById(R.id.txtHasil2);

double hasil1 = getIntent().getDoubleExtra("hasil1", 0);
double hasil2 = getIntent().getDoubleExtra("hasil2", 0);

txtHasil1.setText("Bahan 1: " + String.format("%.2f", hasil1) + " %");
txtHasil2.setText("Bahan 2: " + String.format("%.2f", hasil2) + " %");
```

Gambar 13. *Source Code* Perintah Isi Tampilan

Gambar 14. Output Tampilan pada Aplikasi Android Hitung Pakan

Berdasarkan gambar 12 dan gambar 13, informasi ditampilkan dalam bentuk “*TextView*”, sehingga pengguna langsung mengetahui berapa kilogram dari setiap bahan yang harus dicampurkan. Dengan format dua angka di belakang koma, hasil terlihat

lebih presisi dan mudah dipahami. Tampilan ini menjadi keluaran akhir yang digunakan oleh pengguna untuk menentukan takaran pencampuran pakan. Output tampilan pada aplikasi android Hitung Pakan dapat dilihat pada gambar 14.

Pengujian

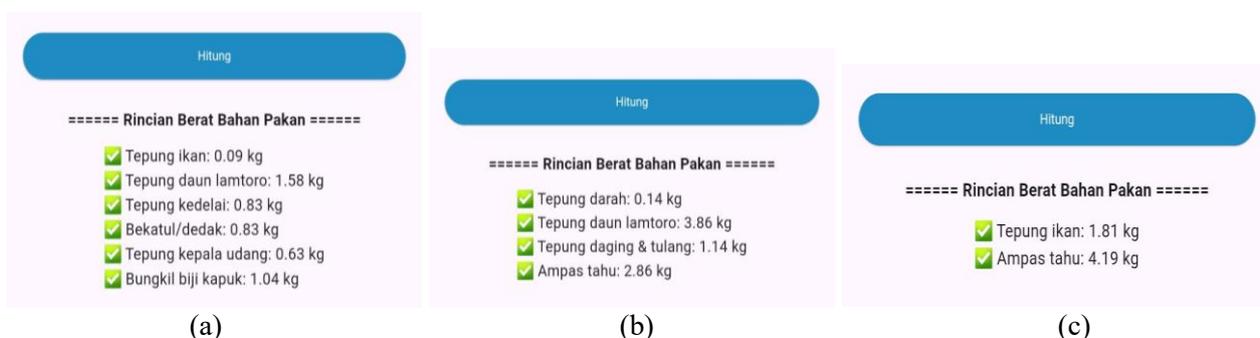
Setelah aplikasi selesai dibuat kemudian dilakukan instalasi pada *smartphone* android masing-masing responden. Instalasi menggunakan file apk. dari android studio yang sebelumnya telah dibuat. Setelah aplikasi terinstal, responden diberikan *task* untuk melakukan formulasi hitung pakan. Tabel 1 menunjukkan daftar task yang diberikan kepada responden.

Tabel 1. *Task* yang Diberikan pada Responden

Resp.	Berat Pakan	Persentase Protein Tujuan	Bahan Protein Tinggi	Bahan Protein Rendah
1	5	30	Tepung ikan (65%) Tepung kedelai (47%) Tepung kepala udang (40%)	Tepung daun lamtoro (28%) Bekatul/dedak (13%) Bungkil biji kapuk (24%)
2	11	35	Tepung ikan (65%) Tepung kedelai (47%)	Tepung kacang hijau (25%) Bekatul/dedak (13%)
3	7	30	Tepung kedelai (47%) Tepung ikan (65%)	Tepung daun lamtoro (28%) Tepung kacang hijau (25%)
4	10	35	Tepung darah (85%)	Tepung jagung (9%) Ampas tahu (22%)

Resp.	Berat Pakan	Persentase Protein Tujuan	Bahan Protein Tinggi	Bahan Protein Rendah
5	8	30	Tepung daging & tulang (50%) Tepung darah (85%) Tepung daging & tulang (50%)	Tepung daun lamtoro (28%) Ampas tahu (22%)
6	6	35	Tepung ikan (65%) Tepung ikan (65%)	Ampas tahu (22%) Bekatul/dedak (13%)
7	12	30	Tepung kedelai (47%) Maggot kering (40%) Tepung kepala udang (40%)	Tepung daun lamtoro (28%) Tepung kacang hijau (25%) Bungkil kelapa (23%) Azolla (25%)
8	14	35	Maggot kering (40%) Tepung kepala udang (40%)	Tepung kacang hijau (25%)
9	9	30	Tepung ikan (65%) Tepung ikan (65%)	Bungkil kelapa (23%) Ampas tahu (22%)
10	13	35	Tepung kepala udang (40%)	Keong mas (14%)

Berdasarkan tabel 1, responden diminta langsung menggunakan Aplikasi Hitung Pakan, dengan input sesuai daftar pada tabel. Beberapa contoh hasil formulasi pakan oleh responden dapat dilihat pada gambar 15.



Gambar 15. Hasil formulasi pakan dengan Aplikasi Hitung Pakan oleh (a) responden 1, (b) responden 5, (c) responden 6

Gambar 15 menunjukkan bahwa secara umum aplikasi yang telah dibuat mampu digunakan untuk formulasi pakan. Hasil formulasi pakan oleh 10 responden kemudian direkap dan dibandingkan dengan hasil perhitungan secara manual menggunakan Metode *Pearson Square*. Perbandingan tersebut dapat dilihat pada tabel 2. Contoh hasil perhitungan secara manual dapat dilihat sebagai berikut.

Responden 1

Pasangan bahan 1:

1. Tepung ikan

$$\text{Persentase} = \frac{(30-28)}{(65-30)+(30-28)} \times 100 = 5,41 \%$$

$$\text{Berat} = \frac{5,41\% \times 5}{3} = 0,09 \text{ kg}$$

2. Tepung daun lamtoro

$$\text{Persentase} = \frac{(65-30)}{(65-30)+(30-28)} \times 100 = 94,59 \%$$

$$\text{Berat} = \frac{94,59\% \times 5}{3} = 1,58 \text{ kg}$$

Pasangan bahan 2:

1. Tepung kedelai

$$\text{Persentase} = \frac{(30-13)}{(47-30)+(30-13)} \times 100 = 50 \%$$

$$\text{Berat} = \frac{50\% \times 5}{3} = 0,83 \text{ kg}$$

2. Bekatul/dedak

$$\text{Persentase} = \frac{(47-30)}{(47-30)+(30-13)} \times 100 = 50 \%$$

$$\text{Berat} = \frac{50\% \times 5}{3} = 0,83 \text{ kg}$$

Pasangan bahan 3:

1. Tepung kepala udang

$$\text{Persentase} = \frac{(30-24)}{(40-30)+(30-24)} \times 100 = 37,5 \%$$

$$\text{Berat} = \frac{37,5\% \times 5}{3} = 0,63 \text{ kg}$$

2. Bungkil biji kapuk

$$\text{Persentase} = \frac{(40-30)}{(40-30)+(30-24)} \times 100 = 62,5 \%$$

$$\text{Berat} = \frac{62,5\% \times 5}{3} = 1,04 \text{ kg}$$

Berdasarkan tabel 2, hasil formulasi pakan ikan yang dihitung menggunakan Aplikasi Hitung Pakan menunjukkan kesesuaian penuh 100% dengan hasil perhitungan manual menggunakan Metode *Pearson Square*. Setiap pasangan bahan, baik bahan dengan kandungan protein tinggi maupun protein rendah, menghasilkan bobot formulasi yang identik antara perhitungan aplikasi dan manual. Hal ini menunjukkan bahwa algoritma *Pearson Square* yang diintegrasikan ke dalam aplikasi dapat berjalan dengan baik tanpa menimbulkan perbedaan nilai.

Tabel 2. Perbandingan Hasil Formulasi Menggunakan Aplikasi Hitung Pakan dengan Perhitungan Manual

Resp.	Berat Pakan	Persentase Protein Tujuan	Bahan Protein Tinggi			Bahan Protein Rendah		
			Nama Bahan	Hasil Aplikasi (kg)	Hasil Hitung Manual (kg)	Nama Bahan	Hasil Aplikasi (kg)	Hasil Hitung Manual (kg)
1	5	30	1. Tepung ikan (65%)	0,09	0,09	1. Tepung daun lamtoro (28%)	1,58	1,58
			2. Tepung kedelai (47%)	0,83	0,83	2. Bekatul/dedak (13%)	0,83	0,83
			3. Tepung kepala udang (40%)	0,63	0,63	3. Bungkil biji kapuk (24%)	1,04	1,04
2	11	35	1. Tepung ikan (65%)	1,38	1,38	1. Tepung kacang hijau (25%)	4,13	4,13
			2. Tepung kedelai (47%)	3,56	3,56	2. Bekatul/dedak (13%)	1,94	1,94
3	7	30	1. Tepung kedelai (47%)	0,74	0,74	1. Tepung daun lamtoro (28%)	6,26	6,26
			2. Tepung ikan (65%)	0,83	0,83	1. Tepung kacang hijau (25%)	2,50	2,50
4	10	35	2. Tepung darah (85%)	1,14	1,14	2. Tepung jagung (9%)	2,19	2,19
			3. Tepung daging & tulang (50%)	1,55	1,55	3. Ampas tahu (22%)	1,79	1,79
			1. Tepung darah (85%)	0,14	0,14	1. Tepung daun lamtoro (28%)	3,86	3,86
5	8	30	2. Tepung daging & tulang (50%)	1,14	1,14	2. Ampas tahu (22%)	2,86	2,86
			1. Tepung ikan (65%)	1,81	1,81	1. Ampas tahu (22%)	4,19	4,19
7	12	30	1. Tepung ikan (65%)	1,31	1,31	1. Bekatul/dedak (13%)	2,69	2,69
			2. Tepung kedelai (47%)	0,42	0,42	2. Tepung daun lamtoro (28%)	3,58	3,58
			3. Maggot kering (40%)	1,33	1,33	3. Tepung kacang hijau (25%)	2,67	2,67
8	14	35	1. Tepung kepala udang (40%)	4,94	4,94	1. Bungkil kelapa (23%)	2,06	2,06
			2. Maggot kering (40%)	4,67	4,67	2. Azolla (25%)	2,33	2,33
9	9	30	1. Tepung kepala udang (40%)	3,00	3,00	1. Tepung kacang hijau (25%)	6,00	6,00
			1. Tepung ikan (65%)	1,24	1,24	1. Bungkil kelapa (23%)	3,10	3,10
10	13	35	2. Tepung ikan (65%)	1,31	1,31	2. Ampas tahu (22%)	3,02	3,02
			3. Tepung kepala udang (40%)	3,50	3,50	3. Keong mas (14%)	0,83	0,83

Kesesuaian hasil ini menunjukkan bahwa aplikasi mampu menggantikan perhitungan manual yang sejauh ini dianggap kompleks dan rawan kesalahan hitung oleh pembudidaya ikan. Dengan penggunaan aplikasi, proses formulasi pakan menjadi lebih cepat, praktis, serta meminimalkan potensi kesalahan perhitungan. Selain itu, hasil yang ditampilkan dalam format desimal dua angka di belakang koma memberikan tingkat presisi yang baik dan mudah dipahami oleh pengguna.

Secara garis besar, tidak adanya perbedaan antara hasil aplikasi dengan perhitungan manual mengindikasikan bahwa aplikasi yang dirancang layak digunakan sebagai alat bantu formulasi pakan ikan. Dengan demikian, aplikasi ini dapat menjadi solusi praktis bagi pembudidaya ikan untuk menyusun formulasi pakan sesuai target protein, sekaligus meningkatkan efisiensi waktu dan akurasi dalam perhitungan.

KESIMPULAN

Penelitian ini berhasil merancang dan membangun aplikasi Hitung Pakan berbasis Android dengan Metode *Pearson Square* untuk formulasi pakan ikan. Hasil uji coba menunjukkan bahwa perhitungan formulasi pakan menggunakan aplikasi memiliki kesesuaian penuh dengan perhitungan manual. Tidak adanya perbedaan hasil membuktikan bahwa aplikasi ini layak digunakan sebagai alat bantu praktis dalam penyusunan pakan ikan. Dengan demikian, aplikasi dapat membantu pembudidaya ikan dalam menghemat waktu, mengurangi kesalahan hitung, serta meningkatkan efisiensi proses formulasi pakan sesuai target protein yang diinginkan.

Saran untuk penelitian selanjutnya agar dapat melakukan pengembangan versi *multiplatform* (misalnya iOS atau berbasis web), agar aplikasi dapat diakses lebih luas oleh berbagai pengguna dan perlu peningkatan aspek edukasi melalui panduan interaktif atau tutorial dalam aplikasi, sehingga pembudidaya pemula lebih mudah memahami prinsip dasar formulasi pakan.

UCAPAN TERIMA KASIH

Ucapan Terimakasih diberikan kepada Universitas Veteran Bangun Nusantara selaku pemberi dana atau sponsor, penyumbang bahan dan sarana penelitian dalam penelitian ini.

REFERENSI

- [1] Z. Abidin, M. Junaidi, Paryono, N. Cokrowati, and S. Yuniarti, "Pertumbuhan dan konsumsi pakan ikan lele (*Clarias sp.*) yang diberi pakan berbahan baku lokal," *J. Depik*, vol. 4, no. 1, pp. 33–39, 2015, doi: 10.13170/depik.1.1.2360.
- [2] B. D. Glencross, "A feed is still only as good as its ingredients: An update on the nutritional research strategies for the optimal evaluation of ingredients for aquaculture feeds," *Aquac. Nutr.*, vol. 26, no. 6, pp. 1871–1883, 2020, doi: 10.1111/anu.13138.
- [3] F. Rahmatia, "Evaluasi Kecernaan Pakan Ikan Nila *Oreochromis niloticus* Pada Tiga Stadia Yang Berbeda," *J. Ilm. Satya Minabahari*, vol. 1, no. 2, pp. 43–51, 2016, doi: 10.53676/jism.v1i2.13.
- [4] S. M. Limbu, "The effects of on-farm produced feeds on growth, survival, yield and feed cost of juvenile African sharptooth catfish (*Clarias gariepinus*)," *Aquac. Fish.*, vol. 5, no. 1, pp. 58–64, 2020, doi: 10.1016/j.aaf.2019.07.002.
- [5] M. Bibin, A. Ardian, and A. N. Mecca, "Pelatihan Budidaya Maggot sebagai Alternatif Pakan Ikan di Desa Carawali," *MALLOMO J. Community Serv.*, vol. 1, no. 2, pp. 73–79, 2021, doi: 10.55678/malलोmo.v1i2.404.
- [6] A. Firdaus, "Perancangan dan Pembuatan Mesin Pelet Ikan Untuk Kelompok Usaha Ikan di Kelurahan Bukit Sangkal Palembang," in *Seminar Nasional AVoER XI*, 2019, pp. 639–643. doi: 10.13140/RG.2.2.29276.72326.
- [7] B. Wardono and A. S. Prabakusuma, "Analisis Usaha Pakan Ikan Mandiri Di Kabupaten Gunungkidul (Analysis of Independently Fish Feed Business in the District Gunungkidul)," *J. Kebijakan Sos. Ekon. Kelaut. dan Perikan.*, vol. 6, no. 1, pp. 75–85, 2016, [Online]. Available: <http://dx.doi.org/10.15578/jksekp.v6i1.1610>
- [8] N. Haryuni, Harliana, and Y. Alam, "Basic Knowledge of Animal Feed Formulation Nining," *Trop. Poult. Sci. Technol.*, vol. 1, no. 1, pp. 25–33, 2024.
- [9] Y. Tell, E. Abell, A. D. C. Mali, and M. S. Maure, "Formulasi Pakan Ikan Mandiri Berbahan Baku Lokal Ramah Lingkungan," *J. Inov. Penelit.*, vol. 3, no. 9, pp. 7603–7610, 2023. <https://doi.org/10.47492/jip.v3i9.2455>.
- [10] A. E. Wibowo, H. Tolle, and R. K. Dewi, "Pengembangan Aplikasi Mobile Penyusunan Ransum Pakan Ternak Sapi dan Kambing Menggunakan Framework Ionic," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 10, pp. 4296–4304, 2018, [Online]. Available:

- <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/2898>
- [11] D. Setiawan, A. Khairunnisa, and L. Trisnawati, "Penerapan Metode Pearson Square Pada Aplikasi Bebek Petelur Kuantitas Untung Untuk Menghitung Nutrisi Pakan Bebek," *J. Inf. Syst. Informatics Eng.*, vol. 9, no. 1, pp. 27–38, 2025, [Online]. Available: <https://doi.org/10.35145/joisie.v9i1.4847>
- [12] D. I. Maulana and D. Susandi, "Rancang Bangun Aplikasi Silase Pakan Ternak Domba Berbasis Android," *J. IKRA-ITH Inform.*, vol. 5, no. 1, pp. 94–100, 2021.
- [13] P. Destarianto, A. Sutirtoadi, and N. Faizah, "Desain Aplikasi Penentuan Formulasi Ransum Pada Ternak Sapi Potong," *J. Teknol. Inf. dan Terap.*, vol. 5, no. 1, pp. 43–48, 2019, doi: 10.25047/jtit.v5i1.78.
- [14] R. Akraman, C. Candiwan, and Y. Priyadi, "Pengkukuran Kesadaran Keamanan Informasi Dan Privasi Pada Pengguna Smartphone Android Di Indonesia," *J. Sist. Inf. Bisnis*, vol. 8, no. 2, pp. 115–122, 2018, doi: 10.21456/vol8iss2pp115-122.