

# **Sistem Informasi Formulasi Ransum Pakan Unggas Menggunakan Linear Programming**

## **Information System of Poultry Ration Formulation Using Linear Programming**

DIARDIAN FEBIANI (G64120113)<sup>1\*</sup>, IRMAN HERMADI, IDAT GALIH PERMANA

### **Abstrak**

Unggas akan berproduksi dengan baik jika nutrisi pada ransum pakannya seimbang dan lengkap sesuai dengan kebutuhannya. Penelitian ini dilakukan untuk membuat sistem informasi yang dapat memformulasikan komposisi berbagai bahan pembuatan ransum serta memenuhi kebutuhan nutrisi unggas sehingga menghasilkan ransum dengan harga minimum untuk membantu peternak unggas. Informasi yang digunakan pada penelitian ini yaitu berbagai bahan pakan dengan harga dan komposisi nutrisinya, serta berbagai jenis unggas dengan kebutuhan nutrisinya. Sistem informasi formulasi ransum pakan unggas ini dibuat dengan metode pengembangan perangkat lunak *extreme programming* (XP). Perhitungan formulasi ransum dengan *linear programming* diimplementasikan ke dalam kode sistem menggunakan perhitungan sesuai metode simpleks untuk menghasilkan harga minimum dari kombinasi berbagai bahan pakan ransum. Pengujian dilakukan dengan uji unit menggunakan *black-box testing* dan uji pengguna menggunakan perbandingan dengan aplikasi WinFeed 2.8.

Kata Kunci: *extreme programming*, formulasi ransum, *linear programming*, sistem informasi, unggas

### **Abstract**

*Poultry will produce well if the nutrients in its ration have a balanced and complete ratio according to its needs. This research aimed to build an information system to formulate the composition of ingredients for making ration that meets the poultry's nutrients requirements with the minimum price in order to aid poultry farmers. The data used in this research are feed ingredients including the price and nutrient compositions, and the varieties of poultry with its nutrient requirements. The information system of poultry feed ration formulation was developed using extreme programming (XP) software development method. The calculation of ration formulation was done using linear programming with simplex method to generate minimum price from feed ingredients combination of the ration. Unit testing was done for the system using black-box testing, and acceptance testing by comparing the result with WinFeed 2.8 application.*

*Keywords: extreme programming, information system, linear programming, poultry, ration formulation*

---

<sup>1</sup>Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor, Bogor 16680

\*Mahasiswa Program Studi S1 Ilmu Komputer, FMIPA-IPB; Surel: diardian\_feb12m@apps.ipb.ac.id

## PENDAHULUAN

### Latar Belakang

Unggas merupakan salah satu jenis hewan ternak yang banyak diproduksi karena potensi bisnisnya besar. Pakan merupakan kebutuhan utama bagi unggas untuk tetap hidup, tumbuh dan berkembang biak. Ransum adalah sekumpulan bahan makanan yang memenuhi persyaratan dan disusun dengan cara tertentu untuk memenuhi kebutuhan gizi unggas (Rasyaf 1989). Unggas berproduksi dengan baik jika nutrisi pada pakannya seimbang dan lengkap sesuai dengan kebutuhannya. Pentingnya keseimbangan nutrisi pada ransum menyebabkan dibutuhkan suatu sistem informasi mengenai formulasi ransum yang optimal. Formulasi ransum optimal mengandung arti bahwa nutrisi dalam ransum dapat memenuhi kebutuhan unggas secara maksimum, namun menghasilkan biaya minimum dari penggunaan berbagai bahan pakan penyusun ransum.

Penelitian sebelumnya membantu *user* khususnya peternak untuk memilih bahan pakan, mengecek komposisi ransum, mengecek kandungan nutrisi, memulai formulasi, dan melihat informasi ransum (Chandra 2015). Penelitian tersebut masih terbatas pada penentuan formulasi ransum dengan mencoba berbagai kemungkinan bahan pakan yang digunakan untuk memenuhi kebutuhan nutrisi unggas. Namun demikian, model formulasi yang menghasilkan harga ransum minimum dengan ketepatan komposisi nutrisi belum dilakukan.

Sistem informasi yang dapat memformulasikan komposisi nutrisi, bahan pakan, serta perhitungan harga minimum dari berbagai bahan pakan yang digunakan diperlukan untuk efisiensi pembuatan ransum pakan unggas. Pada penelitian ini, sistem informasi yang dapat melakukan perhitungan formulasi ransum pakan unggas dibuat dengan menggunakan rumusan *linear programming*. Hal tersebut dilakukan untuk mendapatkan formulasi yang memenuhi kebutuhan nutrisi unggas dengan harga minimum dari pembuatan ransum agar memaksimalkan keuntungan yang diperoleh peternak unggas.

### Perumusan Masalah

Penelitian ini dilakukan untuk merumuskan sistem informasi formulasi ransum pakan unggas yang menghasilkan komposisi nutrisi dan bahan pakan yang seimbang bagi unggas, serta harga minimum dari penyusunan bahan pakan pada ransum.

### Tujuan Penelitian

Tujuan penelitian ini yaitu mengembangkan sistem berbasis web yang dapat digunakan untuk membuat formulasi ransum pakan unggas dengan menentukan komposisi bahan yang terkandung dalam ransum pakan unggas, dan harga minimum kombinasi berbagai bahan yang memenuhi kebutuhan nutrisi unggas.

### Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat langsung membantu peternak unggas atau industri pakan unggas skala kecil dan menengah untuk membuat pakan unggas yang murah dengan penggunaan bahan yang memenuhi kebutuhan nutrisi unggas.

### Ruang Lingkup Penelitian

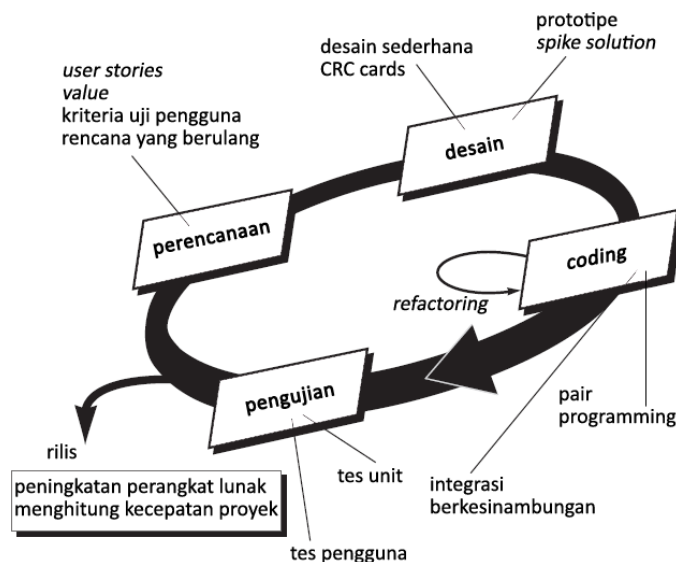
Lingkup penelitian ini adalah:

- 1 Informasi yang digunakan terdiri dari komposisi nutrisi berbagai bahan pakan, kebutuhan nutrisi berbagai jenis unggas, dan harga berbagai bahan pakan.
- 2 Jenis unggas yang disusun ransumnya yaitu ayam broiler, ayam petelur (*layer*), ayam kampung, puyuh, dan itik.

## METODE

### Tahapan Penelitian

Formulasi ransum dirancang untuk memecahkan persamaan dengan berbagai bahan yang memenuhi jumlah indikator kebutuhan gizi dan dengan harga terendah disesuaikan dengan formula ransum terbaik, memberikan potensi produksi maksimum hewan ternak dan unggas, dan meningkatkan efisiensi konversi pakan (Peng dan Li 2011). Sistem informasi formulasi ransum pakan unggas ini dibuat dengan metode pengembangan perangkat lunak *agile* yang mengizinkan tim pengembang untuk fokus ke perangkat lunak dibanding desain dan dokumentasinya (Sommerville 2011). Salah satu model dari *agile* yang digunakan pada penelitian ini adalah *extreme programming* (XP) dengan menggunakan pendekatan berorientasi obyek sebagai paradigma pengembangan dan mencakup seperangkat aturan dan praktik-praktik yang terjadi dalam konteks empat kerangka kegiatan: perencanaan, desain, *coding*, dan pengujian (Pressman 2010). Pengembangan sistem dilakukan dengan model XP untuk meningkatkan kualitas perangkat lunak, melakukan dokumentasi yang cepat, dan melakukan konfirmasi ke *user* setiap suatu fungsi diselesaikan. Proses XP dapat dilihat pada Gambar 1.



Gambar 1 Proses tahapan *extreme programming* (XP)

Untuk menyesuaikan pengembangan yang dilakukan oleh satu orang individu pada suatu penelitian, XP ini disesuaikan menjadi *Personal Extreme Programming* (PXP). PXP merupakan penerapan XP yang dimodifikasi sehingga dapat sesuai bagi kondisi *programmer* tunggal dan proses pengembangan perangkat lunak yang dibuat (Agarwal dan Umphress 2008). Pengembangan sistem informasi formulasi ransum pakan unggas ini dilakukan dengan tahapan-tahapan sebagai berikut.

#### 1 Perencanaan

Perencanaan merupakan serangkaian persiapan yang dilakukan untuk memperoleh informasi terkait dengan kebutuhan klien. Dalam penelitian ini, klien yang dimaksud adalah peternak unggas, baik peternak langsung maupun pengusaha bidang industri pakan unggas baik skala kecil maupun menengah. Pada tahap ini, pengembang sudah harus mengetahui, baik fitur maupun fungsi sistem dibuat dengan melakukan diskusi terkait kebutuhan pengguna. *User stories* menggambarkan keluaran yang dibutuhkan, fitur-fitur, dan fungsionalitas untuk perangkat lunak yang akan dibangun (Pressman 2010). Dari setiap keluaran, fitur, atau fungsi tersebut, didapatkan *value*

untuk menandakan bagian-bagian pentingnya. Dampak dari penggunaan metode XP adalah penambahan keluaran, fitur, dan fungsi, mengubah *value*, serta menghapus kebutuhan dapat dilakukan pada tahap perencanaan.

## 2 Desain

Dalam tahap ini, desain merupakan panduan pengimplementasian *user stories* dalam bentuk yang sederhana. Hal penting yang perlu disusun pada tahap ini yaitu *class-responsibility-collaborator* (CRC) *card* dan prototipe. Pengembang menggunakan CRC *card* selama kegiatan analisis untuk membantu mengidentifikasi kelas dan ruang lingkup masing-masing kelas (Satzinger *et al.* 2010). Pembuatan prototipe yang disebut *spike solution* direkomendasikan untuk dibuat pada tahap ini untuk membuat wujud nyata berupa tampilan yang dapat langsung dievaluasi dan diimplementasikan. Selain itu, untuk melengkapi dokumentasi pembuatan sistem maka ditambahkan pula penyusunan diagram *use case*, diagram kelas sebagai wujud nyata dari CRC *card*, *entity relationship diagram* (ERD), diagram aktivitas, dan diagram sekuens pada tahap ini.

## 3 Coding

Pada tahap ini, implementasi dilakukan dari segala hal yang telah dipersiapkan pada tahap perencanaan dan desain. Pada penelitian ini, *coding* dilakukan menggunakan editor teks *Sublime Text 2* dengan menggunakan bahasa pemrograman PHP dan Python, serta *framework* Yii sebagai pengaplikasian dari desain arsitektural MVC. Formula ransum yang terdapat dalam kode sistem dibentuk dengan fungsi dasar seperti pada Persamaan 1 berikut.

$$S = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

$$\text{Fungsi kendala : } a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m$$

$$\text{Asumsi : } x_1, x_2, \dots, x_n \geq 0 \quad (1)$$

dengan:

$S$  = fungsi tujuan yaitu nilai total harga minimum dari pembuatan ransum

$x_j$  = nilai penggunaan bahan pakan dalam bentuk persentase

$c_j$  = koefisien harga tiap pilihan bahan pakan

$a_{ij}$  = koefisien nilai komposisi nutrisi yang terkandung dalam suatu bahan pakan

$b_i$  = nilai pembatas berupa nilai minimum dan maksimum nutrisi yang dibutuhkan oleh unggas serta nilai minimum dan maksimum penggunaan bahan pakan

$m$  = jumlah pembatas

$n$  = jumlah bahan pakan yang digunakan dalam komposisi pembuatan ransum

## 4 Pengujian

Pada tahap ini, terdapat dua jenis pengujian yaitu pengujian unit dan pengujian pengguna. Pengujian unit fokus terhadap komponen perangkat lunak secara individual, melatih komponen antarmuka, struktur data, dan fungsionalitas sebagai usaha untuk mengetahui *error* (Pressman 2010). Pengujian pengguna dispesifikasi oleh klien dan fokus terhadap keseluruhan fitur dan fungsionalitas sistem yang dapat dilihat langsung dan diulas oleh klien (Pressman 2010).

## HASIL DAN PEMBAHASAN

SiMURAA merupakan nama dari sistem informasi formulasi ransum pakan unggas. Pengembangan SiMURAA menggunakan metode XP yang dimodifikasi menjadi PXP agar sesuai dengan pengembangan sistem yang dilakukan oleh satu orang. Tahapan pengembangan yang dilakukan meliputi perencanaan, desain, *coding*, dan pengujian.

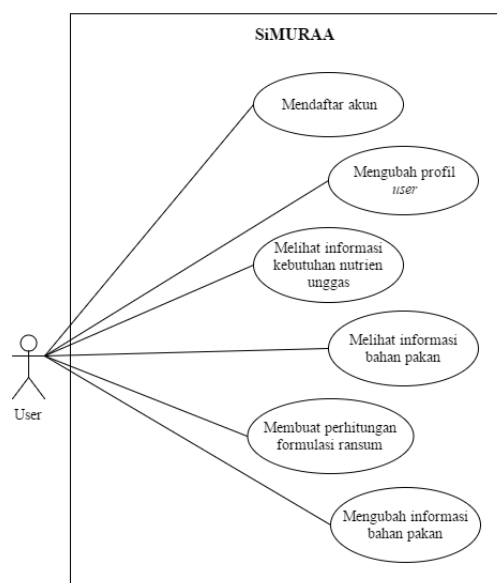
### Perencanaan

Pengembangan sistem SiMURAA diawali dengan perencanaan sesuai dengan kebutuhan *user*. *User* yang dapat menggunakan SiMURAA yaitu peternak unggas, pemilik industri pakan unggas, dan pelajar atau dosen yang ingin mempelajari sistem formulasi ransum menggunakan *linear programming*. Pada tahap ini, pertemuan dan komunikasi secara langsung dengan *feed formulator* dilakukan. Diskusi dilakukan bersama dengan *feed formulator*, termasuk pengambilan informasi yang diaplikasikan pada sistem. Pencarian informasi mengenai sistem perhitungan formulasi ransum yang sejenis juga dilakukan untuk menambah pengetahuan mengenai perhitungan formulasi ransum.

Fungsi utama yang menjadi *value* pada SiMURAA adalah perhitungan formulasi ransum dengan kriteria uji pengguna menggunakan aplikasi WinFeed 2.8. Aplikasi WinFeed merupakan aplikasi perhitungan formulasi ransum yang menggunakan *linear programming* sehingga serupa dengan perhitungan yang diterapkan dalam SiMURAA. Fungsi lain yang diinginkan ada pada sistem ini yaitu fungsi untuk melakukan *login* bagi *user*, melihat informasi mengenai unggas beserta kebutuhan nutriennya, melihat informasi berbagai bahan pakan beserta kandungan nutriennya, serta mengubah berbagai kriteria bahan pakan seperti harga, nilai maksimal dan minimal penggunaan bahan pakan untuk ransum. *User stories* yang didapat dalam tahap perencanaan ini diinterpretasikan dalam bentuk diagram *use case*.

### Desain

Setelah melakukan perencanaan dan mendapatkan berbagai informasi, desain dilakukan sebagai wujud konkret dari ide pengembangan sistem. Hal ini dilakukan untuk mengurangi risiko kesalahan pada tahap *coding* yang disebabkan ketidaksesuaian sistem yang dihasilkan oleh pengembang dengan sistem yang diinginkan oleh *user*. *User stories* yang telah disusun pada tahap perencanaan dapat digambarkan dalam bentuk diagram *user case* pada Gambar 2.



Gambar 2 Diagram *use case* SiMURAA

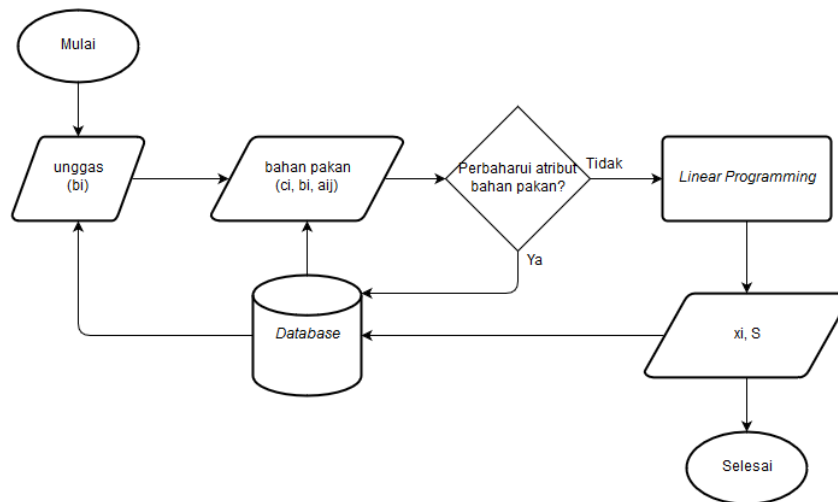
Untuk dapat menjalankan seluruh kegiatan pada *use case* selain mendaftar akun, user harus melakukan *login* terlebih dahulu untuk dapat masuk ke dalam SiMURAA. Setelah selesai melakukan aktivitas, *logout* dari sistem juga dapat dilakukan oleh *user*. Kebutuhan fungsional *user* SiMURAA sesuai dengan diagram *use case* dapat dilihat pada Tabel 1.

Tabel 1 Kebutuhan fungsional *user* SiMURAA

Kode fungsi	Kebutuhan fungsional	Keterangan
SM-001	Mendaftar akun	<i>User</i> melakukan pendaftaran dengan memberikan data yang telah ditentukan. Setelah mendaftar, <i>user</i> dapat masuk ke dalam sistem.
SM-002	Mengubah profil <i>user</i>	<i>User</i> dapat mengubah informasi diri yang telah diberikan ketika pendaftaran akun.
SM-003	Melihat informasi kebutuhan nutrisi unggas	<i>User</i> dapat melihat informasi seluruh unggas dan kebutuhan nutrisinya, serta menambah unggas dan kebutuhan nutrisinya.
SM-004	Melihat informasi bahan pakan	<i>User</i> dapat melihat informasi seluruh bahan pakan beserta harga dan kandungan nutrisinya.
SM-005	Membuat perhitungan formulasi ransum	<i>User</i> dapat melakukan perhitungan formulasi ransum dengan memilih jenis unggas dan bahan pakan yang akan digunakan.
SM-006	Mengubah informasi bahan pakan	Informasi bahan pakan secara <i>default</i> telah tersedia pada sistem. Namun, <i>user</i> dapat mengubah nilai minimum dan maksimum penggunaan bahan pakan dan harga bahan pakan pada saat sebelum perhitungan formulasi ransum.

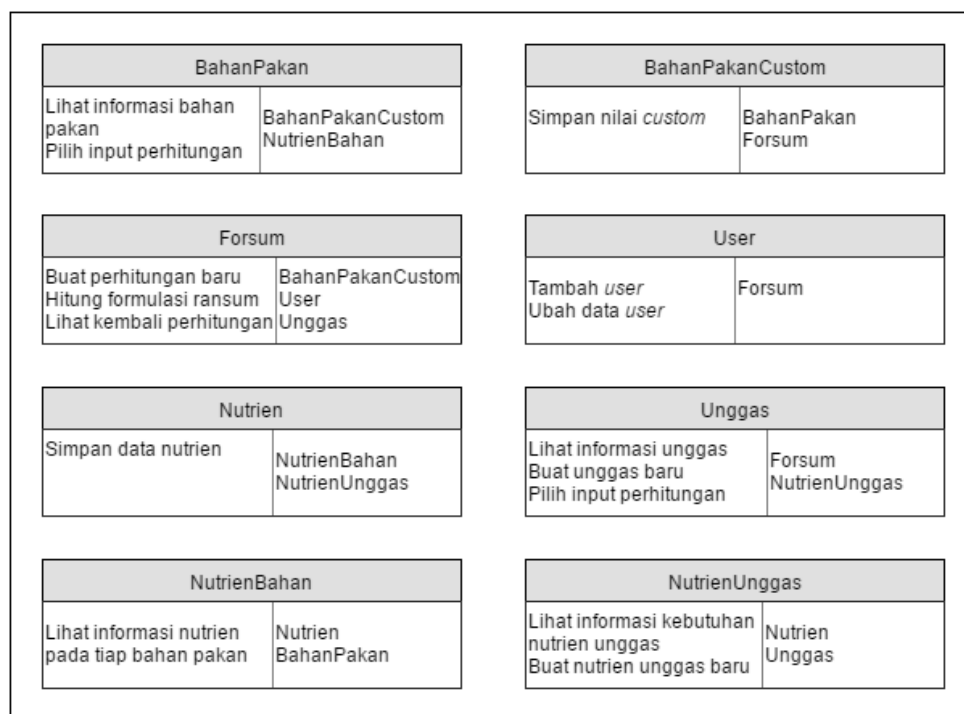
Fungsi utama dari SiMURAA yaitu perhitungan formulasi ransum menggunakan *linear programming*. *Linear programming* yang diaplikasikan dalam sistem membutuhkan *input* yang sudah ditentukan agar dapat menghasilkan *output* sesuai dengan tujuan yang ingin dihasilkan dari perhitungan. *Output* yang dihasilkan berupa nilai persentase penggunaan bahan-bahan penyusun ransum ( $x_i$ ) dan harga minimum dari seluruh bahan yang digunakan untuk pembuatan ransum ( $S$ ). *Input* yang diperlukan tersebut yaitu nilai minimum dan maksimum kebutuhan nutrisi suatu unggas ( $b_i$ ), harga tiap bahan pakan yang digunakan dalam pembuatan ransum ( $c_i$ ), nilai minimum dan maksimum penggunaan bahan pakan dalam ransum ( $b_i$ ), dan nutrisi-nutrisi yang terkandung dalam tiap bahan pakan yang digunakan dalam pembuatan ransum ( $a_{ij}$ ).

*User* dapat memulai perhitungan formulasi ransum dengan memilih nama unggas yang telah tersedia dalam *database*. Data kebutuhan nutrisi unggas diberikan ke sistem dan dijadikan data *input* bagi perhitungan *linear programming*. Selanjutnya, *user* memilih beberapa bahan pakan yang akan menjadi penyusun ransum. Data harga, nilai minimum dan maksimum penggunaan, dan kandungan nutrisi bahan pakan diberikan ke sistem dari *database* untuk digunakan sebagai *input*. *User* dapat mengubah harga serta nilai minimum dan maksimum penggunaan bahan pakan sebelum proses perhitungan. Setelah seluruh *input* lengkap, sistem melakukan perhitungan *linear programming* dengan menggunakan metode simpleks. Proses perhitungan dapat dikatakan selesai dan berhasil apabila menghasilkan seluruh nilai  $x_i$  dan  $S$ . Alur proses perhitungan menggunakan *linear programming* dapat dilihat pada Gambar 3.



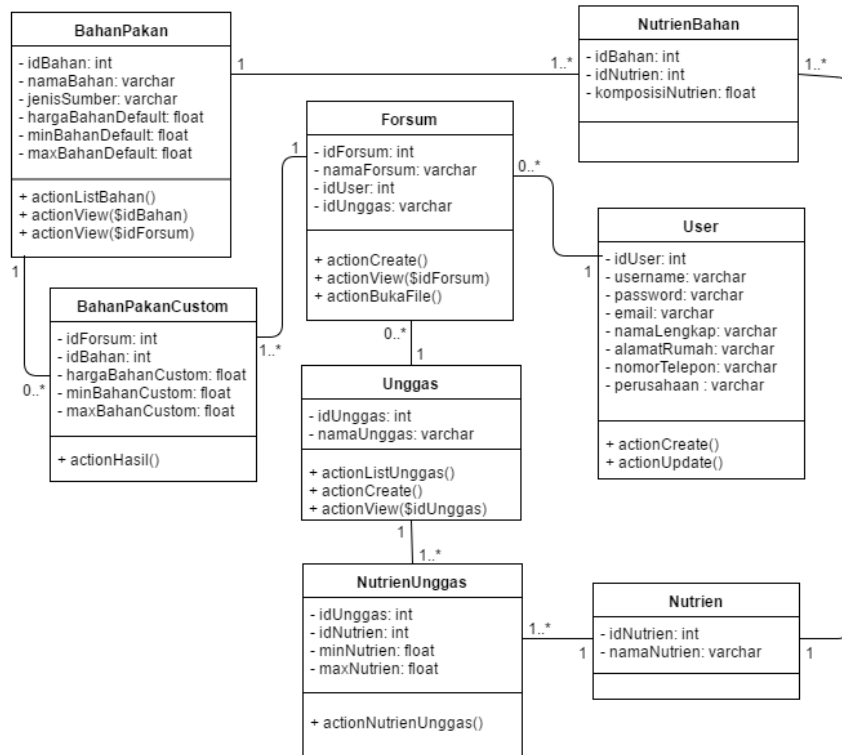
Gambar 3 Alur proses perhitungan menggunakan *linear programming*

Dua hal yang difokuskan pada desain dengan model XP yaitu pembuatan desain sederhana *CRC card* dan prototipe. Tabel dalam *CRC card* terdiri dari tiga bagian yaitu bagian atas sebagai nama kelas (*class*), bagian bawah kiri sebagai fungsi kelas (*responsibility*), dan bagian bawah kanan sebagai kelas lain yang berhubungan dengan kelas tersebut (*collaborator*). *CRC card* yang dibuat untuk SiMURAA dapat dilihat pada Gambar 4.



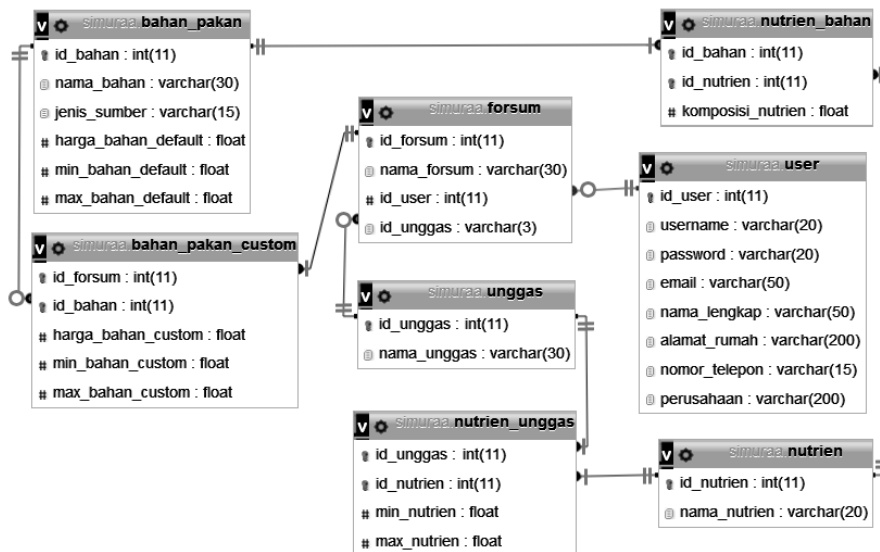
Gambar 4 *CRC Card* SiMURAA

Hasil nyata dari *CRC cards* yang telah dibuat yaitu adanya diagram kelas. Pembuatan diagram kelas dibagi menjadi tiga bentuk yaitu konseptual, spesifikasi, dan implementasi. Diagram kelas dalam bentuk implementasi dapat dilihat pada Gambar 5.



Gambar 5 Diagram kelas SiMURAA

Dengan dibuatnya diagram kelas, pembuatan *database* pada SQL dapat dilakukan menggunakan *object relational mapping* (ORM). Hasil yang didapatkan yaitu jumlah kelas pada diagram kelas sama dengan jumlah entitas pada ERD. Terdapat 8 kelas yang diubah menjadi 8 entitas yaitu kelas BahanPakan menjadi entitas bahan\_pakan, kelas BahanPakanCustom menjadi bahan\_pakan\_custom, kelas Unggas menjadi entitas unggas, kelas NutrienUnggas menjadi entitas nutrisi\_unggas, kelas Nutrien menjadi entitas nutrisi, kelas NutrienBahan menjadi entitas nutrisi\_bahan, kelas Forum menjadi entitas forum, dan kelas User menjadi entitas user. ERD SiMURAA dapat dilihat pada Gambar 6.



Gambar 6 ERD SiMURAA



Desain prototipe dibutuhkan sebagai gambaran antarmuka pengguna yang dapat dilihat langsung dan disesuaikan dengan keinginan pengguna. Contoh prototipe antarmuka halaman memulai perhitungan formulasi ransum dapat dilihat pada Gambar 7.

The image shows a web application prototype for SiMURAA. On the left is a dark sidebar menu with the title 'Menu' and several options: 'Beranda', 'Profil', 'Daftar Kebutuhan Nutrien Unggas', 'Daftar Bahan Pakan', 'Formulasi Ransum' (which is highlighted), and 'Tentang Kami'. The main content area has a light gray background and a title 'Buat Perhitungan Baru'. It contains two input fields: 'Nama forsum :' followed by a text box, and 'Nama Unggas :' followed by a dropdown menu. Below these fields is a dark button labeled 'Buat Perhitungan'. At the bottom right of the main area, there is a small copyright notice: '© 2016 SiMURAA'.

Gambar 7 Prototipe halaman memulai perhitungan formulasi ransum

### Coding

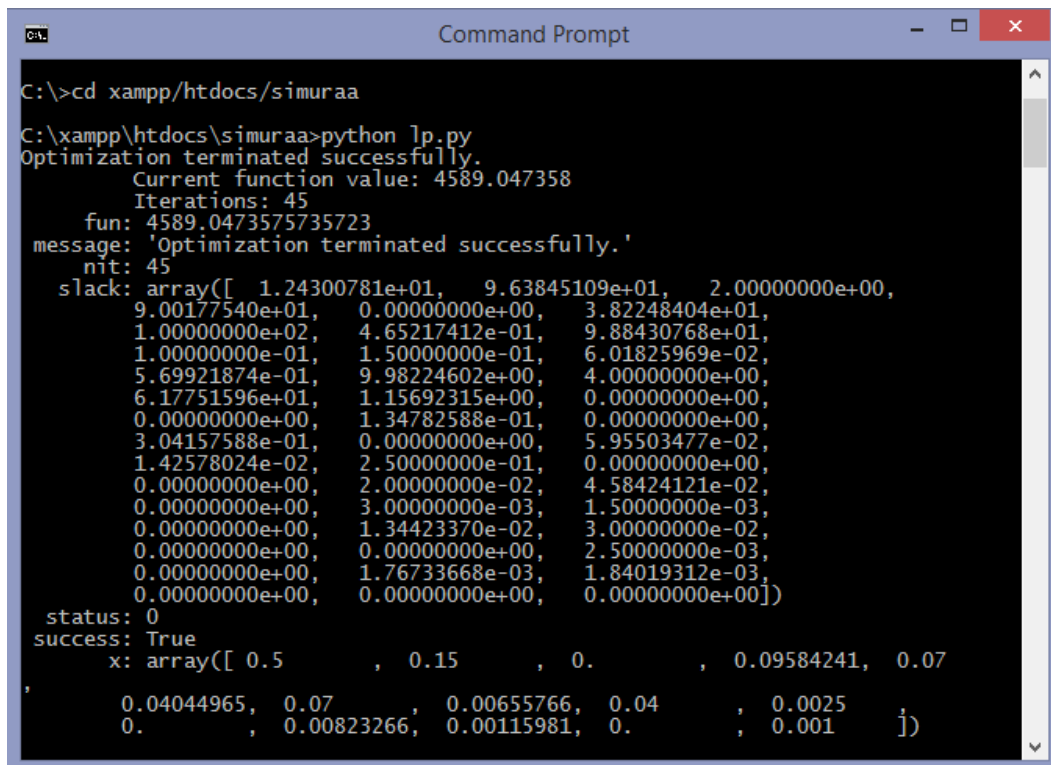
Tiga hal penting dalam *coding* pada XP yaitu *pair programming*, *refactoring*, dan integrasi berkesinambungan. Konsep *pair programming* dihilangkan dengan tetap mempertimbangkan peran yang harus dilakukan oleh 2 *programmer* pada *pair programming*. Peran sebagai seseorang yang dapat memberi masukan terhadap hasil *coding* dapat dilakukan oleh pembimbing penelitian dan *feed formulator*. *Refactoring* pada penelitian ini dapat dilakukan dengan merapikan atau memperbaiki struktur kode sistem yang telah dibuat untuk mengurangi risiko kesalahan pada program. Integrasi berkesinambungan pada konsep XP merupakan dampak dari *pair programming*. *Coding* yang dilakukan oleh satu orang menyebabkan *pair programming* dihilangkan pada penelitian ini dan integrasi berkesinambungan kurang perlu untuk dilakukan.

Pengembangan SiMURAA yang berbasis web menggunakan bahasa PHP dengan *framework* Yii. Terdapat 6 menu pada SiMURAA yaitu Beranda, Profil, Daftar Kebutuhan Nutrien Unggas, Daftar Bahan Pakan, Formulasi Ransum, dan Tentang Kami. Pada menu Profil, *user* dapat melihat profil diri dan mengubahnya. Pada menu Daftar Kebutuhan Nutrien Unggas, *user* dapat melihat informasi mengenai jenis-jenis unggas beserta nutrisi yang dibutuhkan unggas tersebut. Pada menu Daftar Bahan Pakan, *user* dapat melihat daftar bahan pakan beserta nutrisi yang dikandungnya. Pada menu Formulasi Ransum, *user* dapat melakukan perhitungan formulasi ransum yang menjadi fungsi utama pada SiMURAA.

Untuk mengaplikasikan *linear programming* di dalam SiMURAA, perhitungan dilakukan menggunakan bahasa Python yang diintegrasikan dengan bahasa utama sistem yaitu PHP. Dengan menggunakan *SciPy Library* modul `scipy.optimize.linprog` yang merupakan *open source library*, optimasi perhitungan *linear programming* dapat dilakukan. Kendala pertama yang dapat diterima yaitu pertidaksamaan dengan ruas kiri kurang dari atau sama dengan ( $\leq$ ) ruas kanan. Kendala kedua yang dapat diterima yaitu

persamaan linear dengan ruas kiri sama dengan ( $=$ ) ruas kanan. Namun, pada perhitungan ini terdapat pertidaksamaan dengan ruas kiri lebih dari atau sama dengan ( $\geq$ ) ruas kanan. Tanda pertidaksamaan lebih dari atau sama dengan ( $\geq$ ) harus diubah menjadi kurang dari atau sama dengan ( $\leq$ ) dengan cara mengalikan pertidaksamaan dengan negatif sehingga tanda pertidaksamaan berubah.

Sistem mengambil data-data dari *database* yang dibutuhkan sebagai masukan untuk menghasilkan keluaran harga termurah dan komposisi penggunaan tiap bahan pakan penyusun ransum. Untuk dapat mengeluarkan hasil, masukan yang telah disebutkan pada Persamaan 1 yaitu  $c_j$ ,  $a_{ij}$ , dan  $b_i$  harus dilengkapi. Jika perhitungan menemukan solusi fisibel atau memenuhi seluruh kendala maka hasil perhitungan yang dieksekusi di *Command Prompt* menghasilkan keluaran yang dapat dilihat pada Gambar 8.



```

C:\>cd xampp\htdocs\simuraa
C:\xampp\htdocs\simuraa>python lp.py
Optimization terminated successfully.
    Current function value: 4589.047358
    Iterations: 45
    fun: 4589.0473575735723
message: 'Optimization terminated successfully.'
nit: 45
slack: array([ 1.24300781e+01,  9.63845109e+01,  2.00000000e+00,
 9.00177540e+01,  0.00000000e+00,  3.82248404e+01,
1.00000000e+02,  4.65217412e-01,  9.88430768e+01,
1.00000000e-01,  1.50000000e-01,  6.01825969e-02,
5.69921874e-01,  9.98224602e+00,  4.00000000e+00,
6.17751596e+01,  1.15692315e+00,  0.00000000e+00,
0.00000000e+00,  1.34782588e-01,  0.00000000e+00,
3.04157588e-01,  0.00000000e+00,  5.95503477e-02,
1.42578024e-02,  2.50000000e-01,  0.00000000e+00,
0.00000000e+00,  2.00000000e-02,  4.58424121e-02,
0.00000000e+00,  3.00000000e-03,  1.50000000e-03,
0.00000000e+00,  1.34423370e-02,  3.00000000e-02,
0.00000000e+00,  0.00000000e+00,  2.50000000e-03,
0.00000000e+00,  1.76733668e-03,  1.84019312e-03,
0.00000000e+00,  0.00000000e+00,  0.00000000e+00])
status: 0
success: True
x: array([ 0.5      ,  0.15     ,  0.       ,  0.09584241,  0.07
 0.04044965,  0.07       ,  0.00655766,  0.04       ,  0.0025
 0.       ,  0.00823266,  0.00115981,  0.       ,  0.001
])

```

Gambar 8 Hasil perhitungan yang dieksekusi di *Command Prompt*

Dari hasil tersebut, nilai yang ditampilkan pada SiMURAA yaitu *fun* dan *x*. Nilai *fun* merupakan nilai dari fungsi tujuan atau *S* yaitu harga termurah pembuatan ransum yang disusun dari berbagai bahan pakan senilai Rp4 589. *Array x* yang dihasilkan merupakan nilai komposisi dari tiap bahan pakan yang digunakan sesuai urutan pemilihan bahan pakan. Contoh pada Gambar 9 menunjukkan bahwa bahan pakan yang dipilih yaitu 15 bahan pakan. Penggunaan bahan pakan terbanyak yaitu bahan pakan ke-1 sebanyak 0.5 atau setengah dari seluruh komposisi bahan penyusun ransum. Bahan pakan yang tidak digunakan atau bernilai 0 yaitu bahan pakan ke-3, ke-11, dan ke-14. Nilai *nit* merupakan jumlah iterasi yang dilakukan pada perhitungan. *Array slack* merupakan nilai-nilai dari variabel *slack* atau variabel yang ditambahkan di tiap kendala untuk mengubah kendala pertidaksamaan menjadi kendala persamaan.

### Pengujian

Pengujian pada penelitian ini dilakukan terhadap dua hal yaitu tes unit dan tes pengguna. Tes unit dilakukan menggunakan *black-box testing*. Fungsi yang ada diuji

dengan menguji masukan dan keluaran untuk menentukan keberhasilan sistem yang telah dibuat. Tabel pengujian menggunakan *black-box testing* dapat dilihat pada Tabel 2.

Tabel 2 Hasil pengujian menggunakan *black-box testing*

Kasus uji	Nilai input	Skenario uji	Hasil yang diharapkan	Hasil uji
Mendaftar akun	Benar	<i>User</i> mendaftar akun sesuai dengan formulir yang disediakan pada halaman daftar akun	<i>User</i> berhasil mendaftar dan data <i>user</i> masuk ke <i>database</i>	Berhasil
	Salah	<i>User</i> mendaftar akun namun tidak mengisi formulir pada halaman daftar akun	<i>User</i> tidak dapat mendaftar dan muncul pesan untuk tidak mengosongkan formulir	Berhasil
Mengubah profil <i>user</i>	Benar	<i>User</i> mengubah beberapa informasi diri pada halaman ubah profil	<i>User</i> berhasil mengubah informasi dirinya dan <i>database user</i> tersebut berubah	Berhasil
Melihat informasi kebutuhan nutrisi unggas	Benar	<i>User</i> melakukan klik pada menu daftar kebutuhan nutrisi unggas	<i>User</i> berhasil melihat informasi unggas beserta kebutuhan nutriennya	Berhasil
Melihat informasi bahan pakan	Benar	<i>User</i> melakukan klik pada menu daftar bahan pakan	<i>User</i> berhasil melihat informasi bahan pakan beserta kandungan nutriennya	Berhasil
Membuat perhitungan formulasi ransum	Benar	<i>User</i> melakukan perhitungan baru dengan memilih jenis unggas dan bahan pakan yang akan digunakan pada halaman buat perhitungan baru	<i>User</i> berhasil mendapatkan hasil berupa harga termurah dari pembuatan ransum dan nilai persentase penggunaan bahan pakan yang telah dipilih	Berhasil
Mengubah informasi bahan pakan	Benar	<i>User</i> mengubah <i>default</i> nilai minimal dan maksimal dengan nilai 0-1 pada halaman kriteria bahan pakan	<i>User</i> berhasil melanjutkan perhitungan dengan mendapatkan hasil perhitungan formulasi ransum	Berhasil
	Salah	<i>User</i> mengubah <i>default</i> nilai maksimal dengan nilai 5 pada halaman kriteria bahan pakan	<i>User</i> mendapat pesan untuk mengubah nilai maksimal menjadi kurang dari 100	Berhasil

Sementara itu, tes pengguna diuji melalui perbandingan SiMURAA dengan WinFeed 2.8. Hal ini lebih merujuk pada perbandingan hasil perhitungan menggunakan *linear programming* yang digunakan SiMURAA dan WinFeed 2.8. Pengujian perbandingan dilakukan 5 kali dengan penggunaan bahan pakan dan pembatasan kebutuhan nutrisi unggas yang berbeda. Pembatas bahan pakan dan kebutuhan nutrisi berupa nilai minimal (min) dan nilai maksimal (max).

Pengujian dilakukan dengan penggunaan bahan pakan serta pembatasan penggunaan bahan pakan yang berbeda. Bahan pakan yang digunakan beserta nilai pembatasan penggunaannya pada 5 kali pengujian tersebut dapat dilihat pada Tabel 3. Kebutuhan nutrisi yang dibatasi pada 5 kali pengujian dapat dilihat pada Tabel 4. Setelah penentuan bahan pakan yang digunakan beserta batasannya, dan pembatasan kebutuhan nutrisi maka perhitungan dilakukan menggunakan SiMURAA (SM) dan WinFeed 2.8 (WF). Hasil yang diperoleh yaitu harga dan komposisi penggunaan bahan pakan. Harga ransum dan komposisi penggunaan bahan pakan yang diperoleh pada 5 kali pengujian dapat dilihat pada Tabel 5.

Tabel 3 Bahan pakan dan batasan penggunaannya pada pengujian

Bahan Pakan	Pengujian 1		Pengujian 2		Pengujian 3		Pengujian 4		Pengujian 5	
	Min (%)	Max (%)	Min (%)	Max (%)	Min (%)	Max (%)	Min (%)	Max (%)	Min (%)	Max (%)
Jagung lokal	25	50	25	50	25	50	25	40	25	40
Dedak halus	0	15	0	15	0	15	0	15	0	15
Pollard	0	6	0	6	0	6	0	6	0	6
Bungkil kedelai	5	40	5	40	5	40	5	40	5	40
CGM	0	7	0	7	0	7	0	7	0	7
MBM	0	7	0	7	0	7	0	7	0	7
Tepung ikan	0	8	0	8	0	8	0	8	0	8
CPO	1	4	1	4	1	4	1	4	1	4
CaCO <sub>3</sub>	0	2	0	2	0	2	0	2	0	2
DCP	0	1	0	1	0	1	0	1	0	1
L-Lysin	-	-	-	-	0.1	0.3	0.1	0.3	0.1	0.3
DL-Methionine	-	-	-	-	0.1	0.3	0.1	0.3	0.1	0.3

Tabel 4 Batasan kebutuhan nutrisi pada pengujian

Nutrien	Pengujian 1		Pengujian 2		Pengujian 3		Pengujian 4		Pengujian 5	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Bahan kering (%)	86	100	86	100	86	100	86	100	86	100
Abu (%)	-	-	-	-	-	-	-	-	0	4.9
Protein kasar (%)	20	23	20	23	20	23	20	23	20	23
Lemak kasar (%)	-	-	-	-	-	-	5	8.5	5	8.5
Serat kasar (%)	-	-	0	4	0	4	0	4	0	4
BetaN (%)	-	-	-	-	-	-	-	-	0	57
Energi metabolisme (kkal/kg)	3 000	3 200	3 000	3 200	3 000	3 200	3 000	3 200	3 000	3 200
Kalsium (%)	-	-	0.9	1.5	0.9	1.5	0.9	1.5	0.9	1.5
Phospor total (%)	0.6	1.0	0.6	1.0	0.6	1.0	0.6	1.0	0.6	1.0
Phospor tersedia (%)	-	-	-	-	-	-	0.4	0.5	0.4	0.5
Lysin (%)	-	-	-	-	0.5	1.3	0.5	1.3	0.5	1.3
Methionin (%)	-	-	-	-	0.3	0.7	0.3	0.7	0.3	0.7

Tabel 5 Harga ransum dan komposisi penggunaan bahan pakan pada pengujian

	Pengujian 1		Pengujian 2		Pengujian 3		Pengujian 4		Pengujian 5	
Hasil	SM	WF	SM	WF	SM	WF	SM	WF	SM	WF
Harga (Rp)	4 344	4 344	4 379	4 379	4 477	4 477	-	5 223	5 293	5 293
Jagung lokal (%)	50	50	50	50	50	50	-	40	40	40
Dedak halus (%)	15	15	15	15	15	15	-	15	15	15
Pollar (%)	6	6	6	6	6	6	-	0	0	0
Bungkil kedelai (%)	5	5	9.17	9.17	9.36	9.36	-	25.87	25.73	25.73
CGM (%)	7	7	7	7	7	7	-	7	7	7
MBM (%)	5.96	5.96	1.89	1.89	1.34	1.34	-	1.06	1.56	1.56
Tepung ikan (%)	8	8	8	8	8	8	-	3.86	3.45	3.45
CPO (%)	1.04	1.04	1.24	1.24	1.26	1.26	-	3.97	3.95	3.95
CaCO <sub>3</sub> (%)	2	2	1.71	1.71	1.84	1.84	-	2	1.95	1.95
DCP (%)	0	0	0	0	0	0	-	1	1	1
L-Lysin (%)	-	-	-	-	0.1	0.1	-	0.1	0.1	0.1
DL-Methionine (%)	-	-	-	-	0.1	0.1	-	0.15	0.25	0.25

Pada hasil perbandingan SiMURAA dan WinFeed 2.8, dapat dikatakan bahwa hasil yang dikeluarkan keduanya memiliki nilai yang sama. Hal tersebut dapat dilihat dari hasil pada pengujian 1, 2, 3, dan 5. Namun, pada pengujian 4, SiMURAA tidak dapat mengeluarkan hasil seperti yang dikeluarkan pada WinFeed 2.8. Hal ini diduga karena keakurasian SiMURAA yang kurang menoleransi hasil nilai persentase penggunaan bahan pakan dan nilai total kandungan tiap nutrisi jika kurang atau lebih dari batasan yang telah ditentukan.

## SIMPULAN

Sistem formulasi ransum pakan unggas berbasis web yang dinamakan SiMURAA menggunakan perhitungan *linear programming* dengan metode simpleks dan metode pengembangan sistem *personal extreme programming*. SiMURAA dapat menghasilkan ransum unggas yang seimbang dan murah (*least cost balance ration*). Fitur-fitur yang dapat dilakukan oleh *user* yaitu mendaftar akun, mengubah profil *user*, melihat informasi kebutuhan nutrisi unggas, melihat informasi bahan pakan, membuat perhitungan formulasi ransum, dan mengubah informasi bahan pakan. Pengujian SiMURAA dibandingkan dengan program profesional formulasi ransum WinFeed 2.8 memberikan hasil formulasi ransum yang sama.

## SARAN

SiMURAA dapat dikembangkan menjadi lebih praktis atau *usable* bagi peternak atau pemilik industri pakan unggas dengan tampilan berbasis *mobile* pada *tablet* atau *smartphone*, ditambah fitur melihat kembali perhitungan formulasi ransum yang sudah pernah dilakukan, dan ditambah fitur berkiriman pesan ke sesama peternak unggas. SiMURAA dapat dibuat tampilannya menjadi lebih *user friendly* dan dapat memformulasikan berbagai ransum sekaligus atau *multiblend*. Nilai *default* dari atribut bahan pakan yang digunakan dalam perhitungan disarankan bukan berasal dari nilai *default* dari *database* sistem, melainkan berasal dari nilai *default* yang telah dimasukkan oleh *user*. Harga bahan pakan diharapkan dapat mengikuti harga pasar dan memiliki sejarahnya. Selain itu, keakurasian SiMURAA dibuat agar lebih menoleransi hasil jika nilai yang dikeluarkan kurang atau lebih dari batasan yang telah ditentukan sehingga perhitungan mengeluarkan hasil.

## DAFTAR PUSTAKA

- Agarwal R, Umphress D. 2008. Extreme programming for a single person team. Di dalam: *Proceedings of the 46th Annual Southeast Regional Conference on XX - ACM-SE 46*; 2008 Mar 28-29; Auburn, USA. New York (US): ACM Press. hlm 82-87.
- Chandra. 2015. Sistem informasi formulasi ransum pakan ternak dengan model pengembangan *prototyping* [skripsi]. Bogor (ID): Institut Pertanian Bogor.
- Peng Y, Li Q. 2011. The decision-making for feed formula in animal husbandry breeding based on the revised simplex method. Di dalam: *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*; 2011 Agu 8-10; Zhengzhou, China. Piscataway (US): IEEE. hlm 1648 - 1651.
- Pressman RS. 2010. *Software Engineering: A Practitioner's Approach*. Ed ke-7. New York (US): McGraw-Hill.
- Rasyaf M. 1989. *Memelihara Ayam Buras*. Yogyakarta (ID): Kanisius.
- Satzinger JW, Jackson RB, Burd SD. 2010. *System Analysis and Design in a Changing World*. Ed ke-5. Boston (US): Course Technology.
- Sommerville I. 2011. *Software Engineering*. Ed ke-9. Boston (US): Addison-Wesley.
- Thie PR, Keough GE. 2008. *An Introduction to Linear Programming and Game Theory*. Ed ke-3. Hoboken (US): John Wiley & Sons.