

**PENGEMBANGAN SISTEM FORMULASI RANSUM UNTUK KEBUTUHAN TERNAK RUMINANSIA MENGGUNAKAN *LINEAR PROGRAMMING***

**DEPARTEMEN ILMU KOMPUTER**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**INSTITUT PERTANIAN BOGOR**

**BOGOR**

**2017**

**ALIN NUR ALIFAH**

**PERNYATAAN MENGENAI SKRIPSI DAN  
SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA**

Dengan ini saya menyatakan bahwa skripsi berjudul Pengembangan Sistem Formulasi Ransum untuk Kebutuhan Ternak Ruminansia Menggunakan *Linier Programming* adalah benar karya saya dengan arahan dari komisi pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Desember 2017

*Alin Nur Alifah*

NIM G64154068

**ABSTRAK**

ALIN NUR ALIFAH. Pengembangan Sistem Formulasi Ransum untuk Kebutuhan Ternak Ruminansia Menggunakan *Linier Programming*. Dibimbing oleh IRMAN HERMADI dan IDAT GALIH PERMANA.

Formulasi ransum merupakan aspek yang sangat esensial dalam menyeimbangkan nutrisi bagi hewan ternak dengan tujuan mendapatkan harga minimum berdasar pada kandungan nutrisi pakan hewan. Oleh karena itu peternak dituntut untuk mampu menyusun suatu formula ransum yang ekonomis tanpa mengabaikan faktor kebutuhan nutrisi ternak. Penelitian ini bertujuan untuk membuat suatu sistem pendukung pengambilan keputusan yang mampu melakukan formulasi ransum dengan mengadopsi metode pemrograman linier. Sistem dirancang dalam perograman web dan *mobile* sehingga formulasi ransum dapat dilakukan oleh pengguna di peternakan dan pengolahan data dapat dilakukan menggunakan peramban. Metode pengembangan sistem yang dilakukan adalah prototype dengan evaluasi menggunakan Mean Absolute Percentage Error (MAPE) dan Mean Square Error (MSE).

Kata kunci: formulasi ransum, *linier programming, prototype*, ternak ruminansia

**ABSTRACT**

ALIN NUR ALIFAH. Development of Ration Formulation System for Ruminanted Livestock Needs Using Linear Programming. Supervised by IRMAN HERMADI and IDAT GALIH PERMANA.

Feed formulation is an essential aspect in balancing nutrients for livestock in order to get a minimum price based on the nutrient content of livestock feed. Therefore, farmers are required to be able to compile an economical feed formulation without ignoring nutritional needs factors of livestock. This research aims to create/develop a decision support system that is capable of compile feed formulation by adopting the method of linear programming. The system is designed in web and mobile programming so that the feed formulation can be conducted by users in farms and data processing can be done using a web browser. The development method used is prototype with evaluation of Mean Absolute Percentage Error (MAPE) and Mean Square Error (MSE).

Keywords: feed formulation, linier programming, prototype, ruminanted livestock

Skripsi  
sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Ilmu Komputer  
pada  
Departemen Ilmu Komputer

**ALIN NUR ALIFAH**

**PENGEMBANGAN SISTEM FORMULASI RANSUM UNTUK KEBUTUHAN TERNAK RUMINANSIA MENGGUNAKAN *LINEAR PROGRAMMING***

**DEPARTEMEN ILMU KOMPUTER**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**INSTITUT PERTANIAN BOGOR**

**BOGOR**

**2017**

Penguji: 1 Dr Ir Agus Buono, MSi MKom

Judul Skripsi : Pengembangan Sistem Formulasi Ransum untuk Kebutuhan Ternak Ruminansia Menggunakan *Linier Programming*

Nama : Alin Nur Alifah

NIM : G64154068

Disetujui oleh

|  |  |
| --- | --- |
| Irman Hermadi, SKom MS PhD  Pembimbing I | Dr Ir Idat Galih Permana, MSc  Pembimbing II |

Diketahui oleh

Dr Ir Agus Buono, MSi MKom

Ketua Departemen

Tanggal Lulus:

**PRAKATA**

Puji dan syukur penulis panjatkan kepada Allah *subhanahu wa ta’ala* atas segala karunia-Nya sehingga tugas akhir ini berhasil diselesaikan. Penelitian yang menghasilkan sistem formulasi ransum pakan ternak ruminansia ini diharapkan dapat berguna secara langsung bagi para peternak.

Terima kasih penulis ucapkan kepada:

1. Bapak Dr Ir Agus Buono, MSi MKom selaku Ketua Departemen Ilmu Komputer FMIPA IPB sekaligus penguji.
2. Bapak Irman Hermadi, SKom MS PhD dan Bapak Dr Ir Idat Galih Permana, MSc selaku pembimbing yang telah memberikan banyak ilmu dan saran yang membangun bagi penulis.
3. Seluruh dosen IPB yang telah memberikan ilmu kepada penulis selama penulis menempuh pendidikan S1; staf tata usaha dan seluruh staf pegawai Departemen Ilmu Komputer FMIPA IPB yang telah membantu penulis dalam menyelesaikan berbagai hal administrasi.
4. Bapak Dr Ir Tatag Budiardi, MSi dan Ibu Dr Ir Kurnia Suci Indraningsih, MSi selaku orang tua yang sangat penulis cintai dan kakak Titania Aulia, SKPm MSi yang penulis sayangi. Terima kasih untuk ketulusannya dalam memberi doa, dorongan dan semangat untuk terus maju.
5. Kekasih tersayang yang selalu menghibur, menemani, serta memberi dukungan dan semangat.
6. Seluruh teman-teman di IPB terutama Dwia Pungky Arumdani, Defriani Putri, Widuri Putri Wulansari, Wieke Aulia Putri, Shellafuri Biru Mardika, Ardhi Ma’arik, Ramdhan Abdul Ghifari, Irfan Zidny, Alfandio Grasheldi, Samad, Muammar Haikal Perdana, dan Muhammad Fariz yang memberi semangat dan membagikan ilmunya selama penulis menjalani masa perkuliahan dan penelitian.

Semoga skripsi ini bermanfaat dan menambah wawasan bagi pembacanya.

Bogor, Desember 2017

*Alin Nur Alifah*

**DAFTAR ISI**

DAFTAR TABEL ix

DAFTAR GAMBAR ix

DAFTAR LAMPIRAN ix

[PENDAHULUAN 1](#_Toc462131081)

[Latar Belakang 1](#_Toc462131082)

[Perumusan Masalah 1](#_Toc462131083)

[Tujuan Penelitian 1](#_Toc462131084)

[Manfaat Penelitian 2](#_Toc462131085)

[Ruang Lingkup Penelitian 2](#_Toc462131086)

[TINJAUAN PUSTAKA 2](#_Toc462131087)

[*Extreme Programming* 2](#_Toc462131088)

[Formulasi Ransum 2](#_Toc462131089)

[*Linear Programming* 3](#_Toc462131090)

[*Model, View, Controller* (MVC) 3](#_Toc462131091)

[METODE 3](#_Toc462131092)

[Tahapan Penelitian 3](#_Toc462131093)

[Lingkungan Pengembangan 6](#_Toc462131094)

[HASIL DAN PEMBAHASAN 6](#_Toc462131095)

[Perencanaan 6](#_Toc462131096)

[Desain 7](#_Toc462131097)

[*Coding* 13](#_Toc462131098)

[Pengujian 16](#_Toc462131099)

[SIMPULAN DAN SARAN 20](#_Toc462131100)

[Simpulan 20](#_Toc462131101)

[Saran 20](#_Toc462131102)

[DAFTAR PUSTAKA 20](#_Toc462131103)

LAMPIRAN 22

RIWAYAT HIDUP 47

**DAFTAR TABEL**

1. [Kebutuhan fungsional *user* SiMURAA 8](#_Toc461176889)
2. [Hasil pengujian menggunakan *black-box testing* 16](#_Toc461176890)
3. [Bahan pakan dan batasan penggunaannya pada pengujian 17](#_Toc461176891)
4. [Batasan kebutuhan nutrien pada pengujian 18](#_Toc461176892)
5. [Harga ransum dan komposisi penggunaan bahan pakan pada pengujian 18](#_Toc461176893)

**DAFTAR GAMBAR**

1. [Proses tahapan *extreme programming* (XP) 4](#_Toc462304076)
2. [Diagram *use case* SiMURAA 7](#_Toc462304077)
3. [Alur proses perhitungan menggunakan *linear programming* 9](#_Toc462304078)
4. [CRC *Card* SiMURAA 9](#_Toc462304079)
5. [Diagram kelas SiMURAA 10](#_Toc462304080)
6. [ERD SiMURAA 11](#_Toc462304081)
7. [Prototipe halaman memulai perhitungan formulasi ransum 11](#_Toc462304082)
8. [Diagram aktivitas membuat perhitungan formulasi ransum 12](#_Toc462304083)
9. [Diagram sekuens membuat perhitungan formulasi ransum 13](#_Toc462304084)
10. [Halaman untuk memulai formulasi ransum 14](#_Toc462304085)
11. [Hasil perhitungan yang dieksekusi di *command prompt* 15](#_Toc462304086)
12. [Hasil perhitungan formulasi ransum menggunakan SiMURAA 19](#_Toc462304087)
13. [Hasil perhitungan formulasi ransum menggunakan Winfeed 2.8 19](#_Toc462304088)

**DAFTAR LAMPIRAN**

1. [Diagram kelas konseptual 22](#_Toc461176931)
2. [Diagram kelas spesifikasi 22](#_Toc461176932)
3. [Kamus data SiMURAA 23](#_Toc461176933)
4. [Prototipe seluruh fungsi pada SiMURAA 26](#_Toc461176934)
5. [Seluruh diagram aktivitas pada SiMURAA 32](#_Toc461176935)
6. [Seluruh diagram sekuens pada SiMURAA 38](#_Toc461176936)
7. [Seluruh halaman pada SiMURAA 41](#_Toc461176937)

# PENDAHULUAN

## Latar Belakang

Subsektor peternakan memiliki peranan penting dalam perekonomian Indonesia baik dalam pembentukan Produk Domestik Bruto (PDB) dan penyerapan tenaga kerja maupun dalam penyediaan bahan baku industri. Perannya dalam pertumbuhan ekonomi menunjukkan bahwa PDB peternakan triwulan I tahun 2005 tumbuh 5.8%. Kontribusi PDB subsektor peternakan terhadap sektor pertanian triwulan I tahun 2005 mencapai 13.2%. Sedangkan terhadap besaran PDB Nasional mencapai 2%. Dalam penyerapan tenaga kerja sub sektor peternakan juga mempunyai peranan yang sangat strategis. Menurut hasil sensus pertanian 2003 dari 24,86 juta Rumah Tangga Pertanian di pedesaan dan perkotaan, sekitar 22,63% merupakan Rumah Tangga Usaha Peternakan. Selain itu sub sektor peternakan juga berperan penting dalam penyediaan bahan baku bagi keperluan industri (Makka 2012).

Efisiensi produksi dalam suatu usaha peternakan menjadi faktor penentu keberhasilan peternakan. Efisiensi produksi dapat diwujudkan dengan pemberian pakan yang berkualitas dengan kuantitas yang memadai sesuai dengan kebutuhan ternak. Pakan merupakan salah satu aspek yang sangat penting dalam keberhasilan suatu usaha peternakan. Sehingga formulasi ransum dari sejumlah bahan pakan yang tersedia merupakan aspek yang sangat vital khususnya dalam rangka menyeimbangkan kandungan energi, protein dan nutrien lainnya (Jayanegara 2014). Berdasarkan sudut pandang ekonomi, biaya untuk pembelian pakan ternak merupakan biaya tertinggi dalam agribisnis perternakan. Sehingga biaya tersebut harus ditekan serendah mungkin agar tidak mengurangi pendapatan. Teknologi dapat menjadi jalan keluar dalam permasalahan tersebut, yaitu dengan mengaplikasikan teknologi formulasi pakan ternak yang efisien. Pakan ternak yang diramu dengan baik dan sesuai dengan kebutuhan ternak akan menekan biaya pembelian pakan serendah mungkin (Shiddieqy 2010).

Ransum yang murah dan berkualitas memerlukan suatu teknik atau metode dalam memformulasikannya. Formulasi ransum yang mudah digunakan, cepat, akurat dalam penentuan komposisi bahan dan mendapatkan biaya serendah mungkin dapat menggunakan metode *linier programming*. Selain metode *linier programming*, ada beberapa metode lain yang dapat digunakan, antara lain metode *trial and error*, *equation* dan *pearson’s square*. Diantara metode-metode tersebut, metode *linier programming* adalah yang paling sesuai untuk diterapkan sebagai metode formulasi ransum karena mampu menangani jumlah variabel yang banyak secara efisien (Muzayyanah 2013). Akan tetapi dalam penghitungan secara manual metode ini masih dirasa sangat sulit (Kusnandar 2004).

Penelitian tentang formulasi ransum ternak sapi potong sudah pernah dilakukan oleh Rahman (Rahman 2017). Peneliti mengembangkan sistem formulasi ransum ternak sapi potong berdasarkan nilai ADG (*average daily gain*) dan berat badan ternak menggunakan metode *linier programming*. Sistem tersebut dapat melakukan formulasi dengan kesamaan dan akurasi yang baik karna hasil perbandingan mendapatkan selisih 0. Sehingga pada penelitian ini diusulkan sebuah sistem pengembangan dari sistem informasi yang telah dikembangkan oleh Rahman (Rahman 2017) dengan hewan ternak mencakup seluruh ternak ruminansia.

## Perumusan Masalah

Berdasarkan uraian yang tercantum pada latar belakang maka perumusan masalah pada penelitian ini adalah :

1. Bagaimana menerapkan algoritme pemrograman linier untuk formulasi ransum ternak ruminansia?
2. Bagaimana hasil evaluasi ransum pada formulasi terhadap perhitungan pakar?

## Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mengembangkan sistem formulasi ransum menggunakan algoritme pemrograman linier untuk ternak ruminansia.
2. Melakukan evaluasi ransum hasil formulasi sistem dengan hasil formulasi pakar.

## Manfaat Penelitian

Hasil penelitian diharapkn dapat membantu para peternak dalam melakukan formulasi ransum secara cepat dan tepat.

## Ruang Lingkup Penelitian

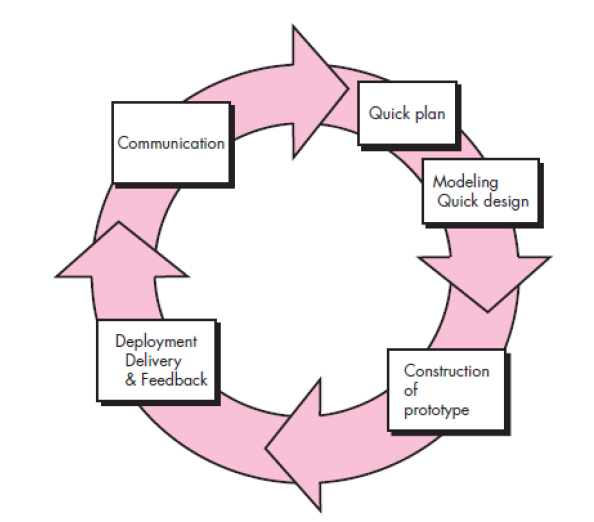
Lingkup dari penelitian ini adalah:

1 Sistem dikembangkan pada sistem berbasis web.

# METODE

## Tahapan Penelitian

Penelitian yang dilakukan mengikuti kaidah pengembangan sistem *prototype*. Metode *prototype* terdiri dari beberapa tahapan yaitu komunikasi (pengumpulan kebutuhan), perencanaan dan pemodelan cepat, pembuatan *prototype*, pengembangan sistem serta pengiriman hasil dan umpan balik (*deployment delivery* dan *feedback*) (Pressman 2010). Gambar 1 menunjukan tahapan proses pada metode *prototype*.



Gambar Tahapan metode *prototype* (Pressman 2010)

### Komunikasi

Tahapan ini mendefinisikan kebutuhan keseluruhan sistem. Mengidentifikasi proses bisnis, jenis ternak dan pakan yang akan digunakan. Jenis ternak yang digunakan adalah ternak ruminansia. Ternak ruminansia adalah jenis hewan ternak yang mampu mencerna pakan hijauan yang berserat tinggi dan pakan konsentrat seperti sapi, kerbau, domba dan kambing. Jenis pakan sendiri terbagi menjadi sumber protein, sumber energi, sumber vitamin dan sumber mineral yang dikelompokkan menjadi dua jenis yaitu pakan hijauan dan pakan konsentrat (Hidayat dan Mukhlas 2015). Sedangkan ransum diartikan sebagai satu atau beberapa jenis pakan yang diberikan kepada hewan ternak dan dapat memenuhi zat gizi yang dibutuhkan ternak untuk berbagai fungsi tubuhnya (Muhammad et al. 2014). Pada tahapan ini akan dilakukan komunikasi antara peternak dan pengembang untuk kebutuhan *transfer knowledge* dari pakar kepada pengembang.

### Perancangan dan Pemodelan

Menurut Pressman (2010) setelah tahap komunikasi dilakukan, selanjutnya adalah tahap perancangan dan pemodelan sistem. Perancangan dan pemodelan yang dibuat disesuaikan dengan kebutuhan sistem yang telah didefinisikan pada tahap komunikasi. Perancangan dapat dideskripsikan melalui tabel kebutuhan fungsional sistem dan basisdata. Pemodelan disepakati menggunakan model *linier programming* dengan metode simpleks. *Linier programming* merupakan metode matematika dalam mengalokasikan sumber daya yang terbatas untuk mencapai suatu tujuan seperti memaksimumkan keuntungan atau meminimumkan biaya. Pemrograman linier banyak diterapkan dalam masalah ekonomi, industri, militer dan sosial. Dalam formulasi ransum dapat digunakan untuk mendapatkan harga seminimal mungkin (Wirdasari 2009). Penelitian Hidayat dan Mukhlas (2015) menjelaskan persamaan matematis pemrograman linier bertujuan untuk meminimumkan dapat dilihat pada persamaan dibawah ini.

:

:

(1)

dengan:

*Z =* fungsi tujuan yaitu nilai total harga minimum dari pembuatan ransum

= nilai penggunaan bahan pakan dalam bentuk persentase

= koefisien harga tiap pilihan bahan pakan

= koefisien nilai komposisi nutrien yang terkandung dalam suatu bahan pakan

= nilai pembatas berupa nilai minimum dan maksimum nutrien 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

Menurut Hidayat dan Mukhlas (2015) pemrograman linier memiliki syarat, yaitu:

1. Pemrograman linier harus memiliki fungsi tujuan (objective function) berupa garis lurus dengan persamaan fungsi Z atau f(Z), *c* adalah cost coefficient.
2. Harus memiliki kendala (constraints), yang dinyatakan garis lurus, dimana *a* = koefisien input output dan *b* = jumlah sumber daya yang tersedia.
3. Nilai x adalah positif atau sama dengan nol. Tidak boleh ada nilai x bernilai negatif.

Pemrograman linier dapat digunakan untuk menentukan campuran makanan ternak dengan efisien. Pemrograman linier mampu menentukan kombinasi terbaik antar pakan yang tersedia. Tiap pakan memiliki kandungan nutrisi dan harga yang berbeda sehingga pemrograman linier harus mampu memformulasikan ransum hingga mendapatkan ransum dengan harga paling minimum. Hasil dari formulasi tergantung pada nilai yang digunakan untuk kandungan nutrisi dan kebutuhan lainnya yang diperlukan dalam ransum. Harga juga dipengaruhi oleh komposisi nutrisi dari bahan pakan yang dipilih dan unit harga dari tiap bahan pakan yang digunakan. Meminimumkan harga pakan akan menjadi fungsi tujuan dari pemodelan ini, dengan kendala-kendala kandungan nutrisi dari setiap bahan pakan dengan kebutuhan nutrisi jenis ruminansia yang diinputkan (Hidayat dan Mukhlas 2015).

### Pembuatan *Protoype*

Membangun prototype dengan mengimplementasikan hasil perancangan pada tahap sebelumnya. *Prototype* dibuat dalam bentuk gambaran antarmuka sistem serta input yang dibutuhkan dan output yang akan dihasilkan. *Prototype*  harus mampu menggambarkan sistem yang akan dikembangkan. Komponen yang digunakan dalam pembuatan *prototype* harus berdasarkan hasil perancangan dari tahap perancangan dan pemodelan.

### *Deployment Delivery* dan *Feedback*

Prototype yang sudah disepakati kemudian dirancang dan dikembangkan menjadi sebuah sistem. Sistem yang telah dikembangkan dilakukan evaluasi oleh pengguna yang memahami alur proses formulasi ransum. Evaluasi bertujuan untuk memastikan alur proses pada sistem yang telah dikembangkan sesuai dengan kebutuhan pengguna dan tidak ada tahapan atau hasil penghitungan yang keliru. Jika hasil evaluasi sudah sesuai dengan kebutuhan pengguna maka pengembangan selesai dilakukan, jika evaluasi belum sesuai kebutuhan maka prototype diperbaiki dengan melakukan iterasi selanjutnya.

Kelayakan sistem juga diuji menggunakan MAPE dan MSE. Pengujian ini bertujuan untuk mengetahui hasil dari optimasi *linier progrmamming*. Menurut Suryaningrum dan Wijaya (2015) MAPE dihitung dengan menggunakan kesalahan absolut pada tiap prediksi dibagi dengan nilai aktual hasil formulasi. MAPE merupakan pengukuran kesalahan yang menghitung ukuran presentase penyimpangan antara data aktual dengan data prediksi. MSE menurut Suryaningrum dan Wijaya (2015) adalah metode lain untuk mengevaluasi nilai prediksi dengan aktual dengan mengkuadratkan masing-masing selisih. Pendekatan ini mengatur kesalahan prediksi yang besar karna tiap kesalah dikuadratkan. MSE merupakan rata-rata selisih kuadrat antara nilai prediksi dengan nilai aktual. Nilai MAPE dan MSE dapat dihitung dengan persamaan berikut:

**RUMUS**

Jika hasil evaluasi MAPE dan MSE memiliki nilai error yang rendah atau akurasi yang tinggi maka system siap dikirimkan. Namun jika hasil evaluasi memiliki nilai error yang cukup tinggi dan akurasi yang rendah maka iterasi dalam pemodelan perlu dievaluasi dan diulang kembali.

# HASIL DAN PEMBAHASAN

Metode *prototyping* memungkinkan pengembangan memiliki iterasi lebih dari satu kali. Pada pengembangan sistem formulasi ransum ini memiliki 2 iterasi. Iterasi pertama berhasil mengembangkan sistem formulasi yang sesuai dengan permintaan pakar. Iterasi kedua adalah pengembangan fungsi-fungsi untuk menunjang formulasi.

## Iterasi 1

**Komunikasi**

Komunikasi berguna sebagai sarana penggalian informasi. Komunikasi dilakukan antara pengembang sistem dengan pakar yang akan bersinggungan langsung dengan sistem. Pada tahap ini pakar menginformasikan cara melakukan formulasi ransum secara manual. Informasi yang dijabarkan oleh pakar mencakup tahapan dalam formulasi, data yang dibutuhkan dan hasil yang didapatkan. Hasil dari komunikasi adalah daftar kebutuhan pengguna yang dapat dilihat pada Tabel X.

**Perencanaan Cepat**

Perencanaan cepat sistem formulasi ransum berdasarkan daftar kebutuhan pengguna yang telah diperoleh. Pada perencanaan cepat memiliki 2 aktor yang akan menggunakan sistem yaitu admin sebagai pengelola data master dan pengunjung yang akan melakukan formulasi pada sistem. Aktivitas yang dapat dilakukan oleh masing-masing aktor dapat dilihat melalui diagram *use case* pada Gambar X. Perencanaan cepat juga menghasilkan diagram data model sebagai acuan alur data dan keterhubungan antar data. Diagram data model dapat dilihat pada Gambar X.

**Pemodelan Cepat**

Pemodelan cepat pada tahap ini mencakup pemodelan *linier programming*  dengan metode simpleks pada penghitungan formulasi.

Komunikasi pada iterasi pertama menghasilkan daftar kebutuhan sistem dan tahapan dalam melakukan formulasi. Daftar kebutuhan sistem pada tahap ini dapat dilihat melalui *use case* pada Gambar X. Tahapan dalam melakukan formulasi dapat dilihat pada Gambar X.

## Perancangan dan Pemodelan

Hasil dari tahapan komunikasi dikembangkan menjadi perancangan kebutuhan fungsional sistem dan basisdata. Kebutuhan fungsional dapat dilihat pada Tabel X. Perancangan basisdata dapat dilihat pada Gambar X. Pada tahap ini pemodelan *linier programming* dijabarkan menjadi sebuah alur data yang dapat dilihat pada Gambar X.

**Pembuatan *Prototype***

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 diinginan oleh *user.* *User stories* yang telah disusun pada tahap perencanaan dapat digambarkan dalam bentuk diagram *user case* pada Gambar 2.

1 Use case .png

Gambar 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 Kebutuhan fungsional *user* SiMURAA

| Kode fungsi | Kebutuhan fungsional | Keterangan |
| --- | --- | --- |
| SM-001 | Mendaftar akun | *User* melakukan pendaftaran dengan memberikan data yang telah ditentukan. Setelah mendaftar, *user* dapat masuk ke dalam sistem. |
| SM-002 | Mengubah profil *user* | *User* dapat mengubah informasi diri yang telah diberikan ketika pendaftaran akun. |
| SM-003 | Melihat informasi kebutuhan nutrien unggas | *User* dapat melihat informasi seluruh unggas dan kebutuhan nutriennya, serta menambah unggas dan kebutuhan nutriennya. |
| SM-004 | Melihat informasi bahan pakan | *User* dapat melihat informasi seluruh bahan pakan beserta harga dan kandungan nutriennya. |
| SM-005 | Membuat perhitungan formulasi ransum | *User* 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 *default* telah tersedia pada sistem. Namun, *user* 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 (*xi*) dan harga minimum dari seluruh bahan yang digunakan untuk pembuatan ransum (*S*). *Input* yang diperlukan tersebut yaitu nilai minimum dan maksimum kebutuhan nutrien suatu unggas (*bi*), harga tiap bahan pakan yang digunakan dalam pembuatan ransum (*ci*), nilai minumum dan maksimum penggunaan bahan pakan dalam ransum (*bi*), dan nutrien-nutrien yang terkandung dalam tiap bahan pakan yang digunakan dalam pembuatan ransum (*aij*).

*User* dapat memulai perhitungan formulasi ransum dengan memilih nama unggas yang telah tersedia dalam *database.* Data kebutuhan nutrien 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 nutrien 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 *xi* dan *S.* Alur proses perhitungan menggunakan *linear programming* dapat dilihat pada Gambar 3.

flowchart.png

Gambar 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.

CRC Card.png

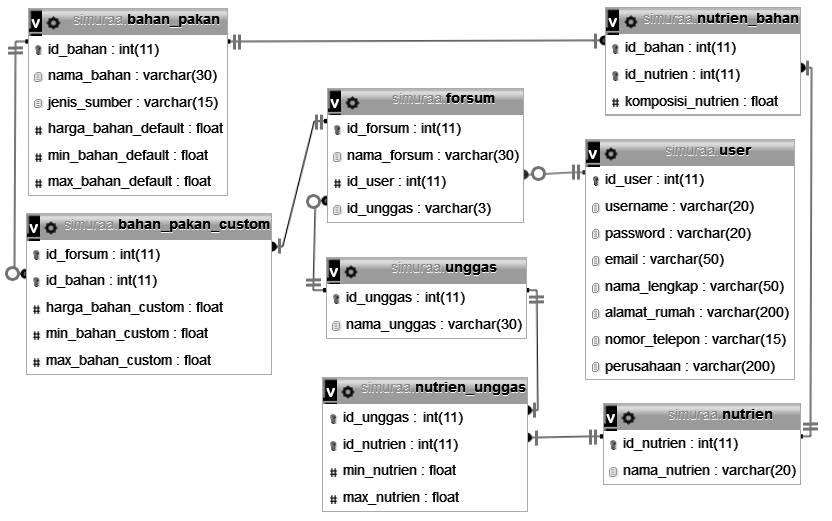
Gambar CRC *Card* SiMURAA

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 konseptual dapat dilihat pada Lampiran 1 dan diagram kelas spesifikasi pada Lampiran 2. Diagram kelas dalam bentuk implementasi dapat dilihat pada Gambar 5.

3c class diagram tahap implementasi.png

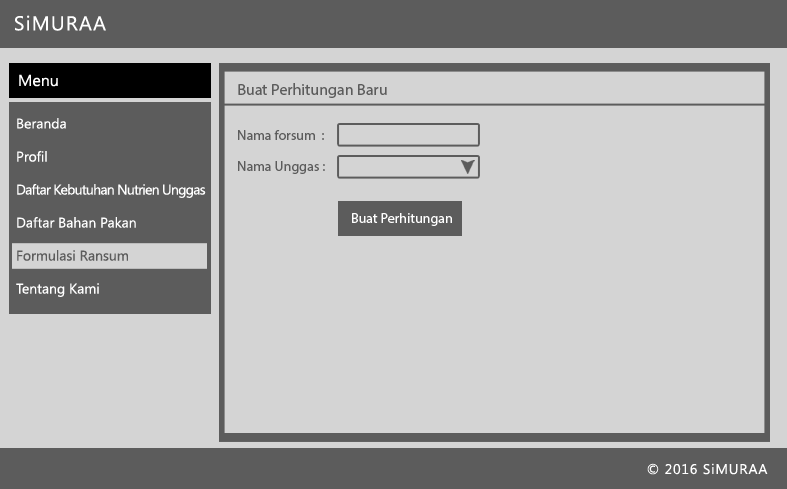
Gambar Diagram kelas SiMURAA

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 delapan kelas yang diubah menjadi delapan entitas yaitu kelas BahanPakan menjadi entitas bahan\_pakan, kelas BahanPakanCustom menjadi bahan\_pakan\_custom, kelas Unggas menjadi entitas unggas, kelas NutrienUnggas menjadi entitas nutrien\_unggas, kelas Nutrien menjadi entitas nutrien, kelas NutrienBahan menjadi entitas nutrien\_bahan, kelas Forsum menjadi entitas forsum, dan kelas User menjadi entitas user. ERD SiMURAA dapat dilihat pada Gambar 6. Pendefinisian atribut-atribut tiap entitas di ERD dapat dilihat pada kamus data di Lampiran 3.



Gambar ERD SiMURAA

Desain prototipe dibutuhkan sebagai gambaran antarmuka pengguna yang dapat dilihat langsung dan disesuaikan dengan keinginan pengguna. Prototipe dibuat untuk keseluruhan fungsi yang ada pada SiMURAA dan dapat dilihat pada Lampiran 4. Contoh prototipe antarmuka halaman memulai perhitungan formulasi ransum dapat dilihat pada Gambar 7.



Gambar Prototipe halaman memulai perhitungan formulasi ransum

Untuk melengkapi dokumentasi sistem, diagram aktivitas dan diagram sekuens perlu dibuat. Diagram aktivitas dibuat untuk menggambarkan alur kerja atau proses bisnis SiMURAA. Jumlah diagram aktivitas yang dibuat disesuaikan dengan jumlah *use case* yang telah didefinisikan pada awal tahap desain. Seluruh diagram aktivitas SiMURAA dapat dilihat pada Lampiran 5. Gambar 8 berikut merupakan diagram aktivitas membuat perhitungan formulasi ransum.

## 2e activity diagram - membuat perhitungan formulasi ransum.png

Gambar Diagram aktivitas membuat perhitungan formulasi ransum

Diagram sekuens dibuat untuk menggambarkan interaksi tiap-tiap objek dengan susunan sesuai urutan eksekusinya. Jumlah diagram sekuens yang dibuat disesuaikan dengan jumlah *use case* yang telah didefinisikan pada awal tahap desain. Seluruh diagram sekuens SiMURAA dapat dilihat pada Lampiran 6. Gambar 9 berikut merupakan diagram sekuens membuat perhitungan formulasi ransum.

## 3e.png

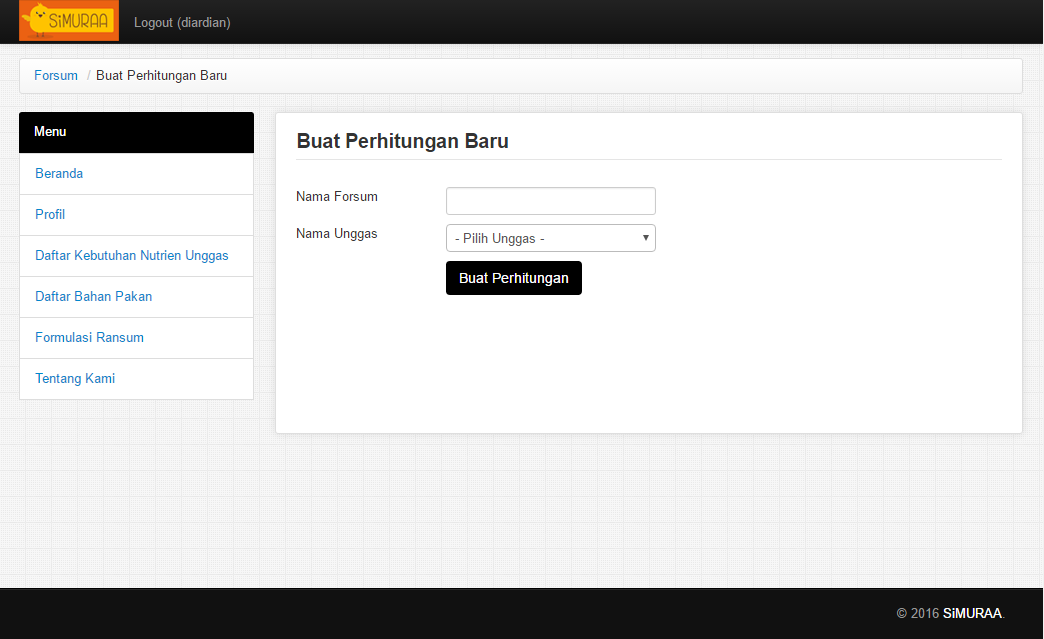
Gambar Diagram sekuens membuat perhitungan formulasi ransum

## 

## *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.

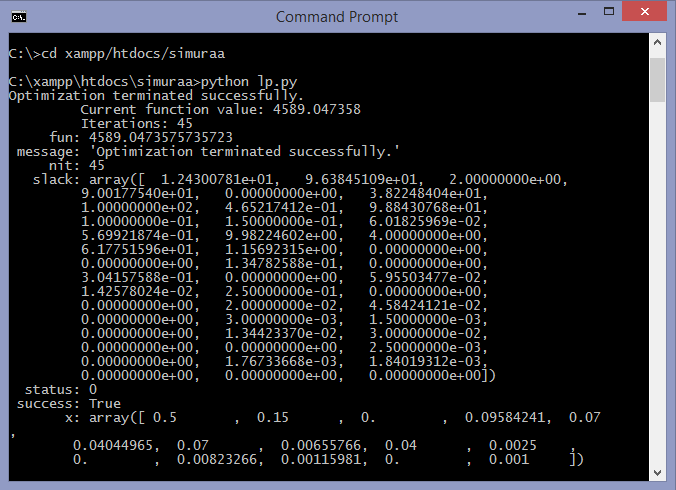
Pengembangan SiMURAA yang berbasis web menggunakan bahasa PHP dengan *framework* Yii. Terdapat 6 menu pada SiMURAA yaitu Beranda, Profil, Daftar Kebutuhan Nutrien Unggas, Daftar Bahan Pakan, Formulasi Ransum, dan Tentang Kami. Pada menu Profil, *user* dapat melihat profil diri dan mengubahnya. Pada menu Daftar Kebutuhan Nutrien Unggas, *user* dapat melihat informasi mengenai jenis-jenis unggas beserta nutrien yang dibutuhkan unggas tersebut. Pada menu Daftar Bahan Pakan, *user* dapat melihat daftar bahan pakan beserta nutrien yang dikandungnya. Pada menu Formulasi Ransum, *user* dapat melakukan perhitungan formulasi ransum yang menjadi fungsi utama pada SiMURAA. Seluruh halaman yang terdapat pada SiMURAA dapat dilihat pada Lampiran 7 dan contoh salah satu halaman untuk memulai formulasi ransum dapat dilihat pada Gambar 10.



Gambar Halaman untuk memulai formulasi ransum

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 (≤) 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 (≥) ruas kanan. Tanda pertidaksamaan lebih dari atau sama dengan (≥) harus diubah menjadi kurang dari atau sama dengan (≤) 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 , , dan 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 11.



Gambar 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 Hasil pengujian menggunakan *black-box testing*

| Kasus uji | Nilai input | Skenario uji | Hasil yang diharapkan | Hasil uji |
| --- | --- | --- | --- | --- |
| Mendaftar akun | Benar | *User* mendaftar akun sesuai dengan formulir yang disediakan pada halaman daftar akun | *User* berhasil mendaftar dan data *user* masuk ke *database* | Berhasil |
|  | Salah | *User* mendaftar akun namun tidak mengisi formulir pada halaman daftar akun | *User* tidak dapat mendaftar dan muncul pesan untuk tidak mengosongkan formulir | Berhasil |
| Mengubah profil *user* | Benar | *User* mengubah beberapa informasi diri pada halaman ubah profil | *User* berhasil mengubah informasi dirinya dan *database user* tersebut berubah | Berhasil |
| Melihat informasi kebutuhan nutrien unggas | Benar | *User* melakukan klik pada menu daftar kebutuhan nutrien unggas | *User* berhasil melihat informasi unggas beserta kebutuhan nutriennya | Berhasil |
| Melihat informasi bahan pakan | Benar | *User* melakukan klik pada menu daftar bahan pakan | *User* berhasil melihat informasi bahan pakan beserta kandungan nutriennya | Berhasil |
| Membuat perhitungan formulasi ransum | Benar | *User* melakukan perhitungan baru dengan memilih jenis unggas dan bahan pakan yang akan digunakan pada halaman buat perhitungan baru | *User* berhasil mendapatkan hasil berupa harga termurah dari pembuatan ransum dan nilai persentase penggunaan bahan pakan yang telah dipilih | Berhasil |
| Mengubah informasi bahan pakan | Benar | User mengubah *default* nilai minimal dan maksimal dengan nilai 0-1 pada halaman kriteria bahan pakan | *User* berhasil melanjutkan perhitungan dengan mendapatkan hasil perhitungan formulasi ransum | Berhasil |
|  | Salah | User mengubah *default* nilai maksimal dengan nilai 5 pada halaman kriteria bahan pakan | *User* 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 pembandingan hasil perhitungan menggunakan *linear programming* yang digunakan SiMURAA dan WinFeed 2.8. Pengujian perbandingan dilakukan 5 kali dengan penggunaan bahan pakan dan pembatasan kebutuhan nutrien unggas yang berbeda. Pembatas bahan pakan dan kebutuhan nutrien berupa nilai minimal (min) dan nilai maksimal (max).

Pengujian 1 menggunakan 10 bahan pakan dan 4 pembatasan kebutuhan nutrien, pengujian 2 menggunakan 10 bahan pakan dan 6 pembatasan kebutuhan nutrien, pengujian 3 menggunakan 12 bahan pakan dan 8 pembatasan kebutuhan nutrien, pengujian 4 menggunakan 12 bahan pakan dan 10 pembatasan kebutuhan nutrien, dan pengujian 5 menggunakan 12 bahan pakan dan 12 pembatasan kebutuhan nutrien. Bahan pakan yang digunakan beserta nilai pembatas penggunaannya pada 5 kali pengujian tersebut dapat dilihat pada Tabel 3. Kebutuhan nutrien yang dibatasi pada 5 kali pengujian dapat dilihat pada Tabel 4.

Setelah penentuan bahan pakan yang digunakan beserta batasannya, dan pembatasan kebutuhan nutrien 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 Bahan pakan dan batasan penggunaannya pada pengujian

|  | Pengujian 1 | | Pengujian 2 | | Pengujian 3 | | Pengujian 4 | | Pengujian 5 | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bahan Pakan | 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 |
| CaCO3 | 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 Batasan kebutuhan nutrien pada pengujian

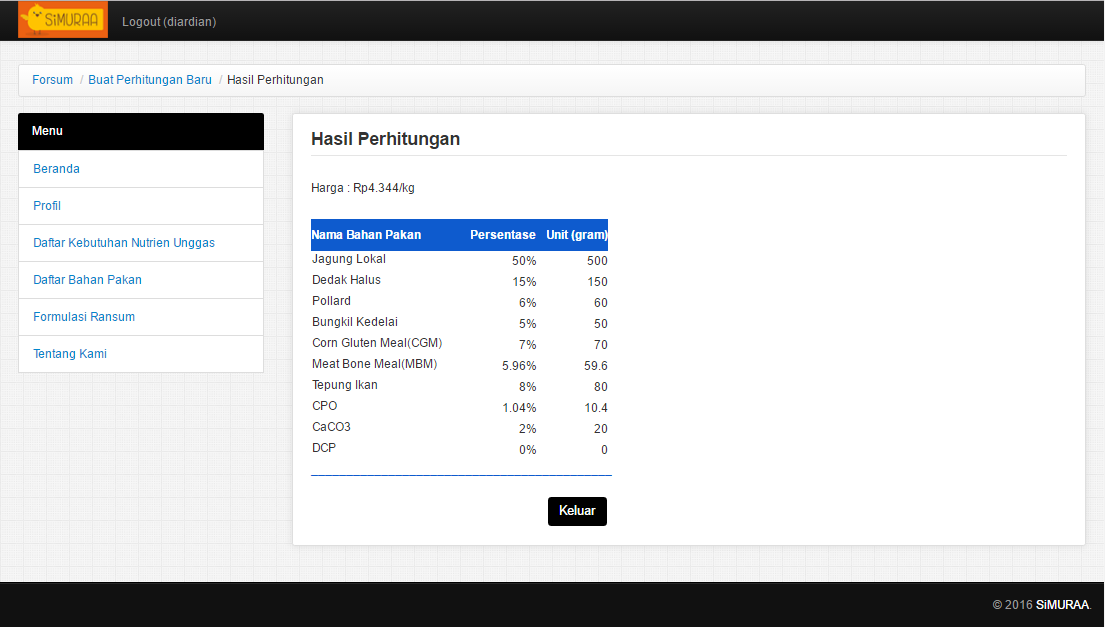
|  | Pengujian 1 | | Pengujian 2 | | Pengujian 3 | | Pengujian 4 | | Pengujian 5 | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Nutrien | Min | Max | Min | Max | Min | Max | Min | Max | Min | Max |
| Bahan kering (%) | 86 | 100 | 86 | 100 | 86 | 100 | 86 | 100 | 86 | 100 |
| 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 Harga ransum dan komposisi penggunaan bahan pakan pada pengujian

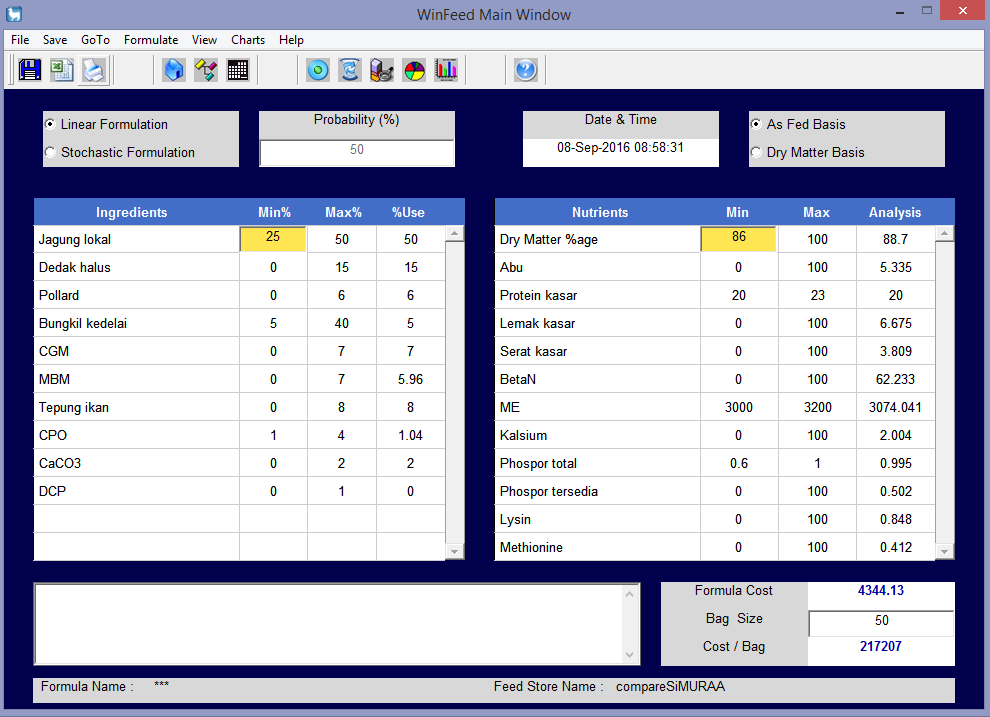
|  | Pengujian 1 | | Pengujian 2 | | Pengujian 3 | | Pengujian 4 | | Pengujian 5 | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Hasil | SM | WF | SM | WF | SM | WF | SM | WF | SM | WF |
| Harga (Rp) | 4 344 | 4 344 | 4 379 | 4 379 | 4 477 | 4 477 | - | 5 223 | 5 293 | 5 293 |
| 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 |
| CaCO3 (%) | 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 nutrien jika kurang atau lebih dari batasan yang telah ditentukan.

Salah satu contoh kasus perbandingan yang berhasil dalam pengujian ini yaitu perhitungan formulasi ransum pada pengujian 1 dengan memilih jenis unggas dengan penentuan 4 kebutuhan nutrien yang dibatasi dan 10 bahan pakan yang digunakan. Hasil harga termurah yang diperoleh menggunakan SiMURAA yaitu Rp4 344/kg dapat dilihat pada Gambar 12, dan hasil harga termurah dengan menggunakan WinFeed 2.8 yaitu Rp4 344/kg dapat dilihat pada Gambar 13.



Gambar Hasil perhitungan formulasi ransum menggunakan SiMURAA



Gambar Hasil perhitungan formulasi ransum menggunakan Winfeed 2.8

# SIMPULAN DAN SARAN

## 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 nutrien 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 berkirim 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.

Lampiran 1 Diagram kelas konseptual

3a class diagram tahap konseptual.png

Lampiran 2 Diagram kelas spesifikasi

3b class diagram tahap spesifikasi.png

Lampiran Kamus data SiMURAA

1. Tabel bahan\_pakan (kelas BahanPakan)

| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_bahan | int (11) | *Primary key*, identitas unik dari entitas bahan pakan |
| 2 | nama\_bahan | varchar (30) | Nama bahan pakan |
| 3 | jenis\_sumber | varchar (15) | Jenis sumber bahan pakan, dapat berupa sumber energi, protein, mineral, asam amino, atau premix |
| 4 | harga\_bahan\_default | Float | Harga bahan pakan yang menjadi nilai *default* dalam perhitungan formulasi ransum |
| 5 | min\_bahan\_default | Float | Nilai minimal penggunaan bahan pakan yang menjadi nilai *default* dalam perhitungan formulasi ransum |
| 6 | max\_bahan\_default | Float | Nilai maksimal penggunaan bahan pakan yang menjadi nilai *default* dalam perhitungan formulasi ransum |

1. Tabel bahan\_pakan\_custom (kelas BahanPakanCustom)

| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_forsum | int (11) | *Composite key*, identitas unik dari entitas forsum |
| 2 | id\_bahan | int (11) | *Composite key*, identitas unik dari entitas bahan pakan |
| 3 | harga\_bahan\_custom | Float | Harga bahan pakan yang dapat diubah oleh *user* dalam perhitungan formulasi ransum |
| 4 | min\_bahan\_custom | Float | Nilai minimal penggunaan bahan pakan yang dapat diubah oleh *user* dalam perhitungan formulasi ransum |
| 5 | max\_bahan\_custom | Float | Nilai maksimal penggunaan bahan pakan yang dapat diubah oleh *user* dalam perhitungan formulasi ransum |

1. Tabel unggas (kelas Unggas)

| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_unggas | int (11) | *Primary key*, identitas unik dari entitas unggas |
| 2 | nama\_unggas | varchar (30) | Nama unggas |

1. Tabel nutrien\_unggas (kelas NutrienUnggas)

| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_unggas | int (11) | *Composite key*, identitas unik dari entitas unggas |
| 2 | id\_nutrien | int (11) | *Composite key*, identitas unik dari entitas nutrien |
| 3 | min\_nutrien | Float | Nilai minimal kebutuhan nutrien unggas yang digunakan dalam perhitungan formulasi ransum |
| 4 | max\_nutrien | Float | Nilai maksimal kebutuhan nutrien unggas yang digunakan dalam perhitungan formulasi ransum |

1. Tabel nutrien (kelas Nutrien)

| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_nutrien | int (11) | *Primary key*, identitas unik dari entitas nutrien |
| 2 | nama\_nutrien | varchar (20) | Nama nutrien |

1. Tabel nutrien\_bahan (kelas NutrienBahan)

| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_bahan | int (11) | *Composite key*, identitas unik dari entitas bahan pakan |
| 2 | id\_nutrien | int (11) | *Composite key*, identitas unik dari entitas nutrien |
| 3 | komposisi\_nutrien | Float | Kandungan nutrien dalam bahan pakan yang digunakan dalam perhitungan formulasi ransum |

1. Tabel forsum (kelas Forsum)

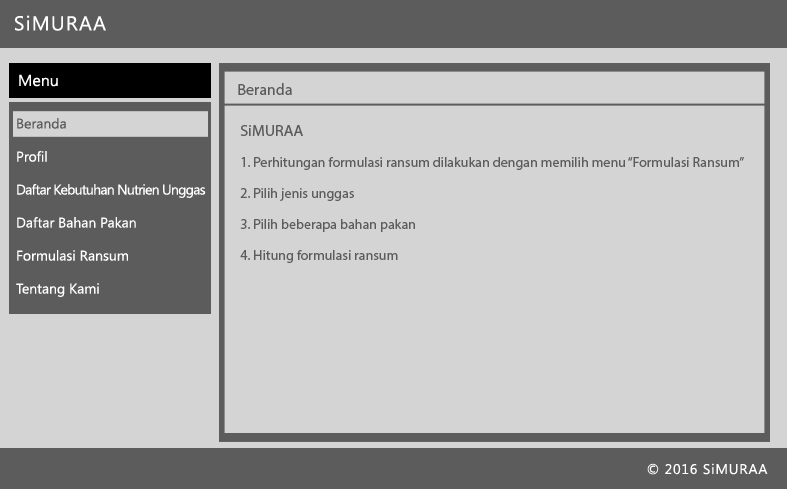
| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_forsum | int (11) | *Primary key*, identitas unik dari entitas forsum |
| 2 | nama\_forsum | varchar (30) | Nama forsum yang dibuat ketika *user* memulai perhitungan formulasi ransum |
| 3 | id\_user | int (11) | *Foreign key*, identitas unik dari entitas user |
| 4 | id\_unggas | varchar (3) | *Foreign key*, identitas unik dari entitas unggas |

1. Tabel user (kelas User)

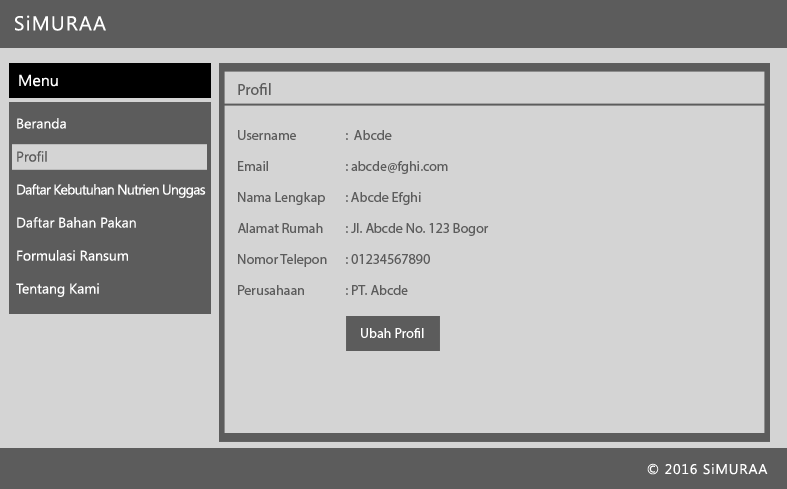
| No | Nama Atribut | Tipe data/lebar | Keterangan |
| --- | --- | --- | --- |
| 1 | id\_user | int (11) | *Primary key*, identitas unik dari entitas user |
| 2 | Username | varchar (20) | nama unik dari setiap *user* |
| 3 | Password | varchar (20) | Kata sandi yang digunakan untuk mendapat akses *login* |
| 4 | Email | varchar (50) | Alamat *e-mail* dari *user* |
| 5 | nama\_lengkap | varchar (50) | Nama lengkap *user* |
| 6 | alamat\_rumah | varchar (200) | Alamat rumah tempat tinggal *user* |
| 7 | nomor\_telepon | varchar (15) | Nomor telepon *user* yang dapat dihubungi |
| 8 | Perusahaan | varchar (200) | Nama perusahaan *user* |

Lampiran Prototipe seluruh fungsi pada SiMURAA

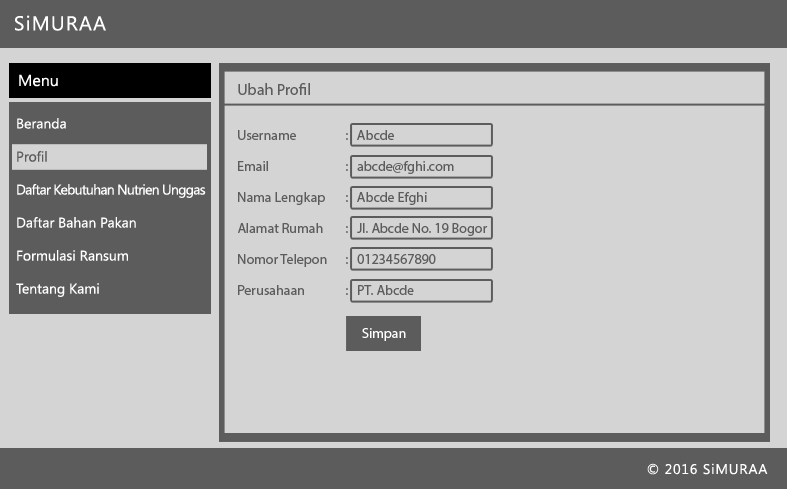
1. Prototipe halaman beranda



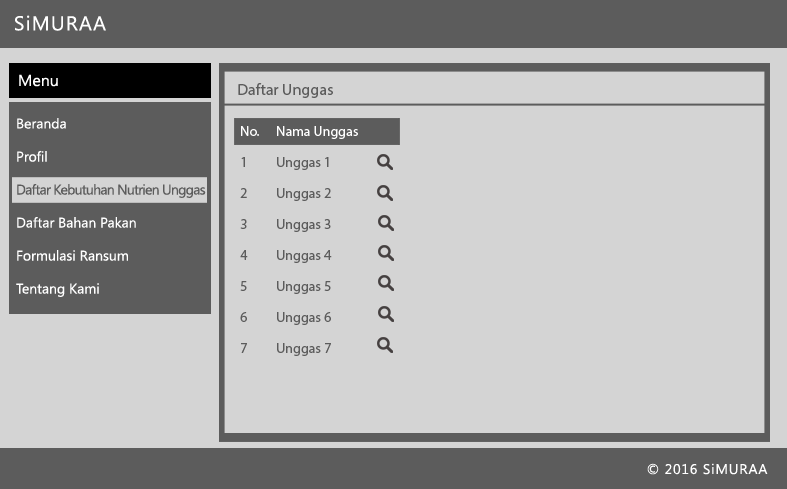
1. Prototipe halaman profil *user*



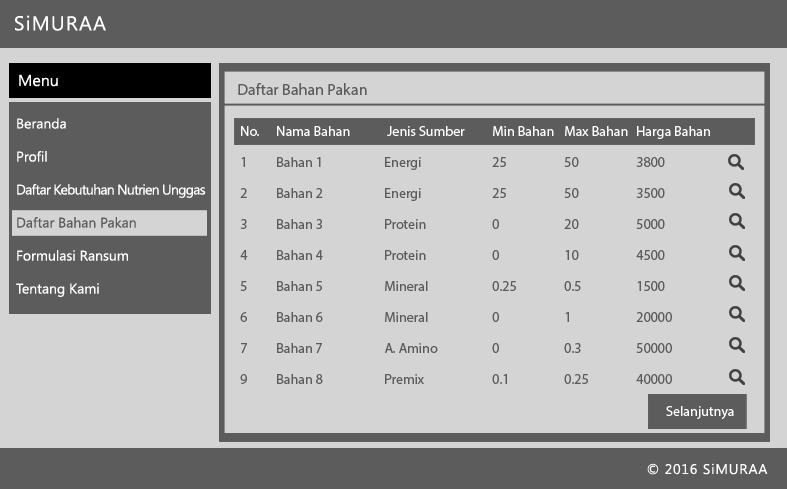
1. Prototipe halaman ubah profil *user*



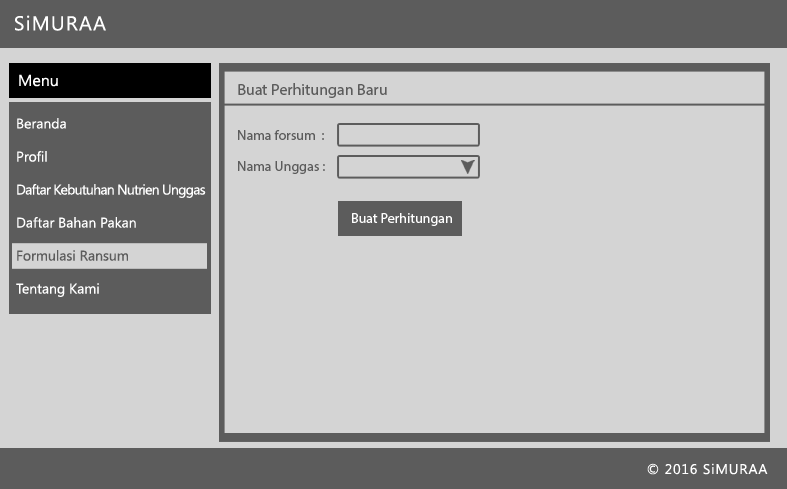
1. Prototipe halaman daftar unggas



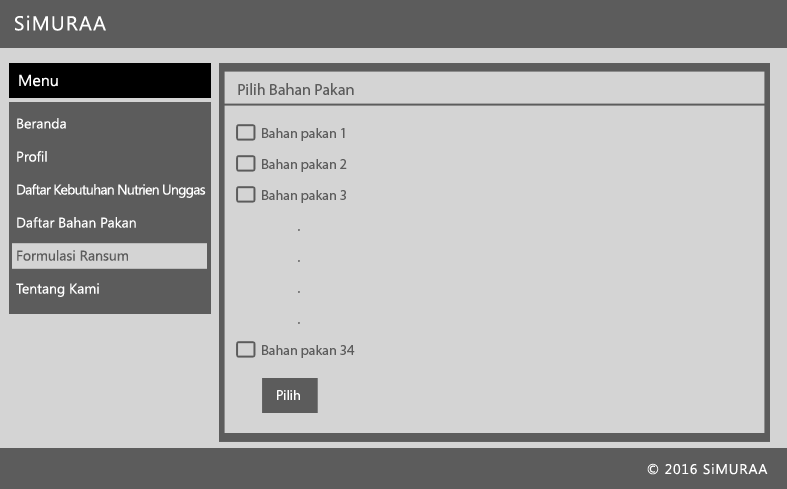
1. Prototipe halaman daftar bahan pakan



1. Prototipe halaman buat perhitungan baru



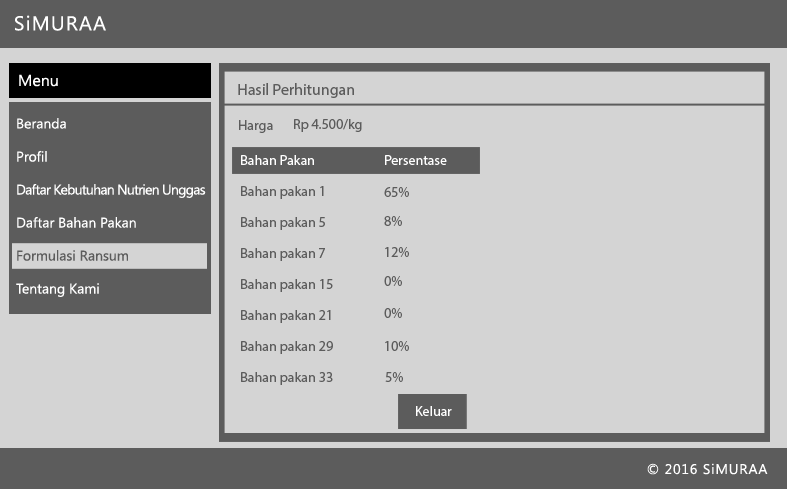
1. Prototipe halaman pilih bahan pakan



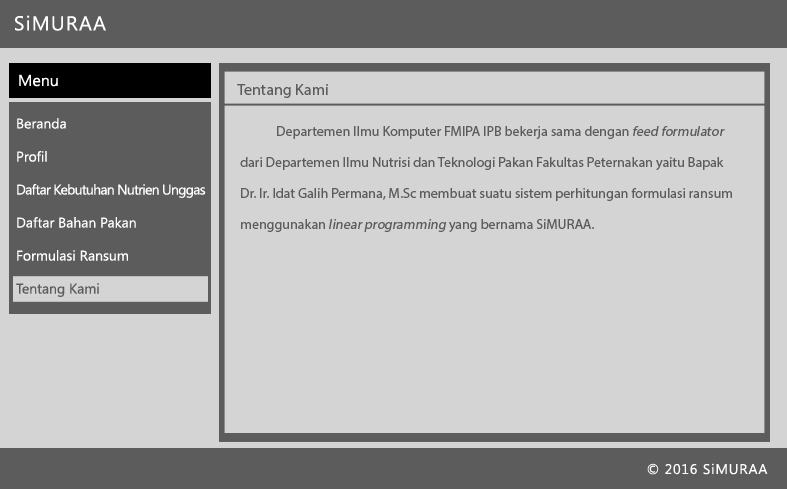
1. Prototipe halaman kriteria bahan pakan



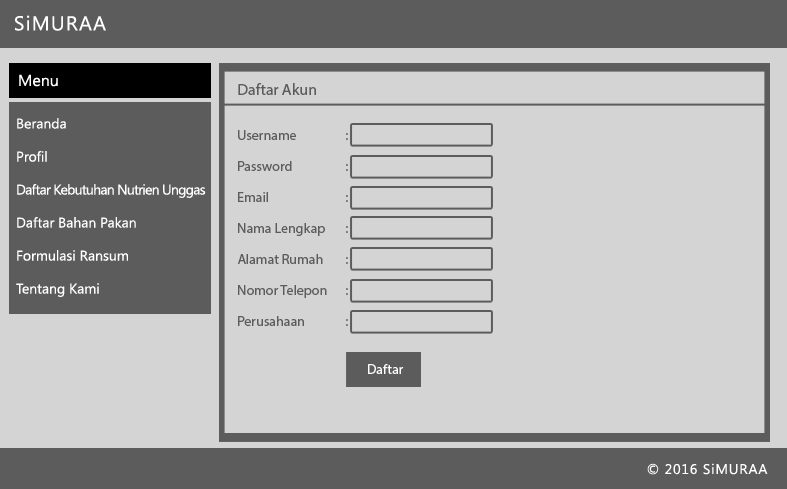
1. Prototipe halaman hasil perhitungan



1. Prototipe halaman tentang kami



1. Prototipe halaman daftar akun



Lampiran Seluruh diagram aktivitas pada SiMURAA

1. Diagram aktivitas mendaftar akun

2a activity diagram - mendaftar akun.png

1. Diagram aktivitas mengubah profil *user*

2b activity diagram - mengubah profil user.png

1. Diagram aktivitas melihat informasi kebutuhan nutrien unggas

2c activity diagram - melihat informasi kebutuhan nutrien unggas.png

1. Diagram aktivitas melihat informasi bahan pakan

2d activity diagram - melihat informasi bahan pakan.png

1. Diagram aktivitas membuat perhitungan formulasi ransum

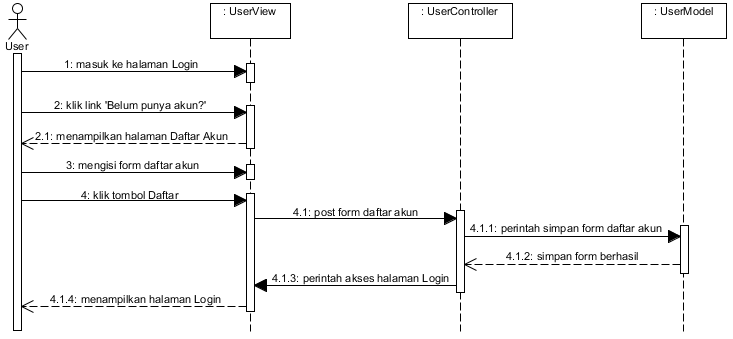
2e activity diagram - membuat perhitungan formulasi ransum.png

1. Diagram aktivitas mengubah informasi bahan pakan

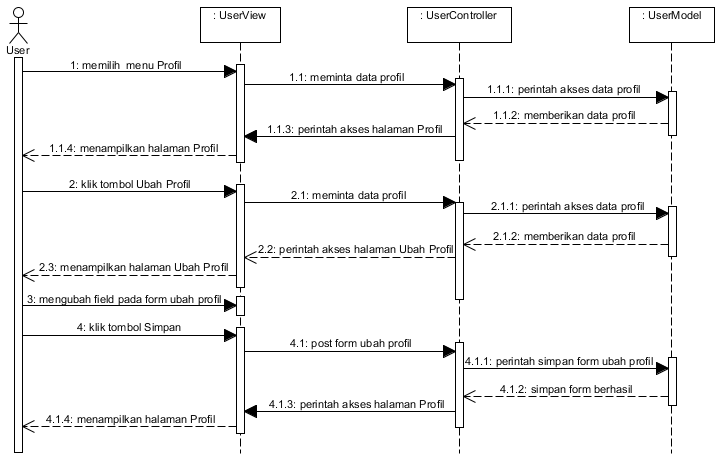
2f activity diagram - mengubah informasi bahan pakan.png

Lampiran Seluruh diagram sekuens pada SiMURAA

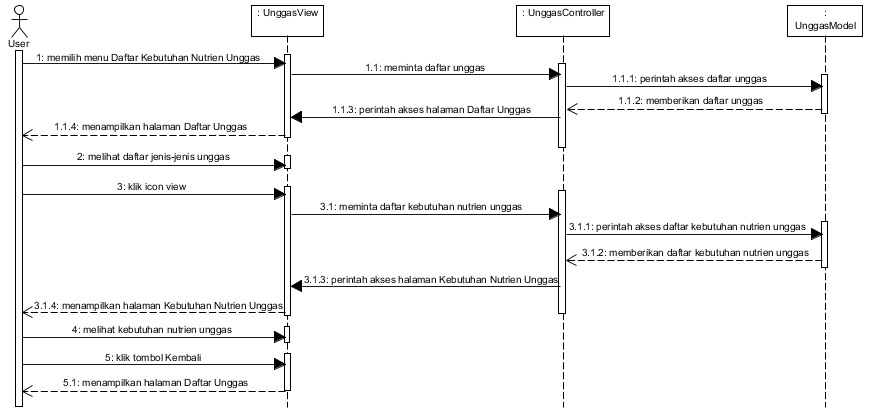
1. Diagram sekuens mendaftar akun



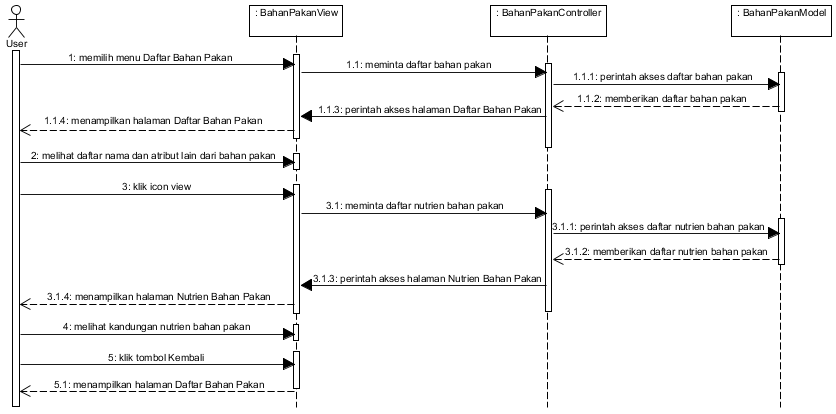
1. Diagram sekuens mengubah profil *user*



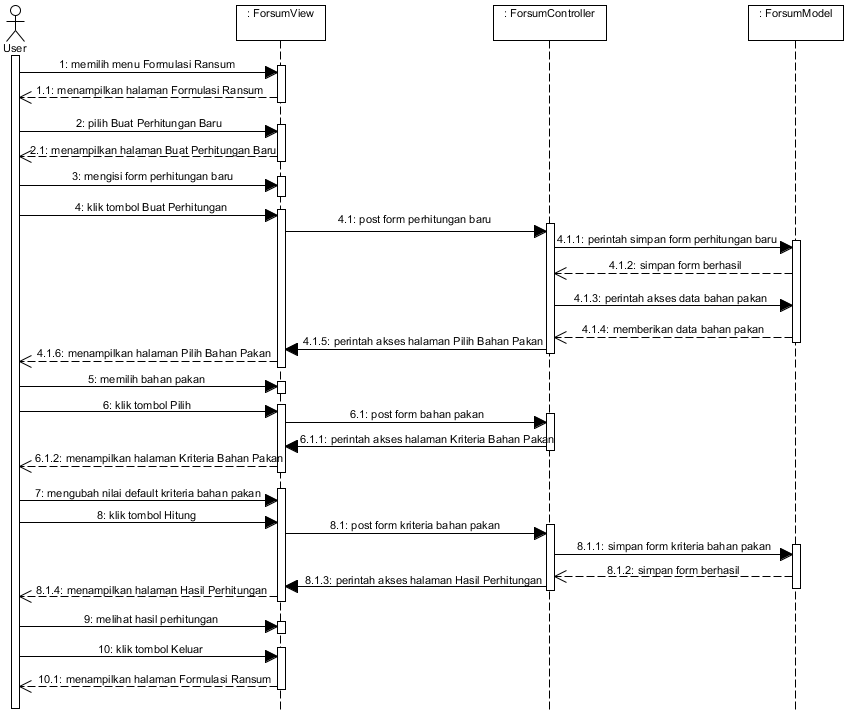
1. Diagram sekuens melihat informasi kebutuhan nutrien unggas



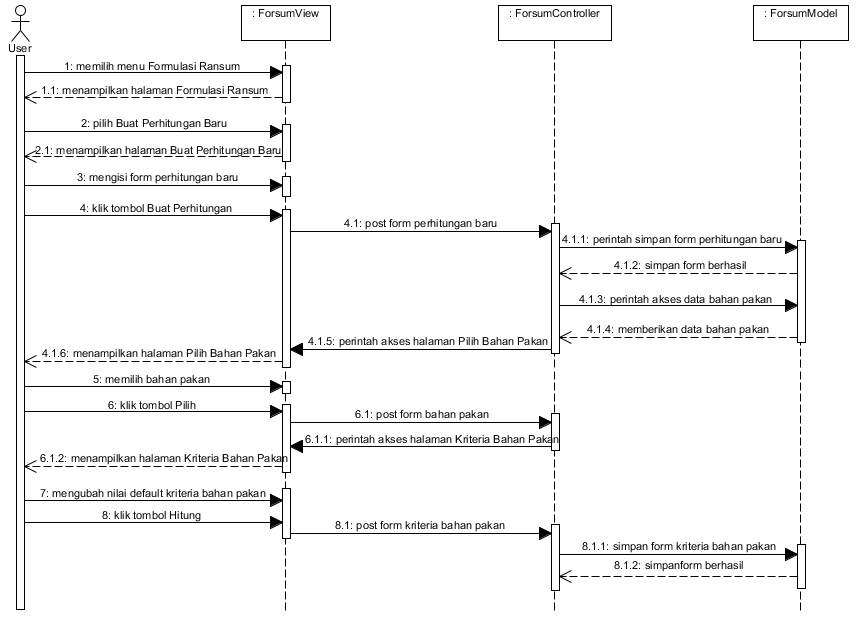
1. Diagram sekuens melihat informasi bahan pakan



1. Diagram sekuens membuat perhitungan formulasi ransum

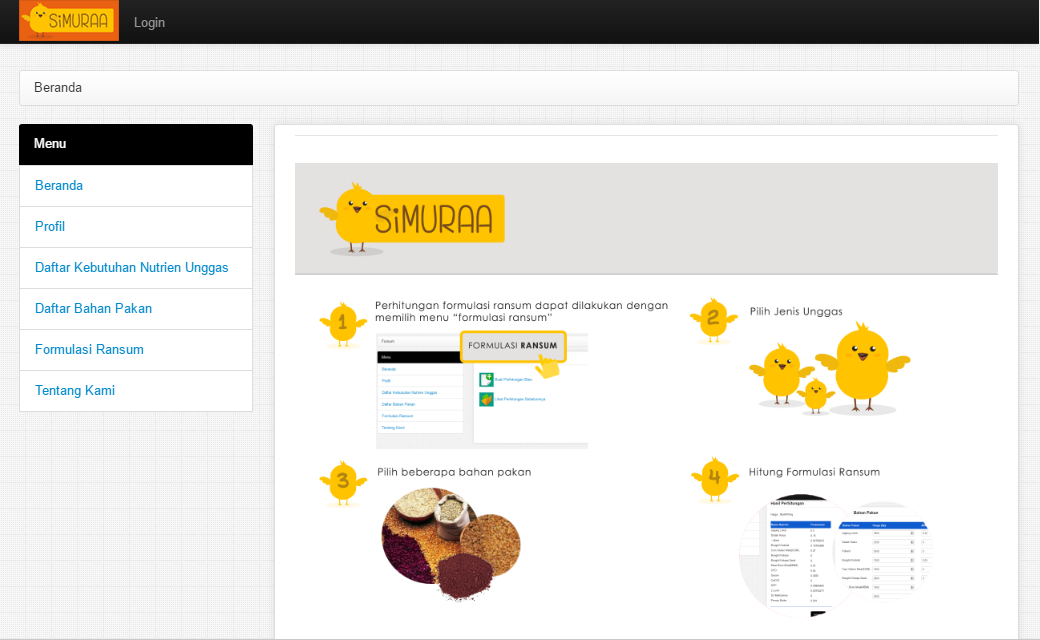


1. Diagram sekuens mengubah informasi bahan pakan

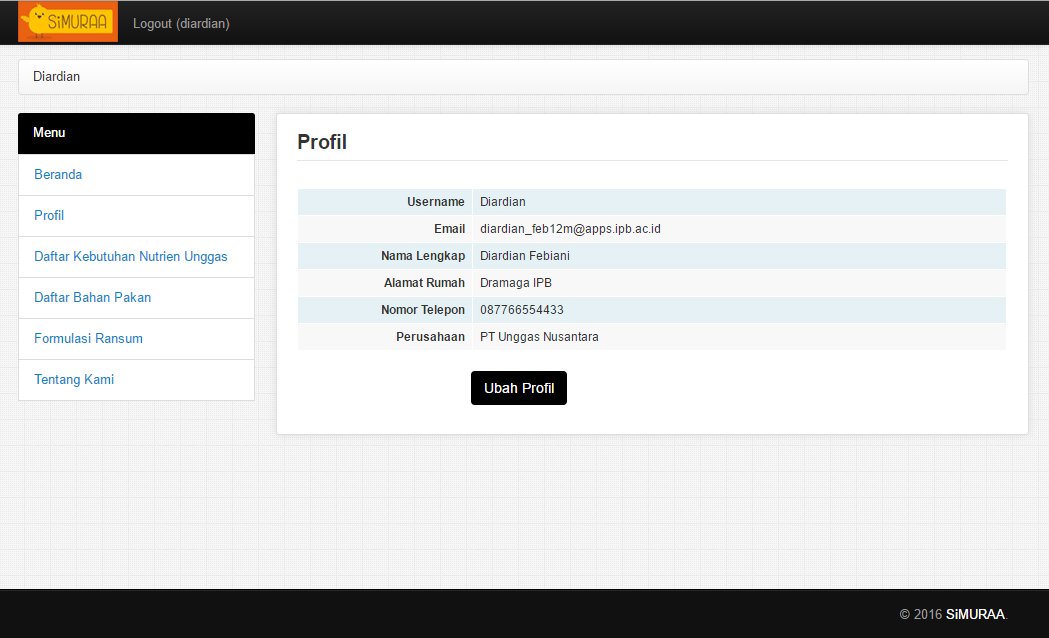


Lampiran Seluruh halaman pada SiMURAA

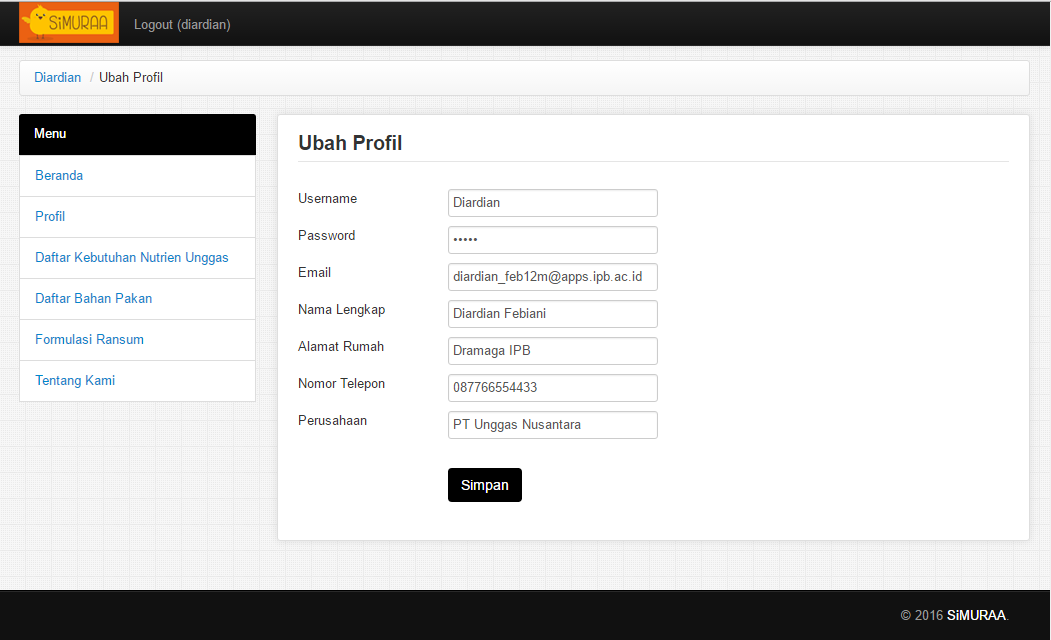
1. Halaman beranda



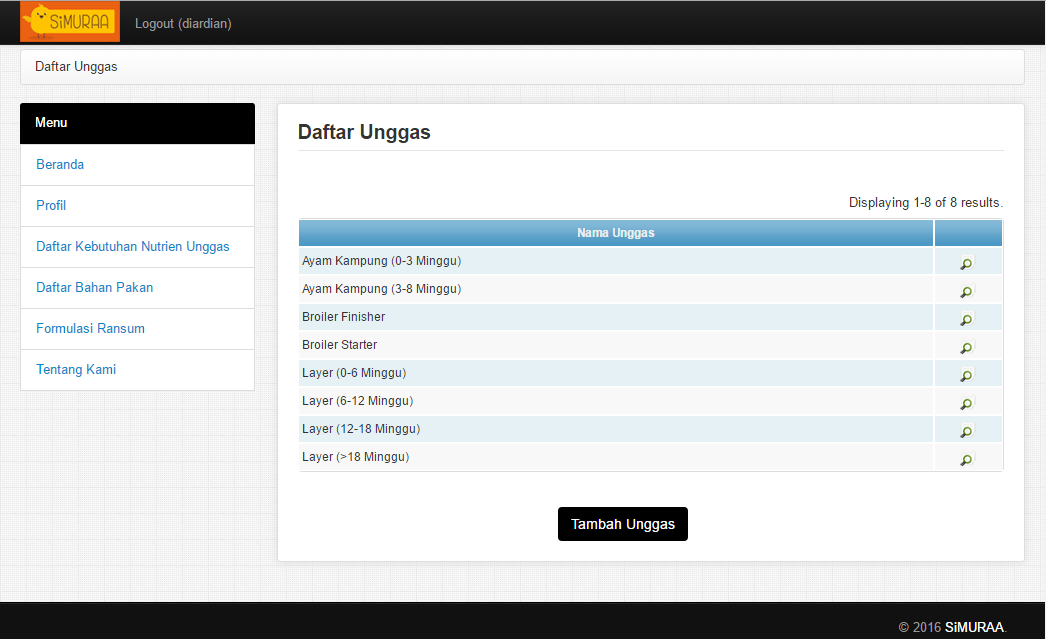
1. Halaman profil *user*



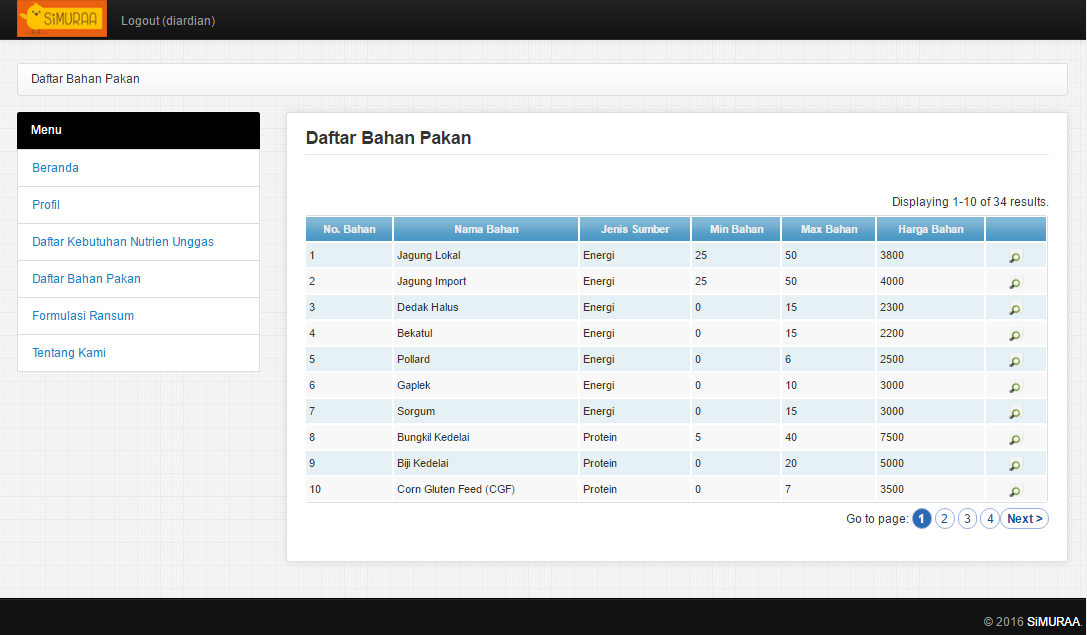
1. Halaman ubah profil *user*



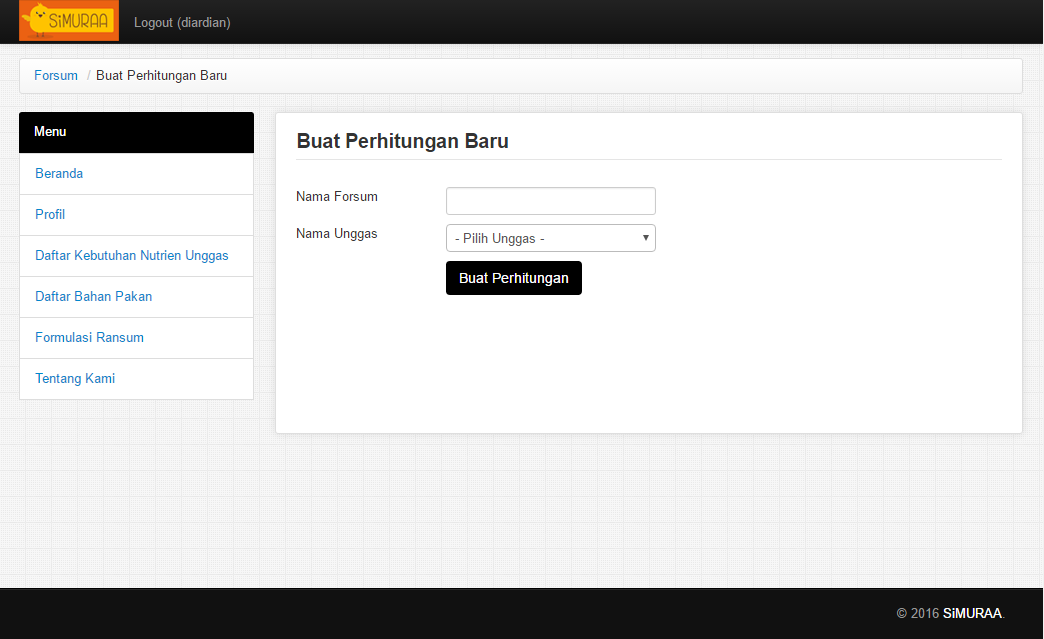
1. Halaman daftar unggas



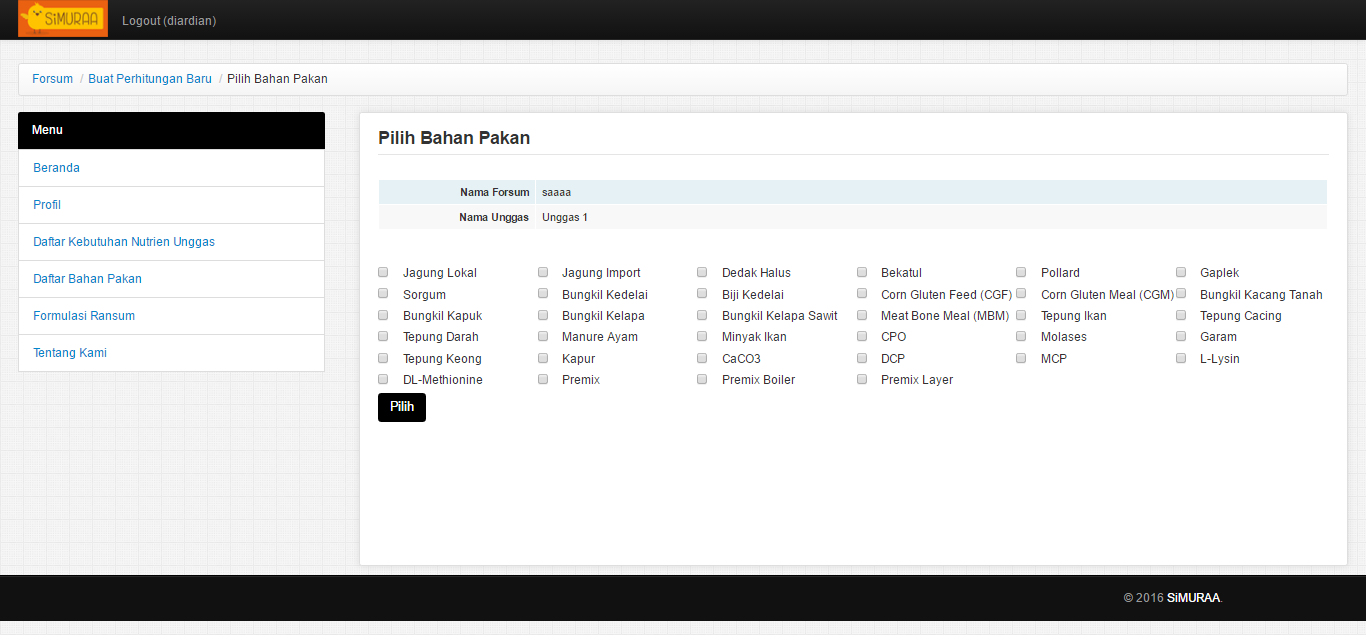
1. Halaman daftar bahan pakan



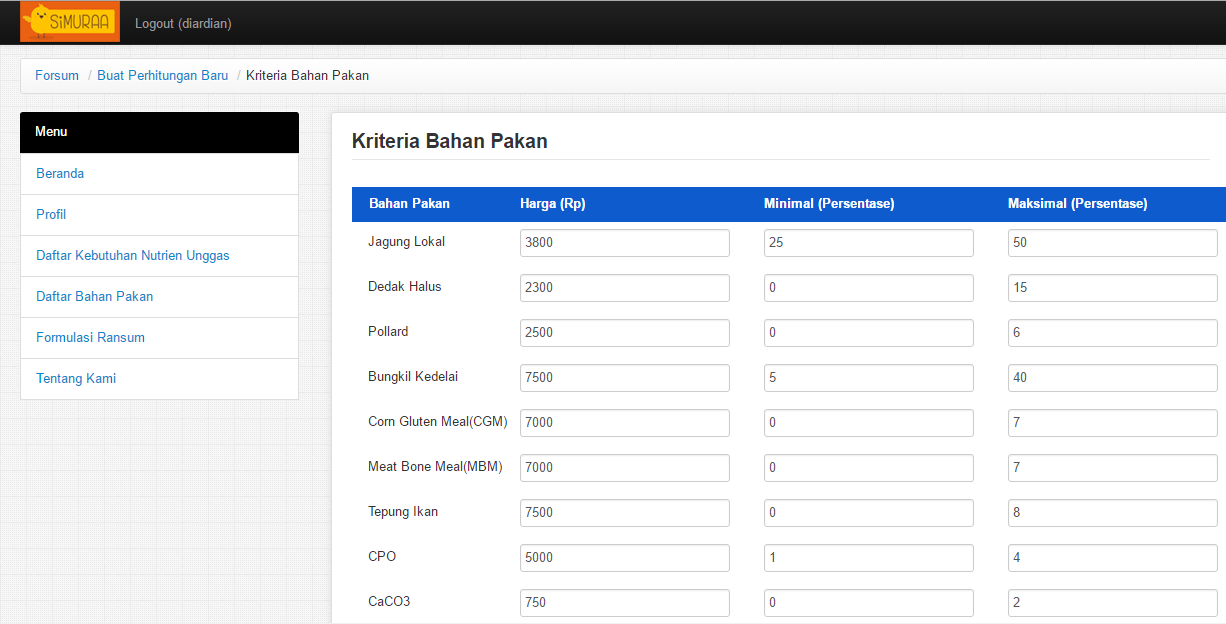
1. Halaman buat perhitungan baru



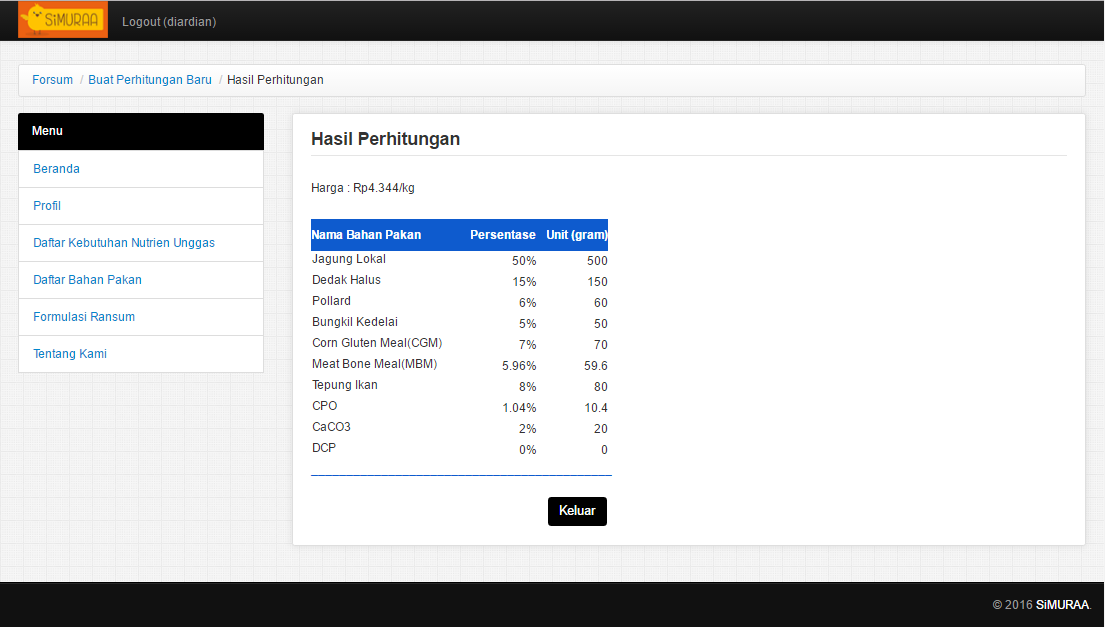
1. Halaman pilih bahan pakan



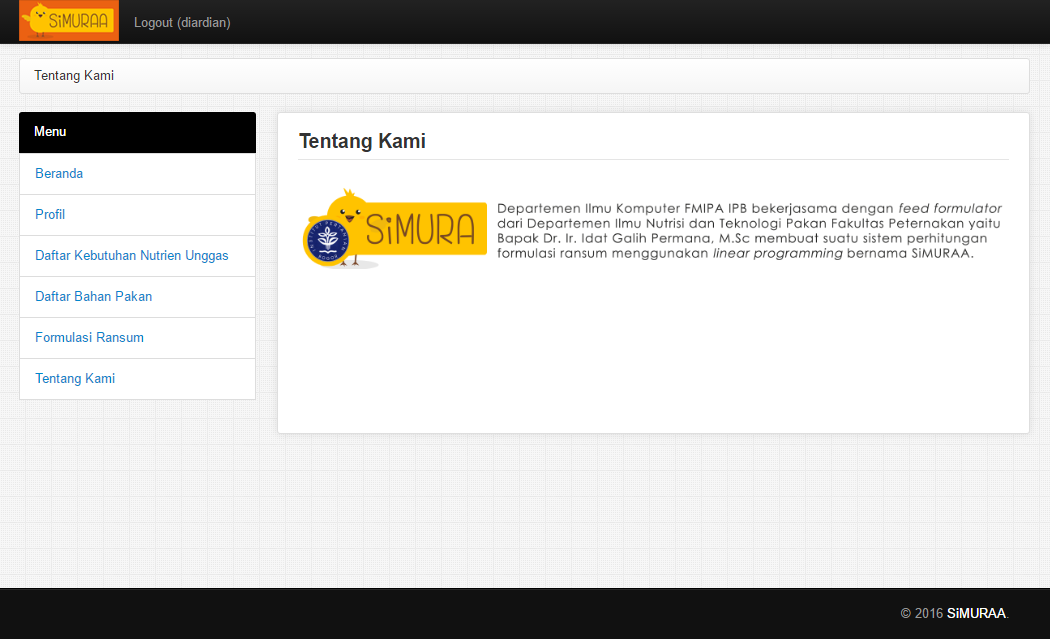
1. Halaman kriteria bahan pakan



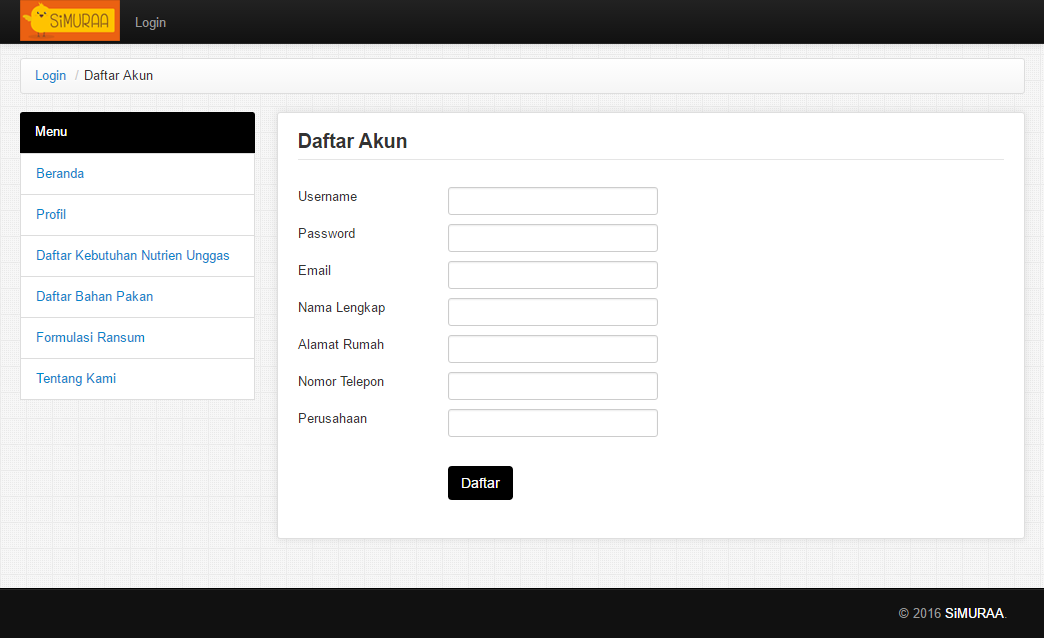
1. Halaman hasil perhitungan



1. Halaman tentang kami



1. Halaman daftar akun



**RIWAYAT HIDUP**

Penulis lahir di Bogor Provinsi Jawa Barat pada tanggal 19 Februari 1994. Penulis adalah anak kedua dari dua bersaudara, anak dari pasangan Tatag Budiardi dan Kurnia Suci Indraningsih. Penulis menempuh pendidikan sekolah menengah pertama di SMP Negeri 1 Bogor dari tahun 2006 hingga 2009 dan melanjutkan sekolah menengah atas di SMA Negeri 1 Bogor pada tahun 2009 hingga 2012. Kemudian, penulis melanjutkan pendidikan di Institut Pertanian Bogor, Fakultas Matematika dan Ilmu Pengetahuan Alam, Departemen Ilmu Komputer dari tahun 2012 hingga 2016.

Penulis pernah bergabung dengan organisasi kampus yaitu Himpunan Mahasiswa Ilmu Komputer selama 1 periode sebagai sekretaris Divisi Internal dan organisasi luar kampus yaitu Forum for Indonesia Chapter Bogor selama 2 periode sebagai sekretaris I dan staf Divisi *Business and Entrepreneurship*. Penulis juga pernah mengikuti berbagai perlombaan dan meraih penghargaan sebagai Juara 3 Kategori Fotografi Himasiera Olah Talenta 2012, Semifinalis Kategori Piranti Cerdas dan *Embedded System* Gemastik 8 2015, dan Finalis IBM Linux Challenge 2016. Kegiatan Praktik Kerja Lapang (PKL) yang pernah dilakukan oleh penulis selama 35 hari kerja pada tahun 2015 yaitu di Badan Pusat Pengkajian dan Penerapan Teknologi (BPPT) Serpong dengan mengembangkan suatu sistem informasi bernama Sistem Informasi Administrasi Surat Masuk PDIS (SI-SAMP).