

Kelompok : 1. Lukas Kurnia Jonathan / 13517006
 2. Eginata Kasan / 13517030
 3. Vivianni / 13517060
 4. Gardahadi / 13517144
 5. Rika Dewi / 13517147

Asisten Mentor : Reinaldo Ignatius Wijaya

Asisten Penilai :

1. Deskripsi Umum Aplikasi

Aplikasi Engi's Farm merupakan sebuah permainan simulasi kehidupan Engi dalam mengelola peternakan warisan kakeknya. Permainan akan berakhir ketika seluruh hewan ternak Engi telah tiada.

Engi dapat melakukan berbagai hal sebagai berikut:

1. bergerak ke atas, bawah, kiri, dan kanan,
2. mengambil air dari sumur dan menyiram tanah untuk menumbuhkan rumput sebagai pakan ternak,
3. berbicara dengan hewan ternaknya,
4. mengambil hasil produk dari hewan ternak,
5. membunuh hewan ternak untuk mendapatkan daging hewan ternak,
6. mencampurkan produk dari hewan ternak untuk mendapatkan produk sampingan,
7. menjual hasil ternak dan produk sampingan untuk mendapatkan uang

Pada Engi's Farm, terdapat kumpulan petak tanah (Cell) yang direpresentasikan dengan sebuah matriks 2D dengan ukuran (X,Y). Setiap Cell dapat berupa Land, daerah untuk beternak hewan; atau Facility, yaitu fasilitas peternakan. Land digunakan untuk tempat hidup hewan. Land dapat dikategorikan sebagai Coop, Grassland, atau Barn. Coop digunakan untuk beternak hewan penghasil telur, Grassland digunakan untuk beternak hewan penghasil susu, Barn digunakan untuk beternak hewan penghasil daging.

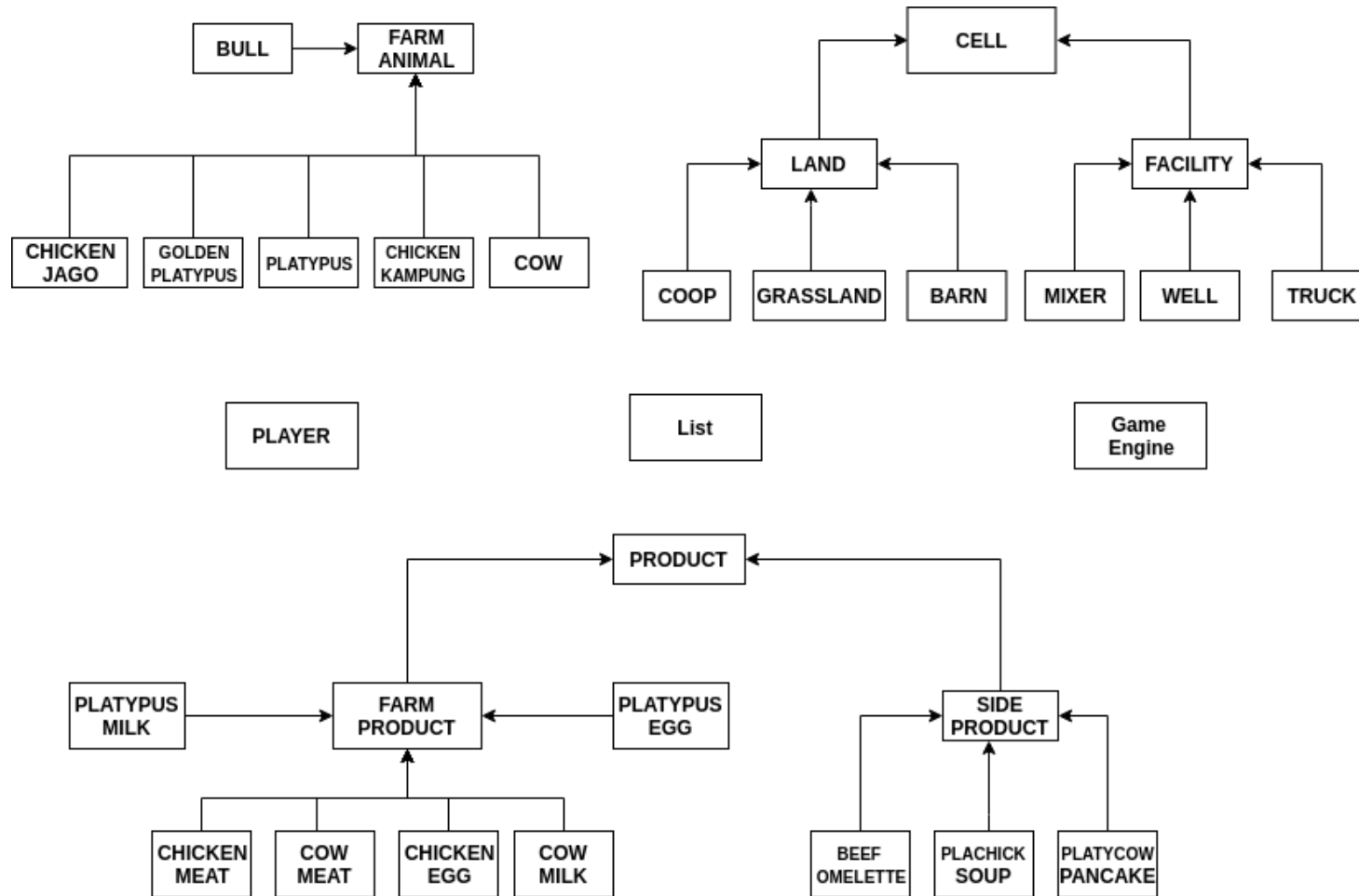
Engi memiliki berbagai macam hewan ternak seperti: ChickenJago, ChickenKampung, Cow, Bull, Platypus, dan GoldenPlatypus. Seluruh hewan ternak milik Engi memakan rumput dari peternakan dan dapat bergerak bebas di lingkungan tempat peternakannya.

Berikut adalah tabel informasi produk yang dihasilkan oleh hewan ternak Engi:

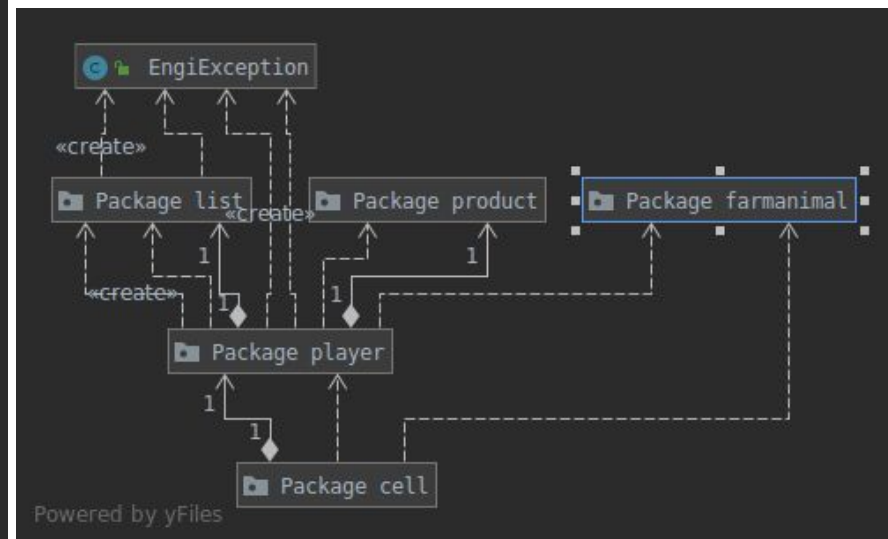
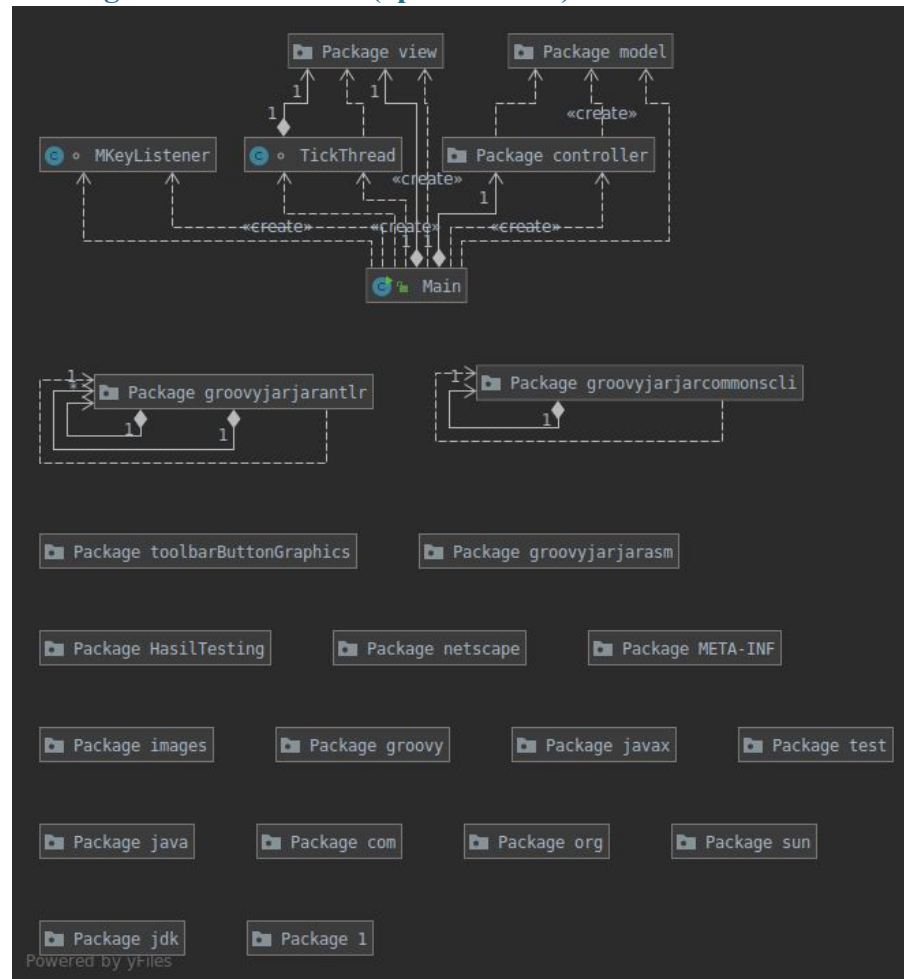
Jenis Hewan	Egg Producing	Milk Producing	Meat Producing
ChickenJago	x	x	✓
ChickenKampung	✓	x	✓
Cow	x	✓	✓
Bull	x	x	✓
Platypus	✓	✓	x
GoldenPlatypus	✓	✓	x

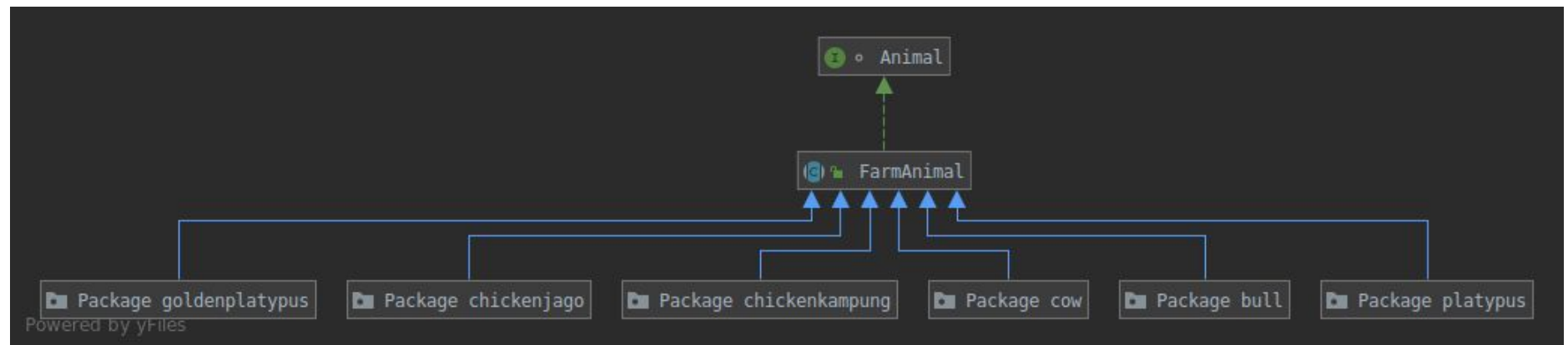
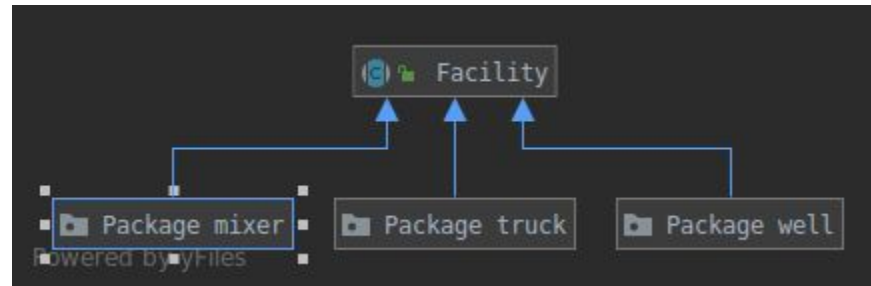
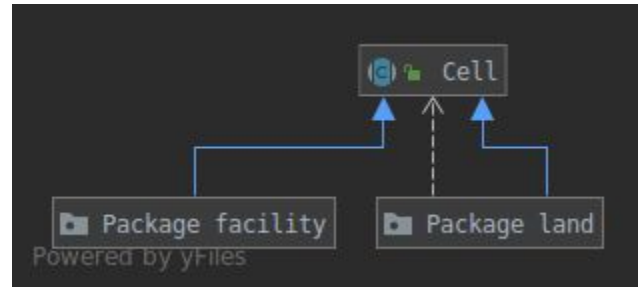
2. Diagram Kelas

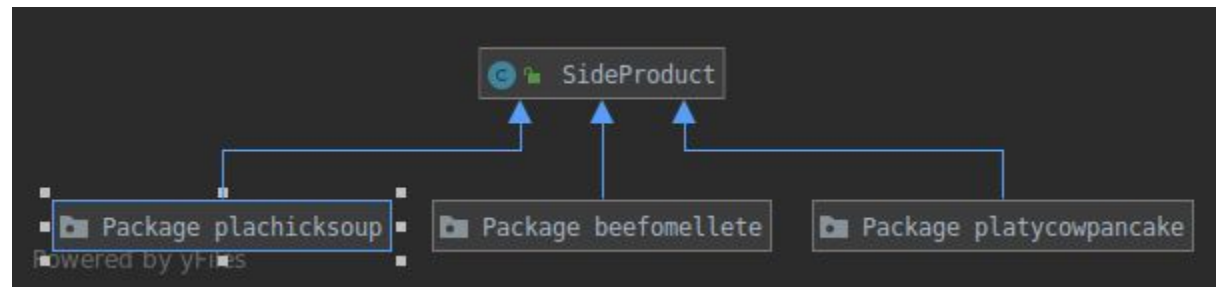
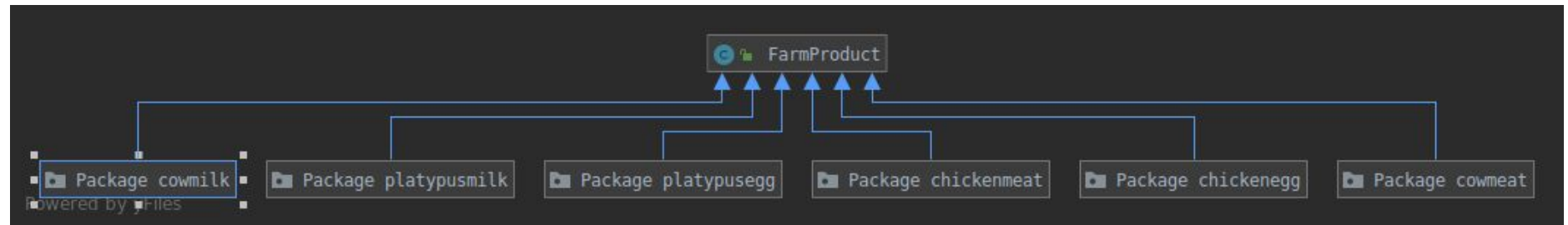
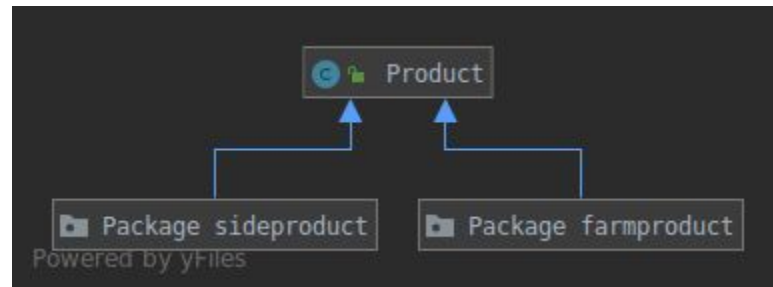
2.1 Diagram Kelas CPP



2.2 Diagram Kelas JAVA (aplikasi baru)







2.3 Ulasan Hasil Translasi

Dalam proses translasi dari bahasa CPP ke Java, kami melakukan hal-hal sebagai berikut:

1. Kelas-kelas dalam bahasa CPP yang beberapa methodnya (tidak seluruhnya) merupakan pure virtual diubah menjadi kelas abstrak dalam Java
2. Kelas-kelas dalam bahasa CPP yang seluruh methodnya adalah pure virtual diubah menjadi interface dalam Java
3. Atribut const dalam bahasa CPP diubah menjadi atribut final di Java
4. Atribut enum dalam bahasa CPP tidak digunakan dan digantikan dengan atribut bertipe String dalam Java
5. Beberapa perubahan sintaks sehingga sesuai dengan bahasa Java seperti bool menjadi boolean, NULL menjadi null, dll.

Dalam aplikasi yang telah dibuat, terdapat perubahan diagram kelas yang terjadi, yaitu: interface *Animal* yang ditambahkan di atas *FarmAnimal*. Perubahan ini dikarenakan adanya kewajiban untuk menggunakan interface dalam pembuatan aplikasi.

3. Implementasi Perubahan

3.1 Tabel Keruntutan

Package	Nama Kelas CPP	Nama Method	Nama Kelas JAVA	Nama Method JAVA	Reuse Ya/Tidak	Deskripsi Perubahan
Model	-	-	Animal	isMeat() isEgg() isMilk() makan()	Ya	Menambahkan interface Animal yang di impements oleh FarmAnimal

4. Skenario Test

Skenario	Hasil
Melakukan perintah “GD” untuk menggerak Player ke bawah	Player bergerak ke bawah
Melakukan perintah “GU” untuk menggerak Player ke atas	Player bergerak ke atas
Melakukan perintah “GL” untuk menggerak Player ke bawah	Player bergerak ke kiri
Melakukan perintah “GR” untuk menggerak Player ke bawah	Player bergerak ke kanan

Menggerakan player ke arah ujung map, hewan, dan facility	Player tidak bergerak, keluar pesan error pada layar
Menggunakan perintah “KILL” untuk membunuh hewan penghasil daging	Jumlah hewan berkurang sesuai jumlah yang di KILL, hewan yang di KILL hilang dari layar
Menggunakan perintah “KILL” pada hewan bukan penghasil daging	Mengeluarkan pesan error pada layar
Melakukan perintah “INTERACT” saat dekat dengan Mixer	Barang pada inventory berubah menjadi hasil olahan mixer
Melakukan perintah “INTERACT” saat dekat dengan Well	Jumlah water player bertambah
Melakukan perintah “INTERACT” saat dekat dengan Truck	Jumlah Money bertambah, barang pada inventory hilang
Melakukan perintah “INTERACT” saat tidak dekat dengan facility	Tidak terjadi apa-apa
Melakukan perintah “TALK” saat dekat hewan	Mengeluarkan pesan animal yang berdekatan dengannya
Melakukan perintah “GROW” saat player berada pada land tidak berumput	Land menjadi grassland berumput, water player berkurang
Membuat jumlah animal 0	Program berhenti
Membuat hewan mati kelaparan	Jumlah hewan berkurang 1
Melakukan random movement pada animal	Animal move randomly

5. Pembagian Tugas dan Log Activity

5.1 Detail Pembagian Tugas Koding

Package	Class	Designer	Implementor	Tester
Controller	GameEngine	Semua Anggota Kelompok	Lukas	Vivianni
-	Main	Semua Anggota Kelompok	Gardahadi	Gardahadi
View	MainFrame	GardaHadi	Gardahadi	Gardahadi
Model	FarmAnimal	Semua Anggota Kelompok	Eginata	Vivianni
Model	Cell, Land, Facility	Semua Anggota Kelompok	Rika	Vivianni
Model	Player, Product	Semua Anggota Kelompok	Vivianni	Vivianni

5.2 Detail Pembagian Tugas Implementasi

Elemen Dokumentasi	Penanggung Jawab
Dokumen FarmAnimal dan turunannya	Eginata
Dokumen Product dan turunannya	Vivianni
Dokumen Cell dan turunannya	Rika
Dokumen Player	Rika
Dokumen GameEngine	Lukas
Dokumen View (Mainframe) dan main program	Gardahadi

5.3 Log Activity

No	Mulai	Selesai	Kegiatan	Hasil	Keterangan
1.	10 April 2019 18.00	10 April 2019 18.30	Asistensi	Mendapatkan masukan untuk menggunakan IntelliJ, berhati-hati dalam passing parameter di Java	Seluruh anggota kelompok
2.	12 April 2019 19.00	12 April 2019 19.05	Inisialisasi Repository	Repository sudah ada (Github terbentuk)	Rika Dewi
3.	14 April 2019 10.00	14 April 2019 21.00	Menerjemahkan FarmAnimal dan turunannya ke java	FarmAnimal.java, Bull.java, Cow.java, ChickenKampung.java, ChickenJago.java, Platypus.java, GoldenPlatypus.java terbentuk	Eginata
4.	14 April 2019 20.00	15 April 2019 12.00	Menerjemahkan Cell dan turunannya ke java	Cell.java, Facility.java, Land.java, Truck.java, Well.java, Mixer.java terbentuk	Rika
5.	14 April 2019 20.00	15 April 2019 20.00	Menerjemahkan GameEngine ke java	GameEngine.java terbentuk	Lukas
6.	15 April 2019 20.00	16 April 2019 20.00	Menerjemahkan Player, Product dan turunannya ke Java	Player, product dan turunannya terbentuk	Vivianni
7.	17 April 2019 10.00	18 April 2019 21.00	Perbaikan bug pada program	beberapa bug dalam program yang ditemukan sudah diperbaiki	Rika

8.	18 April 2019 11.00	18 April 2019 23.00	Menerjemahkan List generik ke Java	List disesuaikan dengan generik di java	Lukas
9.	18 April 2019 20.00	18 April 2019 21.00	Memperbaiki atribut Jumlah pada FarmAnimal	Atribut Jumlah berupa atribut statis protected dan hanya dapat dikurangi oleh kelas luar	Eginata
10.	19 April 2019 18.00	19 April 2019 23.00	Menambahkan testing pada FarmProduct	Testing FarmProduct sudah terbentuk	Vivianni
11.	19 April 2019 11.00	19 April 2019 23.00	Inisialisasi pembuatan UI	UI dasar untuk engifarm dibuat	Gardahadi
12.	20 April 2019 18.00	20 April 2019 00.00	Implementasi multithreading pada program dan fungsi render pada mainframe	Program dapat melakukan multithreading pada tick dan fungsi render sudah ditambahkan dalam mainframe	Gardahadi
13.	20 April 2019 22.00	21 April 2019 03.00	Implementasi sisa fitur pada UI	UI sudah terbentuk dengan baik	Gardahadi
14.	20 April 2019 20.00	21 April 2019 03.00	Debugging dan testing program	Perbaikan bug yang ditemukan pada Program dan menyelesaikan testing	Seluruh anggota kelompok
15.	21 April 2019 03.00	21 April 2019	Membereskan dokumentasi dan program	Program dan dokumentasi selesai	Seluruh anggota kelompok

6. Pengukuran Metriks Aplikasi

Package	Class Count	Abstract Class Count	Ca	Ce	A	I	D
default	4	0	0	12	0	1	0
model	1	0	14	1	0	0.07	0.93
model.cell	1	1	2	4	1	0.67	0.67
model.cell.facility	1	1	3	3	1	0.5	0.5
model.cell.facility.mixer	1	0	0	2	0	1	0
model.cell.facility.truck	1	0	0	3	0	1	0
model.cell.facility.well	1	0	0	4	0	1	0
model.cell.land	2	0	0	7	0	1	0
model.farmanimal	1	1	10	3	1	0.23	0.23
model.farmanimal.bull	1	0	0	5	0	1	0
model.farmanimal.chickenjago	1	0	0	5	0	1	0
model.farmanimal.chickenkampung	1	0	0	6	0	1	0
model.farmanimal.cow	1	0	0	6	0	1	0

model.farmanimal.gold enplatypus	1	0	0	7	0	1	0
model.farmanimal.plat ypus	1	0	0	7	0	1	0
model.list	1	0	1	2	0	0.67	0.33
modell.player	1	0	3	8	0	0.73	0.27
model.product	1	0	3	1	0	0.25	0.75
model.product.farmpro duct	1	0	13	1	0	0.07	0.93
model.product.farmpro duct.chickenegg	1	0	1	1	0	0.5	0.5
model.product.farmpro duct.chickenmeat	1	0	2	1	0	0.33	0.67
model.product.farmpro duct.cowmeat	1	0	2	1	0	0.33	0.67
model.product.farmpro duct.cowmilk	1	0	1	1	0	0.5	0.5
model.product.farmpro duct.platypusegg	1	0	2	1	0	0.33	0.67
model.product.farmpro duct.platypusmilk	1	0	2	1	0	0.33	0.67

model.product.sideproduct	1	0	3	1	0	0.25	0.75
model.product.sideproduct.beefomellete	1	0	1	1	0	0.5	0.5
model.product.sideproduct.plachicksoup	1	0	1	1	0	0.5	0.5
model.product.sideproduct.platycowpancake	1	0	1	1	0	0.5	0.5

Analisis :

- Dilihat dari nilai Ca, aplikasi kami memiliki range $0 \leq Ca \leq 14$ dengan didominasi 0,1,2,3 yang berarti aplikasi kami tidak terlalu bergantung pada kelas lain.
- Dilihat dari nilai Ce, aplikasi kami memiliki range $1 \leq Ce \leq 12$ yang berarti aplikasi kami stabil sehingga mempermudah proses development.
- Dilihat dari nilai I, terdapat 13 dari 29 package kami memiliki range $0,3 < I < 0,7$ yang berarti hampir setengah dari aplikasi memiliki package dengan kestabilan medium. Hal ini cenderung tidak baik, karena package yang baik harusnya sangat stabil atau sangat tidak stabil.
- Dilihat dari nilai A, jumlah dari I+A package kami memiliki nilai cenderung jauh dari 1, sehingga ketidakstabilan package kami tidak diimbangi dengan tingkat abstraknya. Hal ini bukan desain yang baik dalam package.
- Dilihat dari nilai D, terdapat 9 dari 29 package kami memiliki nilai $0.5 < D$ sehingga hanya sekitar 30% dari package kami yang kurang seimbang.

Kesimpulan metrik :

- Berdasarkan hasil analisis, dapat disimpulkan aplikasi secara garis besar sudah mengimplementasikan desain kelas yang baik, namun hal ini masih bisa ditingkatkan. Terutama dengan meningkatkan abstraksi dari beberapa kelas.