

VIRTUAL ZOO (VZ.03)

DOKUMENTASI UTAMA

Diajukan untuk memenuhi Tugas Besar I IF2210 Pemrograman Berorientasi Objek

oleh

MEINZOO:

OKTAVIANUS HANDIKA / 13515035

FELIX LIMANTA / 13515065

RIONALDI CHANDRASETA / 13515077

HOLY LOVENIA / 13515113



TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2017

DAFTAR ISI

DAFTAR GAMBAR	4
DAFTAR TABEL	6
BAB I	7
BAB II	8
2.1 Kelas AirHabitat.....	8
2.2 Kelas Animal.....	9
2.3 Kelas AnimalBehavior.....	11
2.4 Kelas AnimalDiet	13
2.5 Kelas Aves.....	14
2.6 Kelas Barracuda.....	15
2.7 Kelas BehaviorTame.....	16
2.8 Kelas BehaviorWild	17
2.9 Kelas Cage	17
2.10 Kelas Carnivora.....	18
2.11 Kelas Cell	18
2.12 Kelas Chameleon.....	19
2.13 Kelas Clownfish	19
2.14 Kelas Colibri.....	20
2.15 Kelas Crocodile.....	21
2.16 Kelas Driver	21
2.17 Kelas Duck	22
2.18 Kelas Eagle.....	22
2.19 Kelas Elephant.....	23
2.20 Kelas Facility.....	24
2.21 Kelas Giraffe	25
2.22 Kelas Habitat	25
2.23 Kelas Herbivora	26
2.24 Kelas Iguana	27
2.25 Kelas Komodo.....	28
2.26 Kelas LandHabitat	28
2.27 Kelas Lion	29
2.28 Kelas Mammalia.....	29
2.29 Kelas Monkey.....	30
2.30 Kelas Omnivora	30
2.31 Kelas Owl.....	31

2.32 Kelas Park	31
2.33 Kelas Peacock	32
2.34 Kelas Person	33
2.35 Kelas Pisces	34
2.36 Kelas Point	34
2.37 Kelas Python	35
2.38 Kelas Renderable	36
2.39 Kelas Reptilia	37
2.40 Kelas Restaurant	37
2.41 Kelas Road	38
2.42 Kelas RoadEntrance	38
2.43 Kelas RoadExit	39
2.44 Kelas Shark	39
2.45 Kelas Tiger	40
2.46 Kelas WaterHabitat	41
2.47 Kelas WildBunny	41
2.48 Kelas WildColibri	42
2.49 Kelas Wolf	42
2.50 Kelas Zebra	43
2.51 Kelas Zoo	43
BAB III	44
3.1 Rancangan Akhir	44
3.2 Rancangan Awal	44
BAB IV	45
4.1 Direktori Root	45
4.1.1 Direktori src	45
BAB V	2
5.1 Skenario Test	2
5.1.1 Functional Test	2
5.1.2 Unit Test	3
5.2 Penghitungan Metriks Perangkat Lunak	4
5.3 Log Activity	5
5.3.1 Pembagian peran	5

DAFTAR GAMBAR

Gambar 1 UML AirHabitat	8
Gambar 2 UML Animal	10
Gambar 3 UML AnimalBehavior	12
Gambar 4 UML AnimalDiet	14
Gambar 5 UML Aves	14
Gambar 6 UML Barracuda	15
Gambar 7 UML BehaviorTame	17
Gambar 8 UML BehaviorWild	17
Gambar 9 UML Cage	17
Gambar 10 UML Carnivora	18
Gambar 11 UML Cell	18
Gambar 12 UML Chameleon	19
Gambar 13 UML Clownfish	19
Gambar 14 UML Colibri	20
Gambar 15 UML Crocodile	21
Gambar 16 UML Driver	21
Gambar 17 UML Duck	22
Gambar 18 UML Eagle	23
Gambar 19 UML Elephant	23
Gambar 20 UML Facility	24
Gambar 21 UML Giraffe	25
Gambar 22 UML Habitat	26
Gambar 23 UML Herbivora	26
Gambar 24 UML Iguana	27
Gambar 25 UML Komodo	28
Gambar 26 UML LandHabitat	28
Gambar 27 UML Lion	29
Gambar 28 UML Mammalia	29
Gambar 29 UML Monkey	30
Gambar 30 UML Omnivora	30
Gambar 31 UML Owl	31
Gambar 32 UML Park	32
Gambar 33 UML Peacock	32
Gambar 34 UML Person	33
Gambar 35 UML Pisces	34
Gambar 36 UML Point	34
Gambar 37 UML Python	35
Gambar 38 UML Renderable	36
Gambar 39 UML Reptilia	37
Gambar 40 UML Restaurant	37
Gambar 41 UML Road	38
Gambar 42 UML RoadEntrance	38
Gambar 43 UML RoadExit	39
Gambar 44 UML Shark	40

Gambar 45 UML Tiger	40
Gambar 46 WaterHabitat	41
Gambar 47 UML WildBunny	41
Gambar 48 UML WildColibri.....	42
Gambar 49 UML Wolf	42
Gambar 50 UML Zebra	43
Gambar 51 UML Zoo	43
Gambar 52 UML Animal (rancangan awal).....	44
Gambar 53 Snapshot Root.....	45
Gambar 54 Snapshot src	45
Gambar 55 Snapshot animal	45
Gambar 56 Snapshot aves	45
Gambar 57 Snapshot behavior	45
Gambar 58 Snapshot diet	45
Gambar 59 Snapshot mammalia.....	2
Gambar 60 Snapshot pisces	2
Gambar 61 Snapshot reptilian.....	2
Gambar 62 Snapshot infrastructure.....	2
Gambar 63 Snapshot facility.....	2
Gambar 64 Snapshot habitat	2
Gambar 65 Snapshot misc	2

DAFTAR TABEL

Tabel 1 Functional Test	2
Tabel 2 Unit Test	4
Tabel 3 Penilaian Metriks Perangkat Lunak.....	5

BAB I

DESKRIPSI UMUM APLIKASI

Virtual Zoo adalah kebun binatang yang terdiri dari sebidang tanah dengan luas 20 x 20 karakter. Pengunjung dapat melakukan *tour* untuk mengelilingi kebun binatang dan berinteraksi dengan hewan. Dalam kebun binatang tersebut, terdapat 20 jenis hewan yang berbeda dan beragam infrastruktur bangunan yang dapat dimanfaatkan oleh pengunjung.

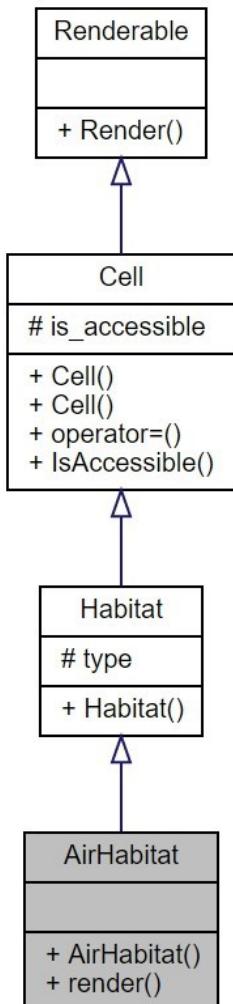
Setiap jenis hewan dapat diklasifikasikan berdasarkan tipe makanan (karnivora / herbivora / omnivora), taksonomi (burung / mamalia / reptil / ikan), tempat tinggal (air / darat / udara), dan perilaku (liar / jinak). Makanan, baik yang berwujud daging ataupun tumbuhan, yang dibutuhkan setiap hewan dapat dihitung dan diakumulasi untuk mengetahui banyak kebutuhan makanan yang harus disediakan oleh kebun binatang. Hewan hanya dapat tinggal di kandang dengan habitat yang sesuai. Hewan yang liar tidak dapat digabungkan dalam satu kandang dengan jenis hewan tertentu lainnya.

Infrastruktur bangunan bisa berbentuk fasilitas dan habitat. Pengunjung dapat mengakses fasilitas tertentu, tetapi habitat pasti tidak dapat dilewati. Fasilitas bisa berupa taman, restoran, dan jalan. Jalan dapat diklasifikasi sebagai jalan biasa, jalan masuk, dan jalan keluar. Sedangkan, habitat dapat dibagi menjadi habitat air, habitat darat, dan habitat udara. Sebuah kandang hewan hanya memiliki satu jenis habitat.

BAB II

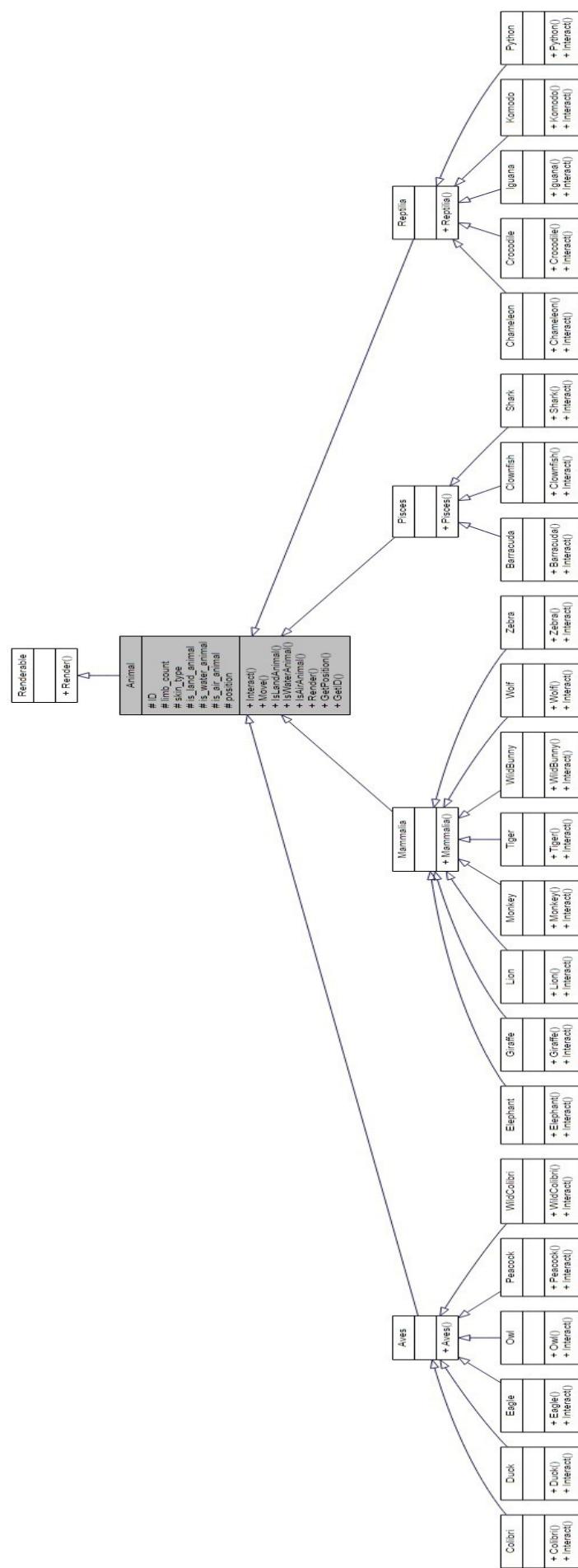
RANCANGAN KELAS

2.1 Kelas AirHabitat



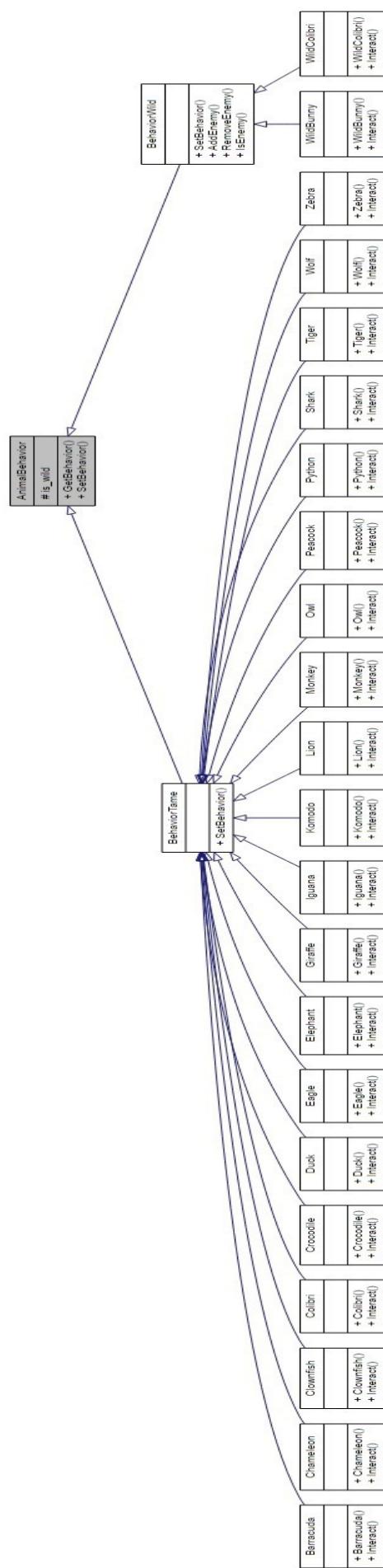
Gambar 1 UML AirHabitat

2.2 Kelas Animal



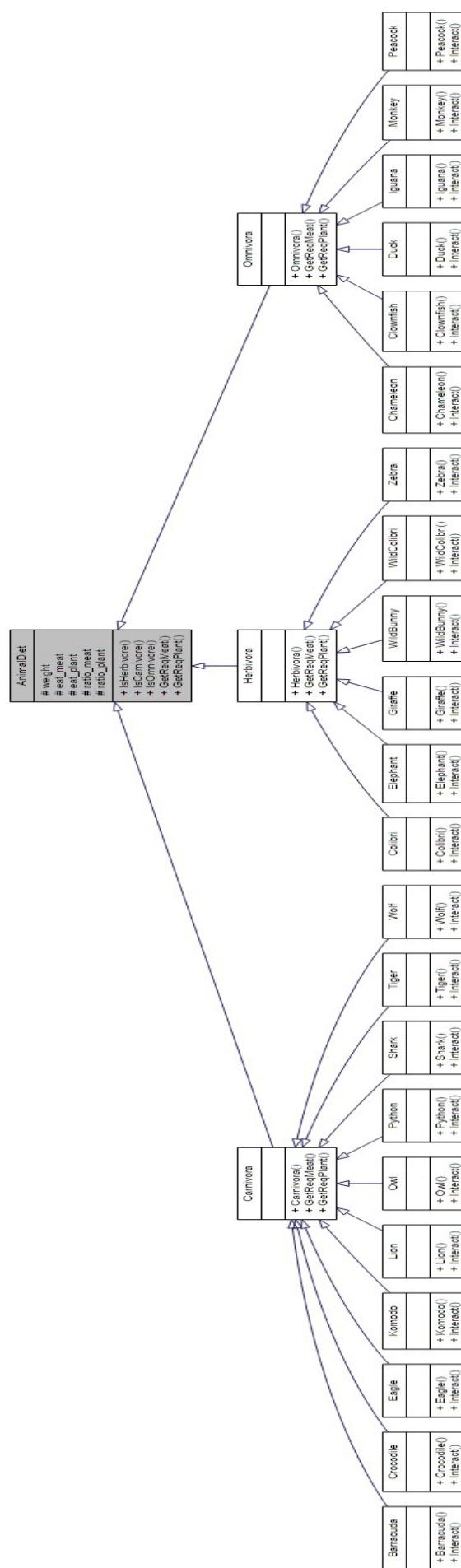
Gambar 2 UML Animal

2.3 Kelas AnimalBehavior



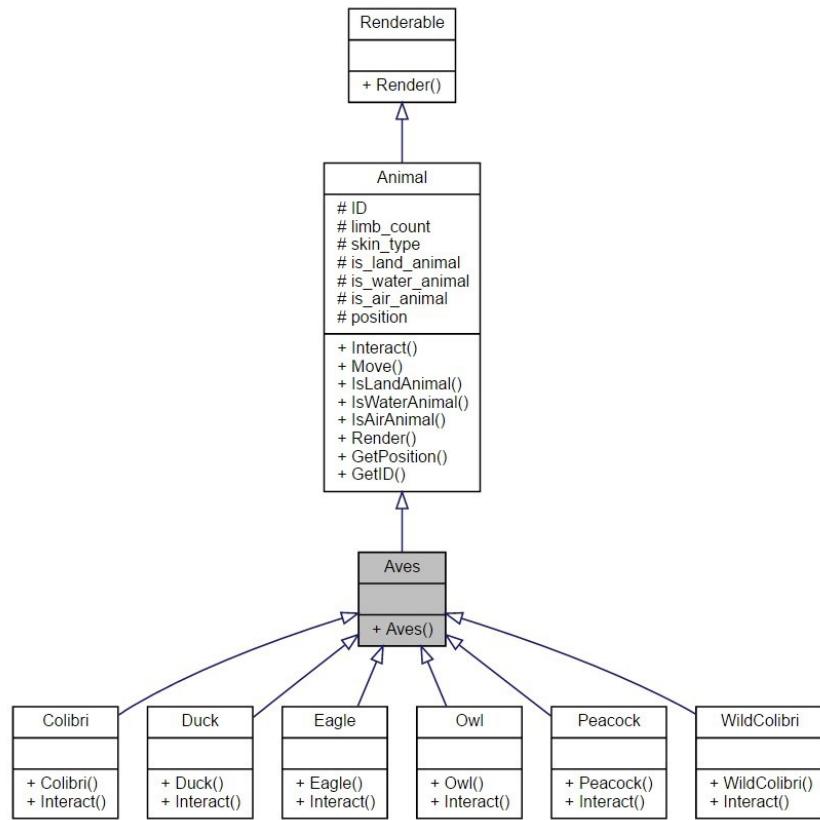
Gambar 3 UML AnimalBehavior

2.4 Kelas AnimalDiet



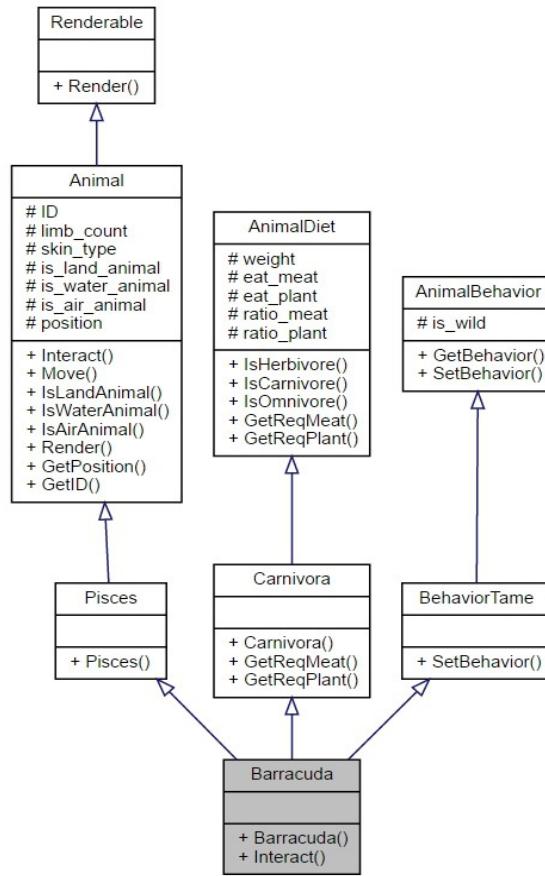
Gambar 4 UML AnimalDiet

2.5 Kelas Aves



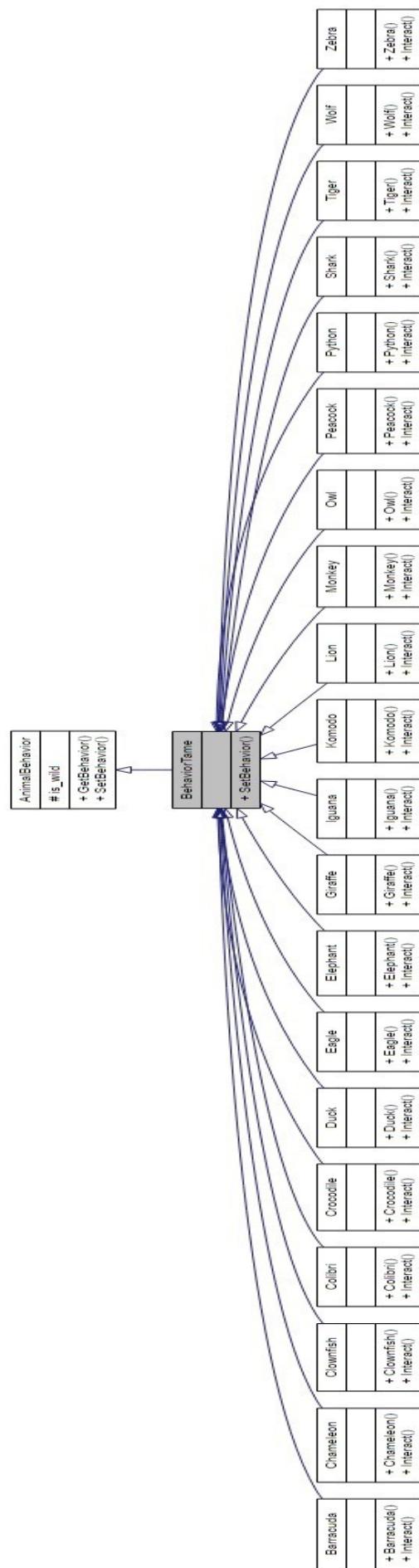
Gambar 5 UML Aves

2.6 Kelas Barracuda



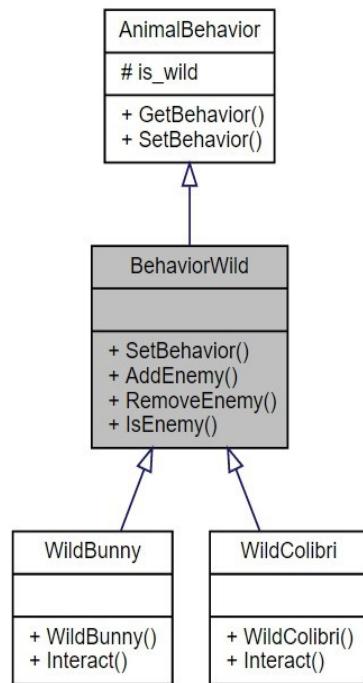
Gambar 6 UML Barracuda

2.7 Kelas BehaviorTame



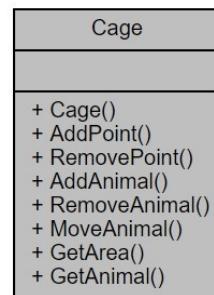
Gambar 7 UML BehaviorTame

2.8 Kelas BehaviorWild



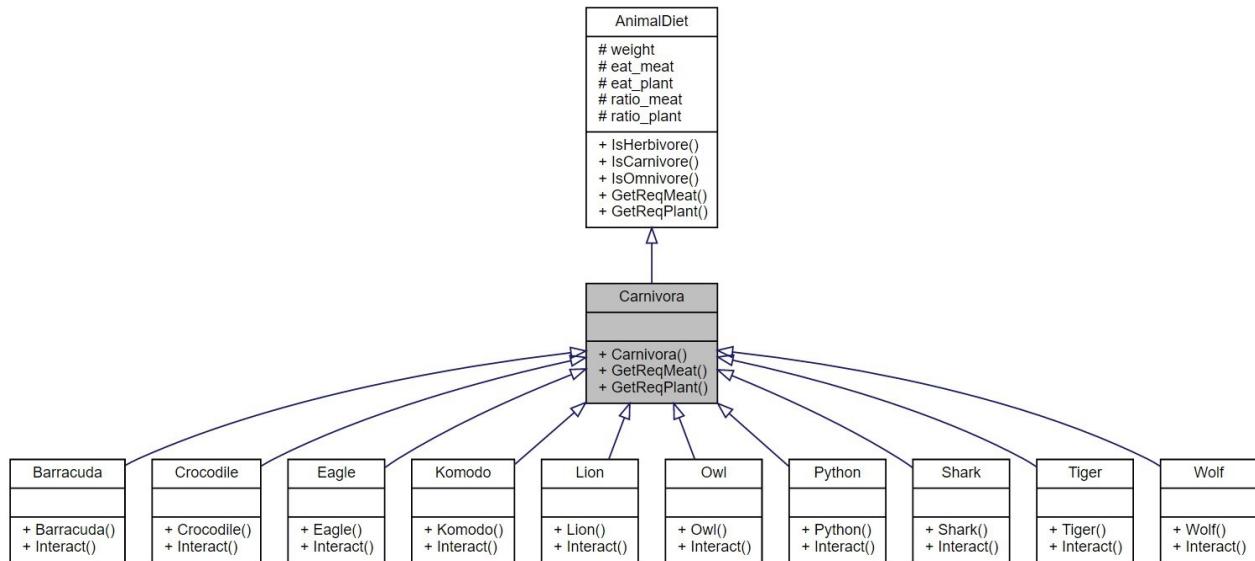
Gambar 8 UML BehaviorWild

2.9 Kelas Cage



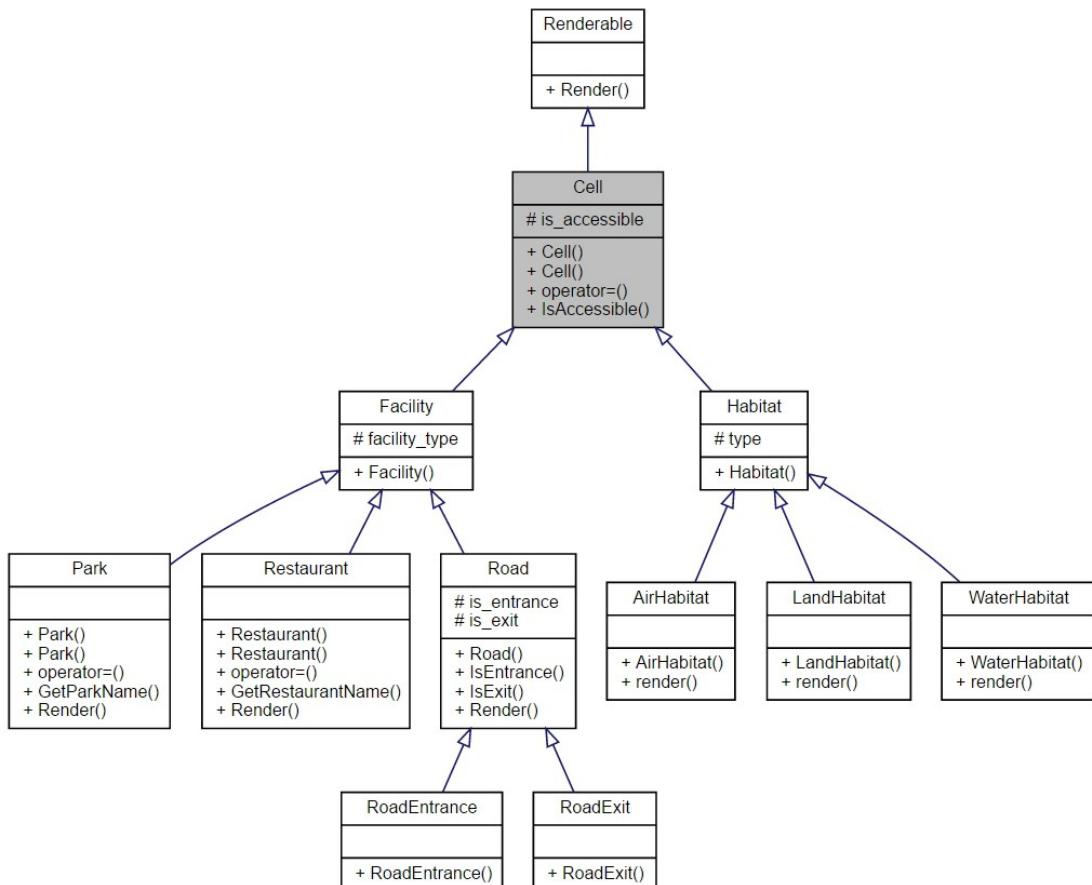
Gambar 9 UML Cage

2.10 Kelas Carnivora



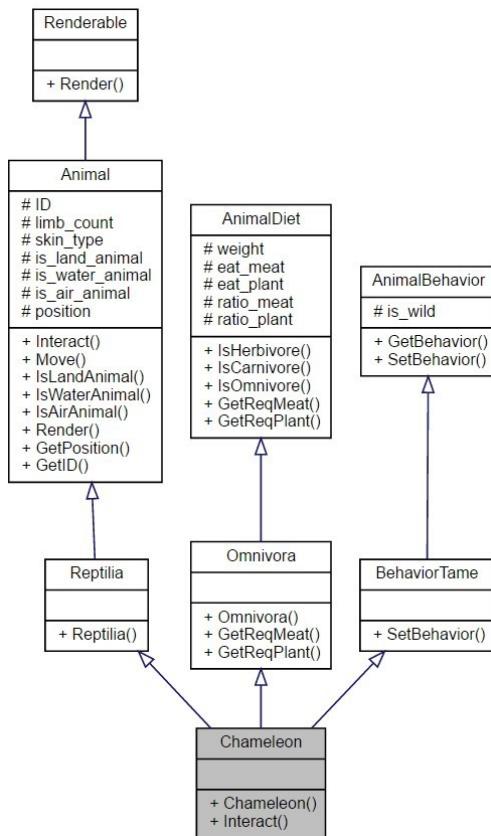
Gambar 10 UML Carnivora

2.11 Kelas Cell



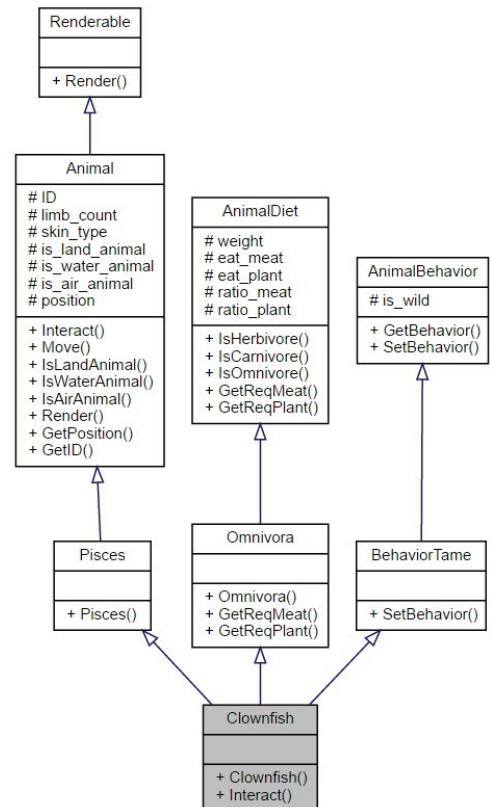
Gambar 11 UML Cell

2.12 Kelas Chameleon



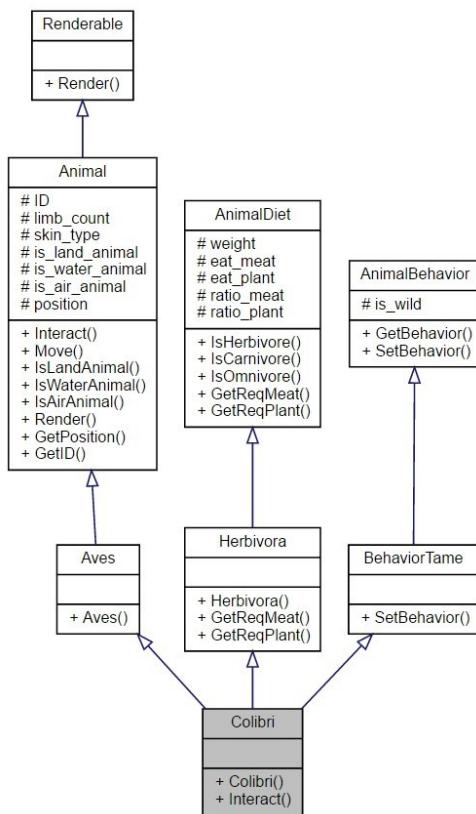
Gambar 12 UML Chameleon

2.13 Kelas Clownfish



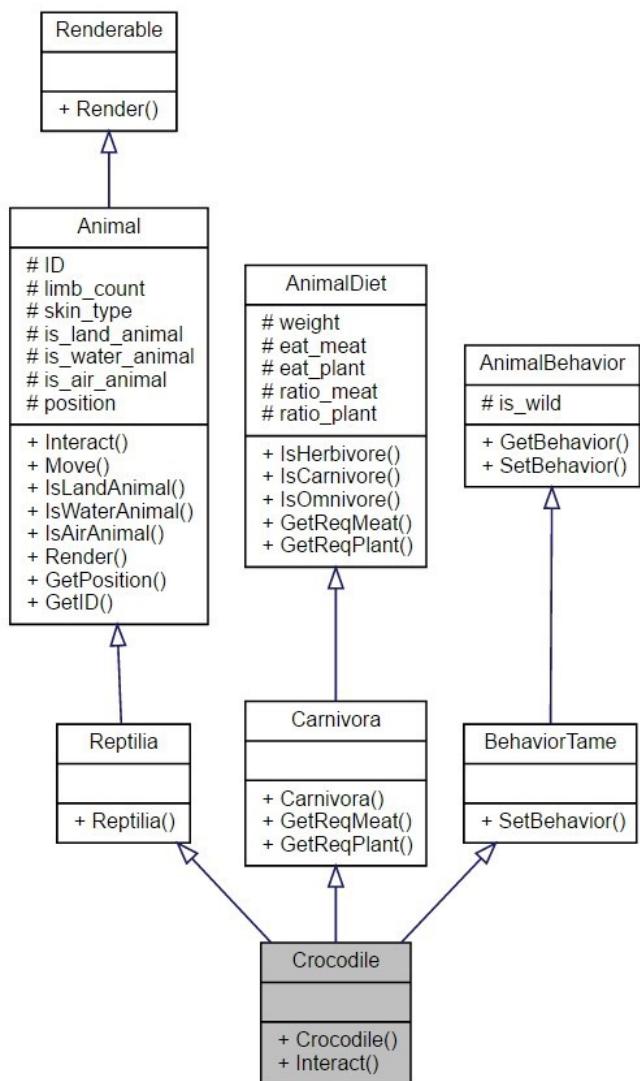
Gambar 13 UML Clownfish

2.14 Kelas Colibri



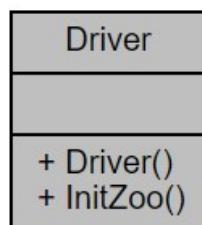
Gambar 14 UML Colibri

2.15 Kelas Crocodile



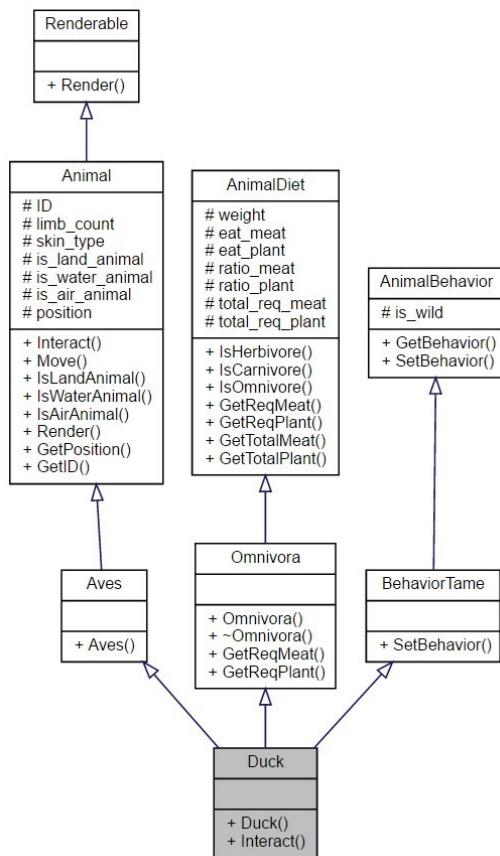
Gambar 15 UML Crocodile

2.16 Kelas Driver



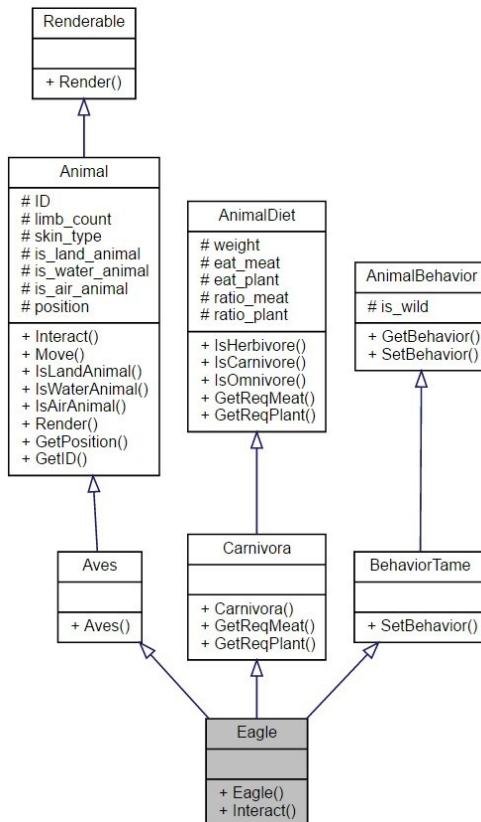
Gambar 16 UML Driver

2.17 Kelas Duck



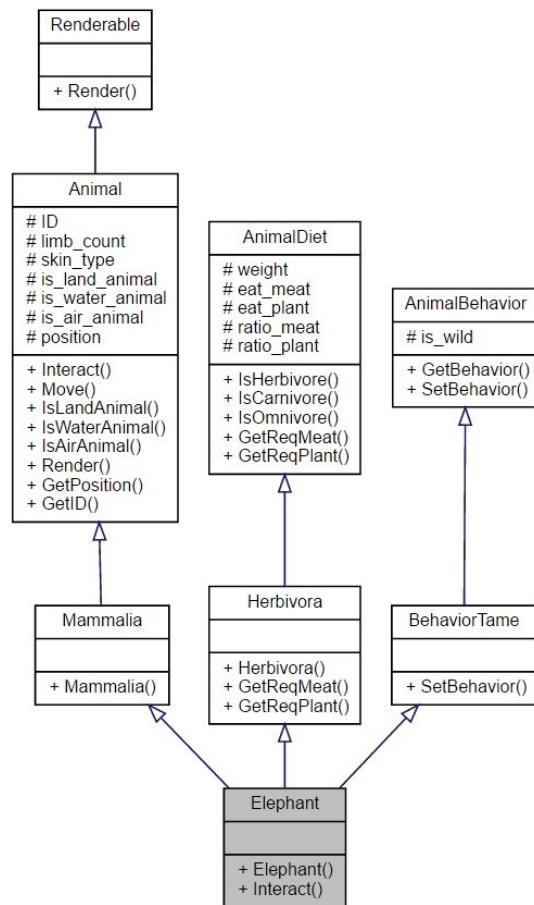
Gambar 17 UML Duck

2.18 Kelas Eagle



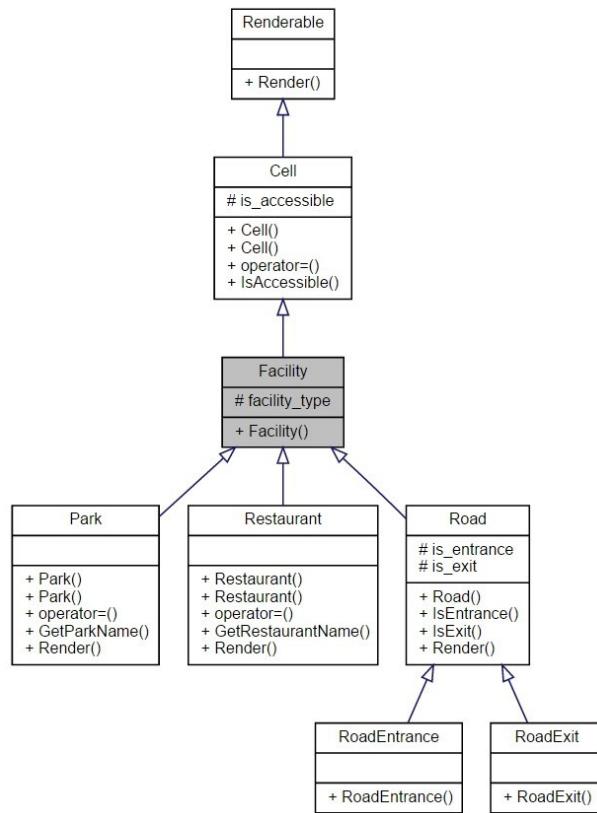
Gambar 18 UML Eagle

2.19 Kelas Elephant



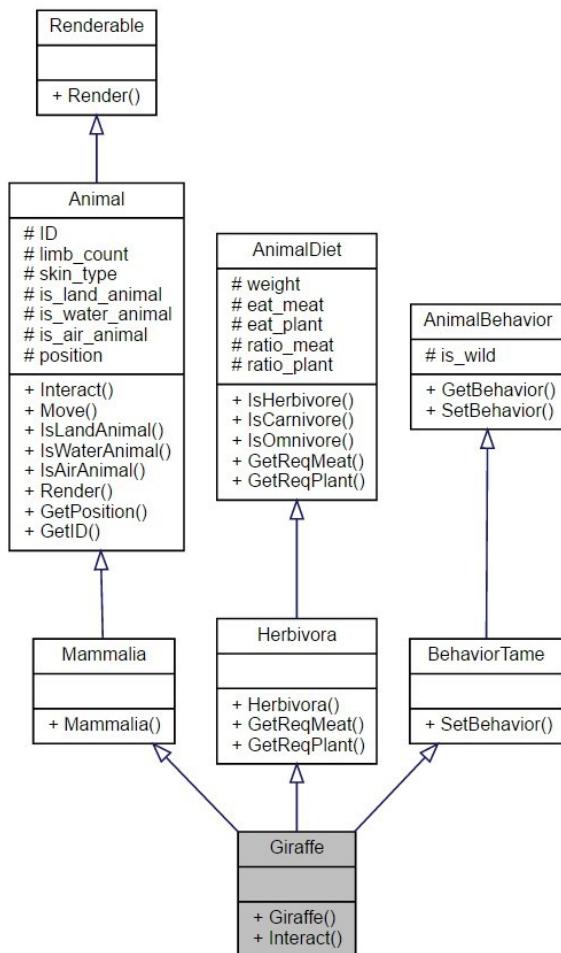
Gambar 19 UML Elephant

2.20 Kelas Facility



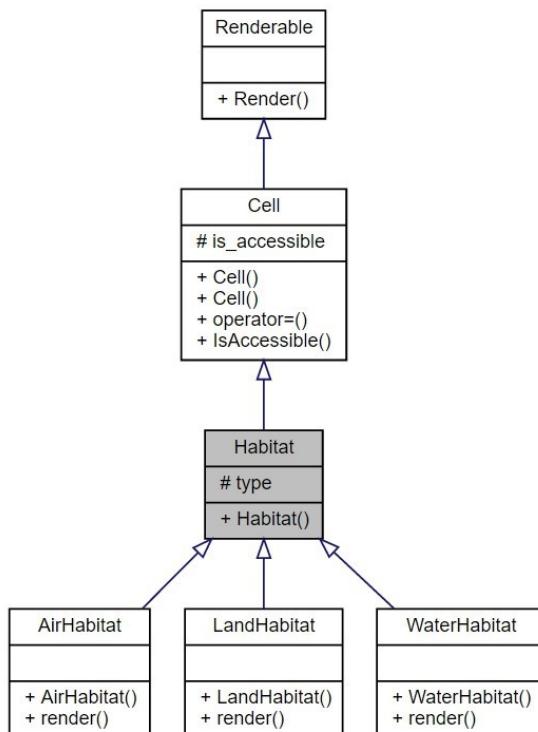
Gambar 20 UML Facility

2.21 Kelas Giraffe



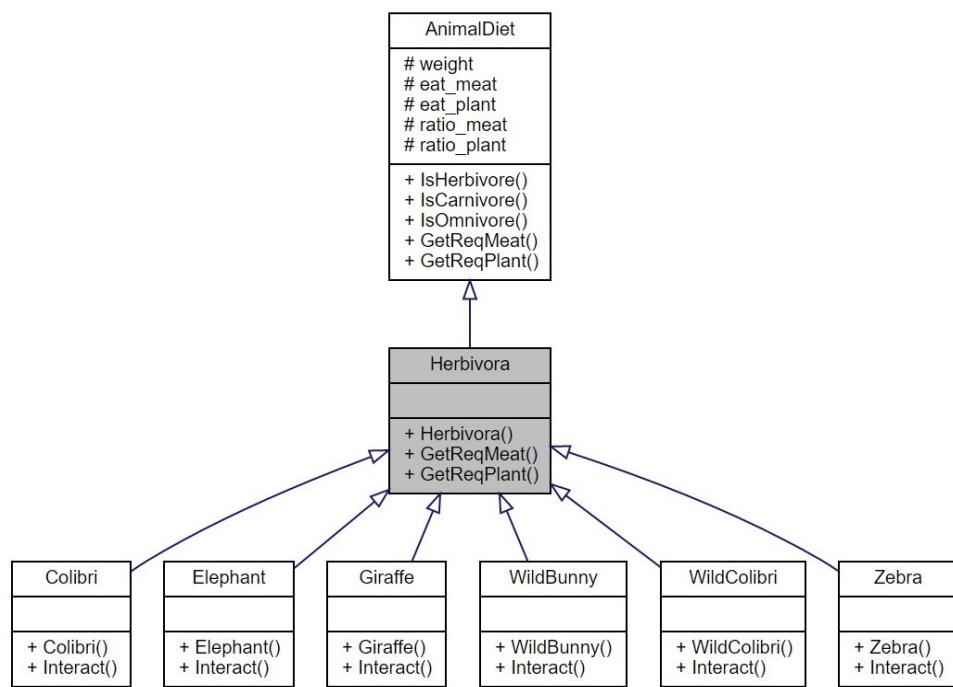
Gambar 21 UML Giraffe

2.22 Kelas Habitat



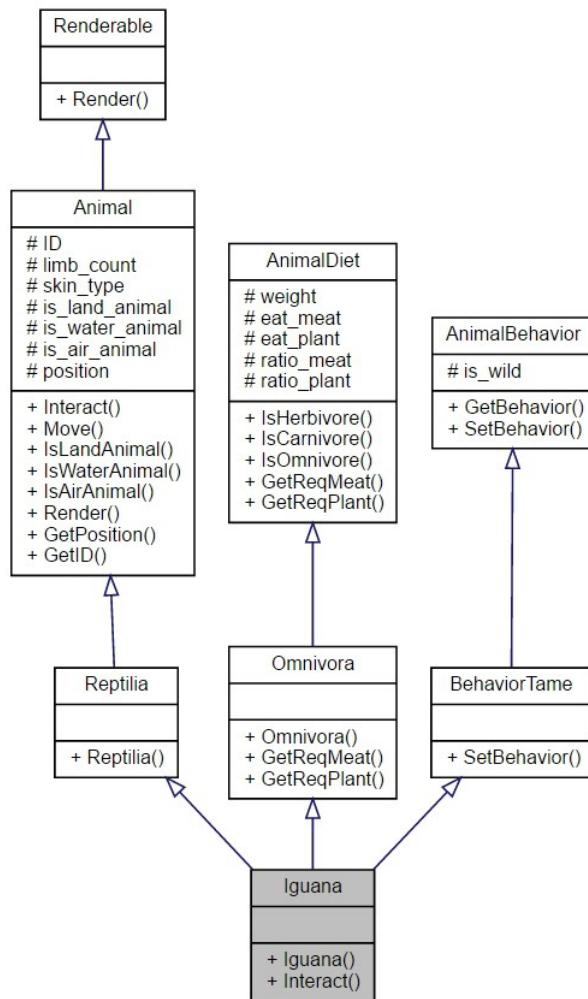
Gambar 22 UML Habitat

2.23 Kelas Herbivora



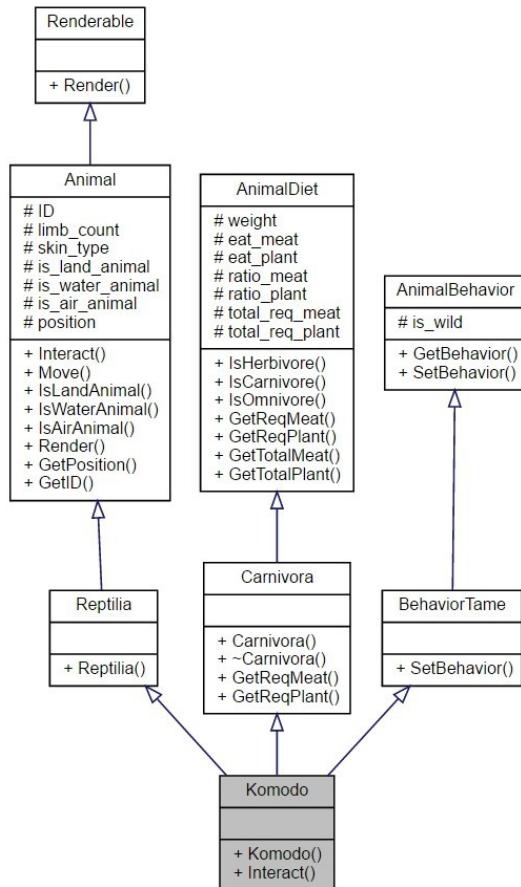
Gambar 23 UML Herbivora

2.24 Kelas Iguana



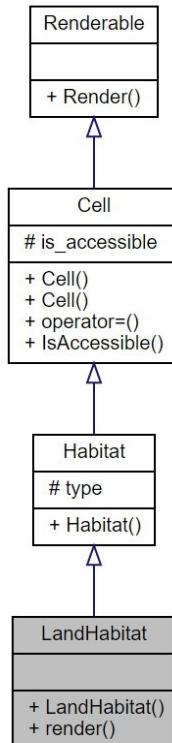
Gambar 24 UML Iguana

2.25 Kelas Komodo



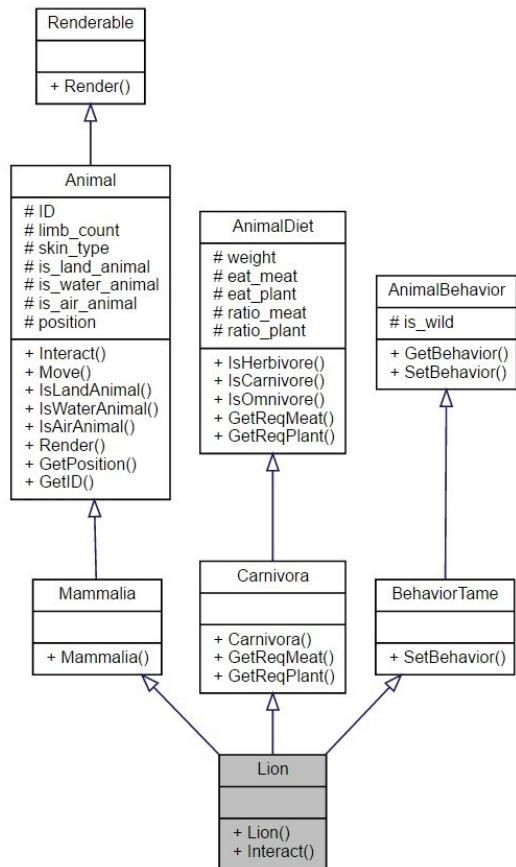
Gambar 25 UML Komodo

2.26 Kelas LandHabitat



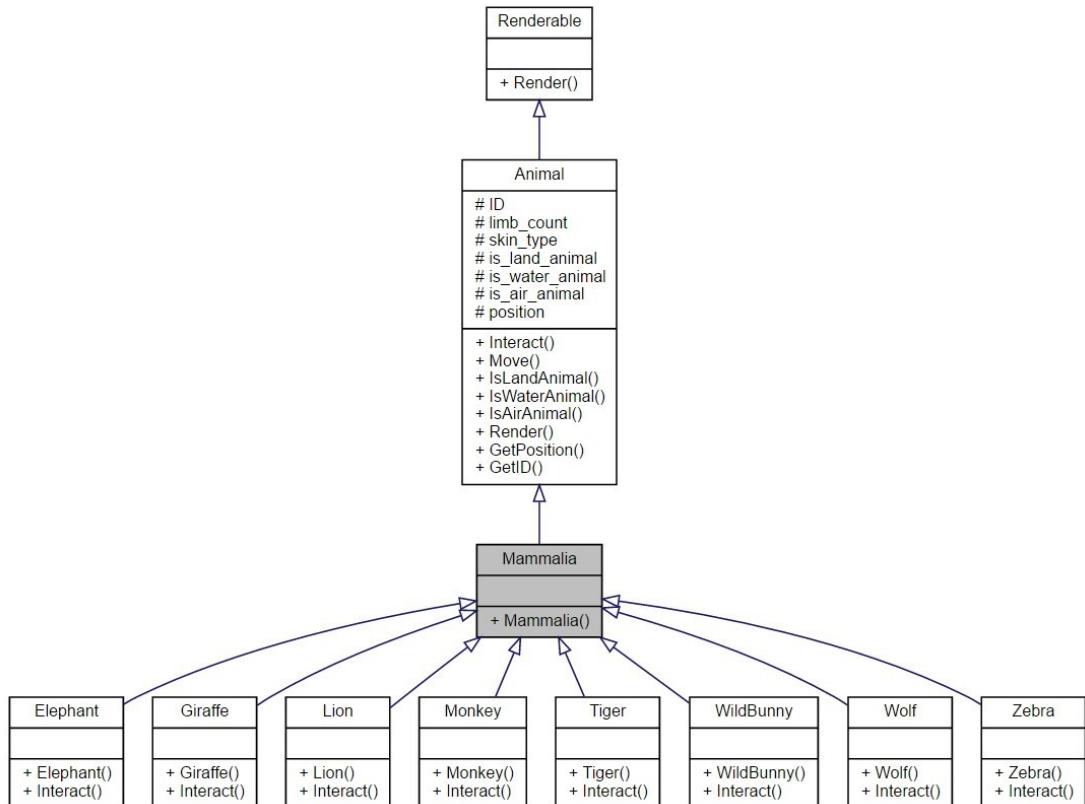
Gambar 26 UML LandHabitat

2.27 Kelas Lion



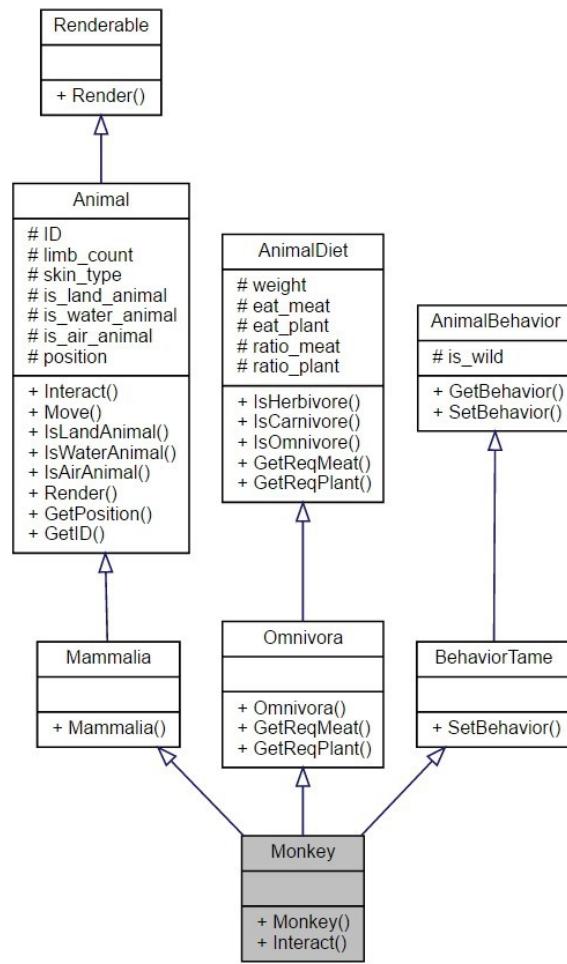
Gambar 27 UML Lion

2.28 Kelas Mammalia



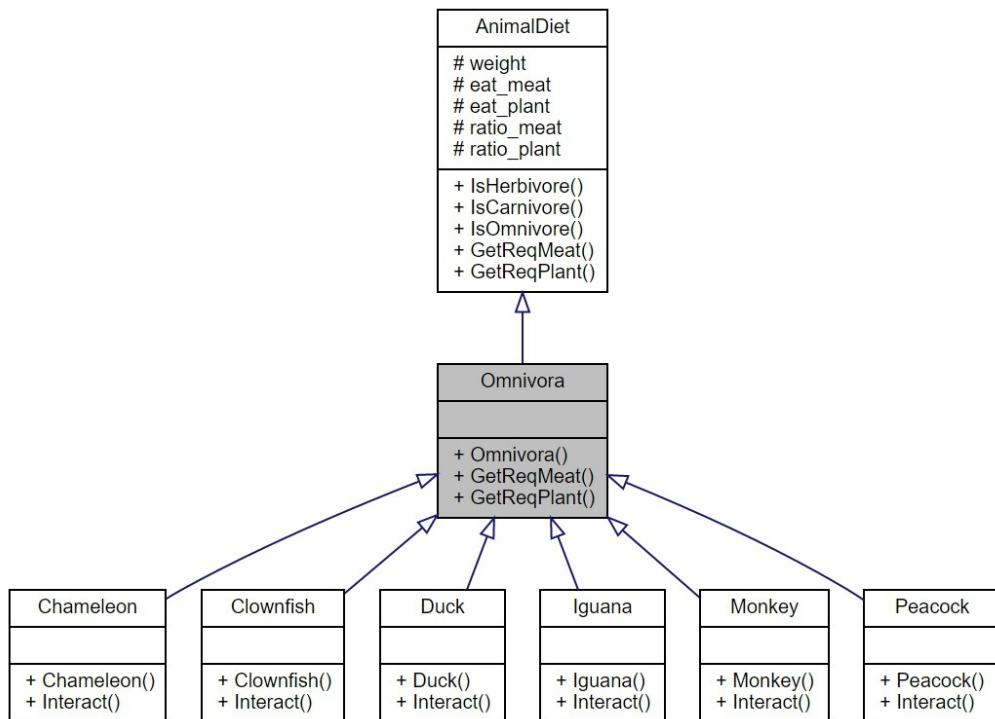
Gambar 28 UML Mammalia

2.29 Kelas Monkey



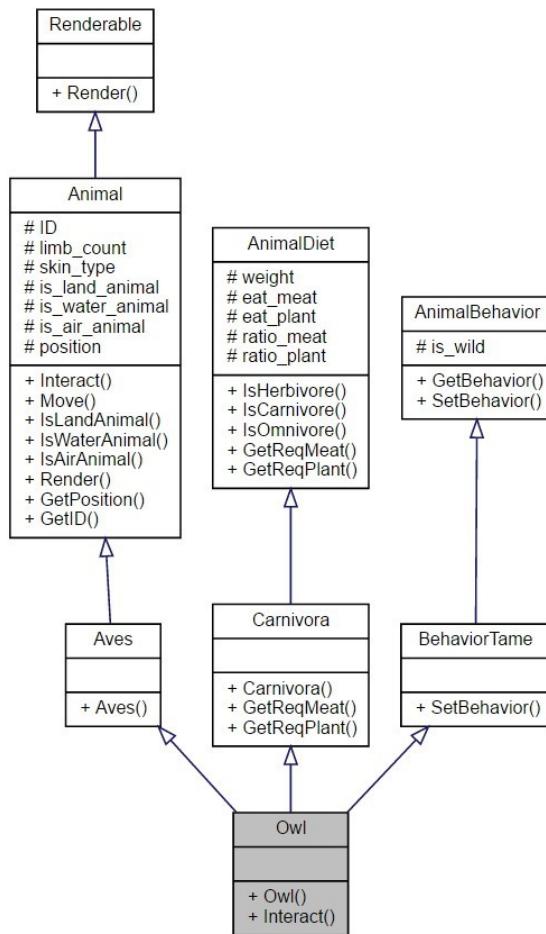
Gambar 29 UML Monkey

2.30 Kelas Omnivora



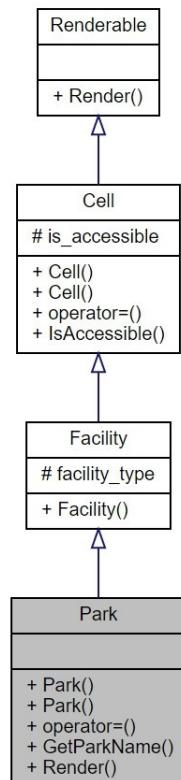
Gambar 30 UML Omnivora

2.31 Kelas Owl



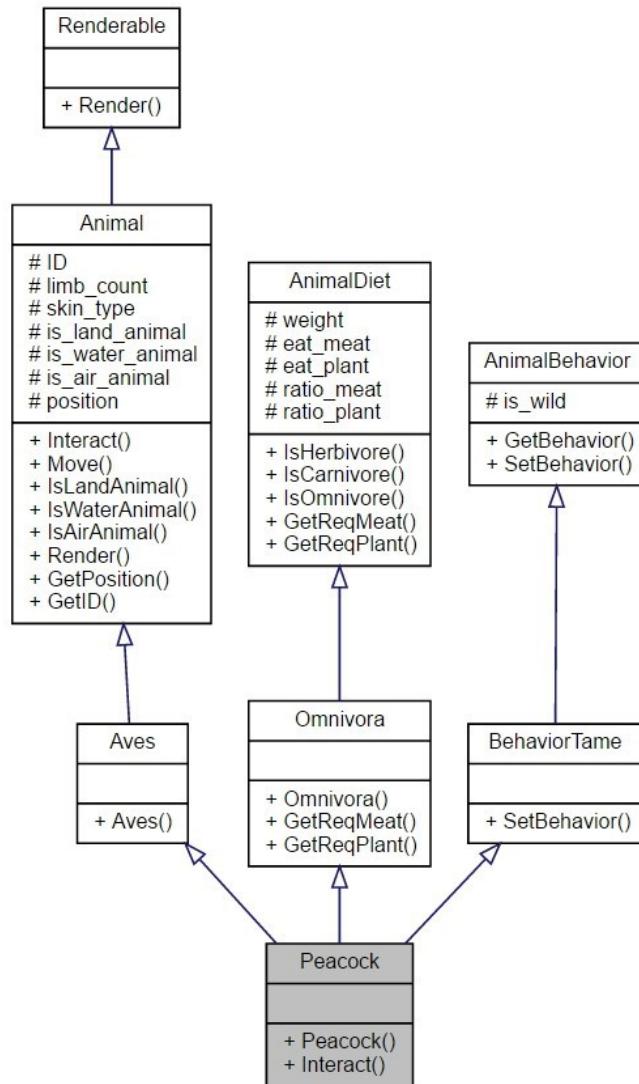
Gambar 31 UML Owl

2.32 Kelas Park



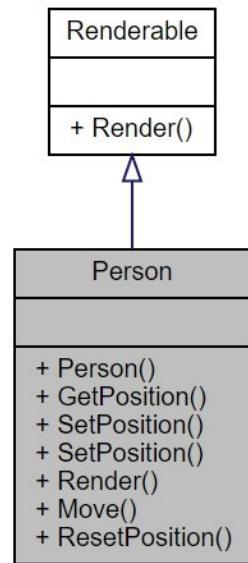
Gambar 32 UML Park

2.33 Kelas Peacock



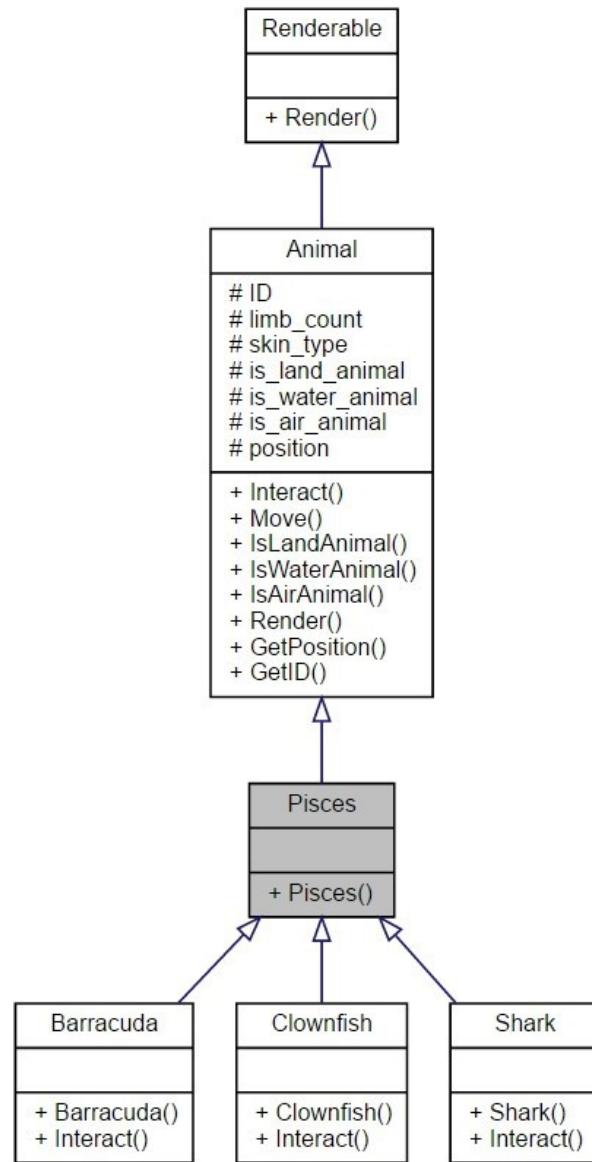
Gambar 33 UML Peacock

2.34 Kelas Person



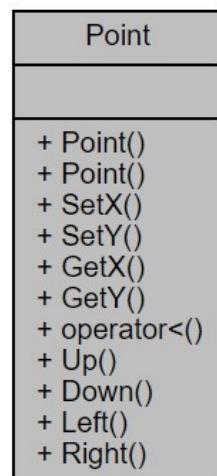
Gambar 34 UML Person

2.35 Kelas Pisces



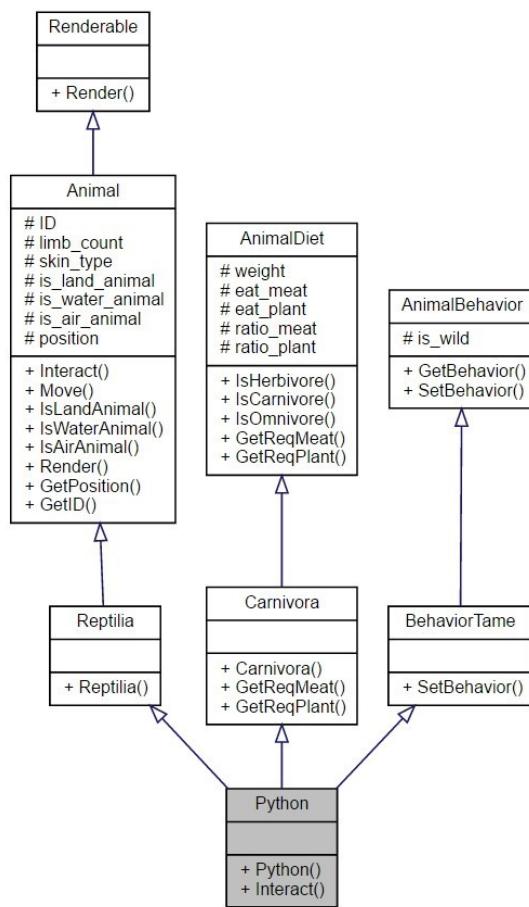
Gambar 35 UML Pisces

2.36 Kelas Point



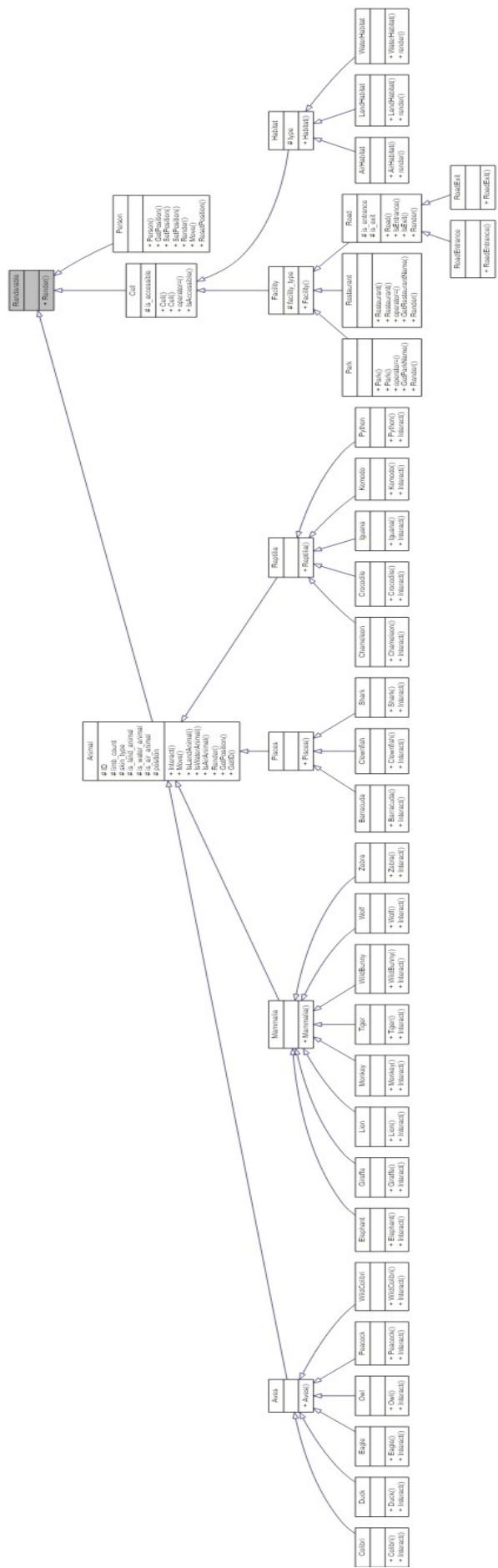
Gambar 36 UML Point

2.37 Kelas Python



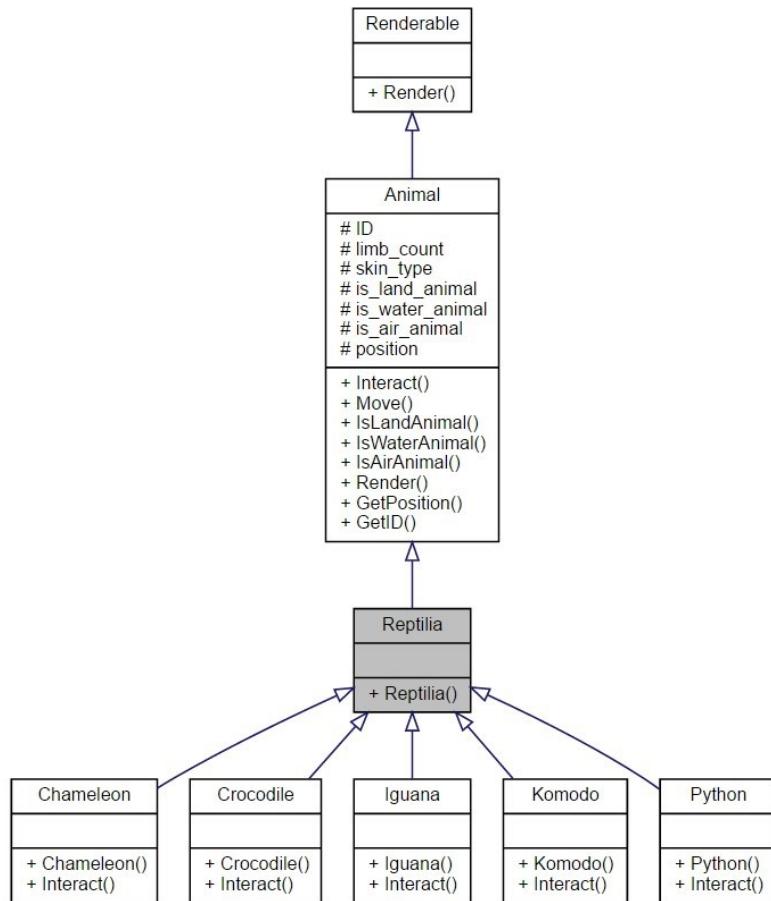
Gambar 37 UML Python

2.38 Kelas Renderable



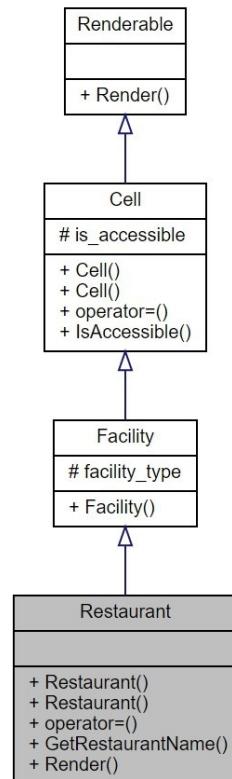
Gambar 38 UML Renderable

2.39 Kelas Reptilia



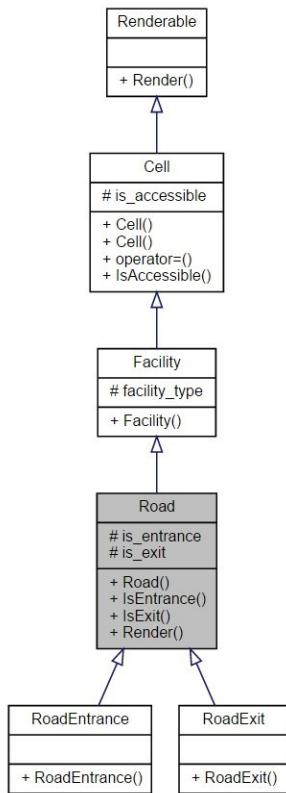
Gambar 39 UML Reptilia

2.40 Kelas Restaurant



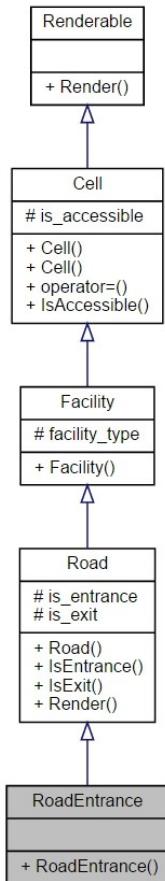
Gambar 40 UML Restaurant

2.41 Kelas Road



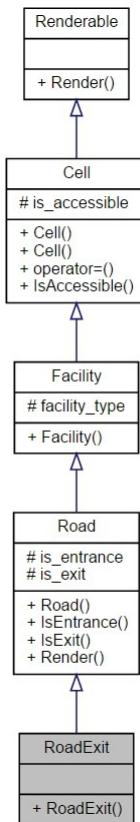
Gambar 41 UML Road

2.42 Kelas RoadEntrance



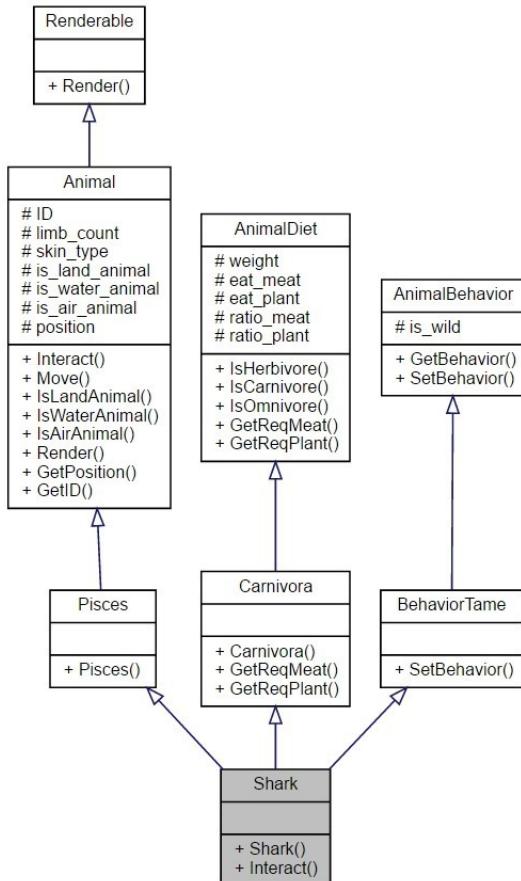
Gambar 42 UML RoadEntrance

2.43 Kelas RoadExit



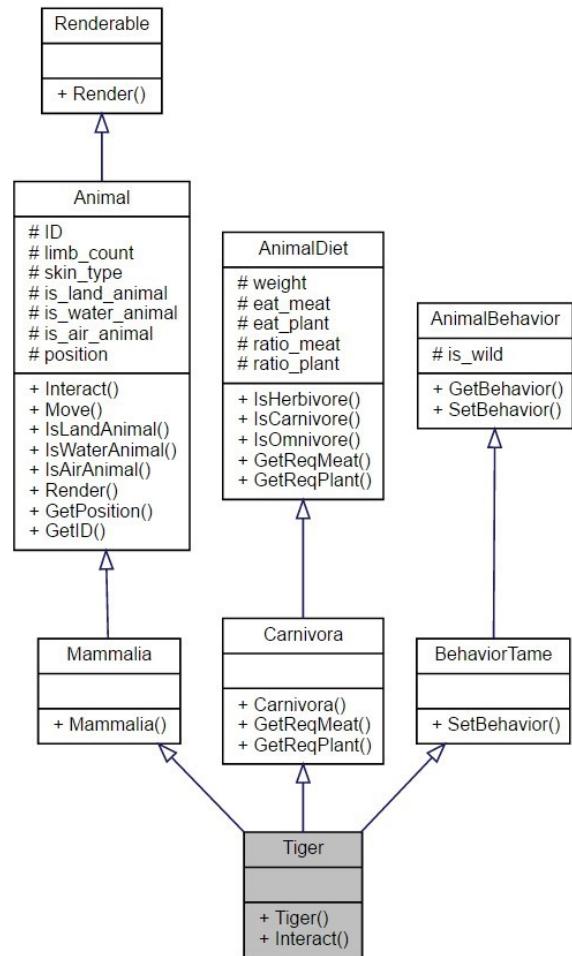
Gambar 43 UML RoadExit

2.44 Kelas Shark



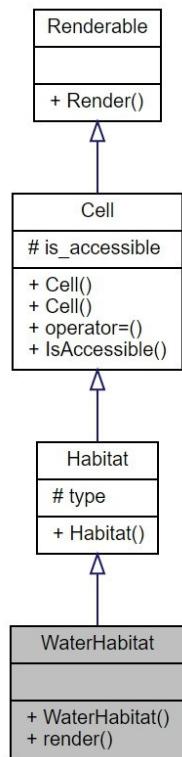
Gambar 44 UML Shark

2.45 Kelas Tiger



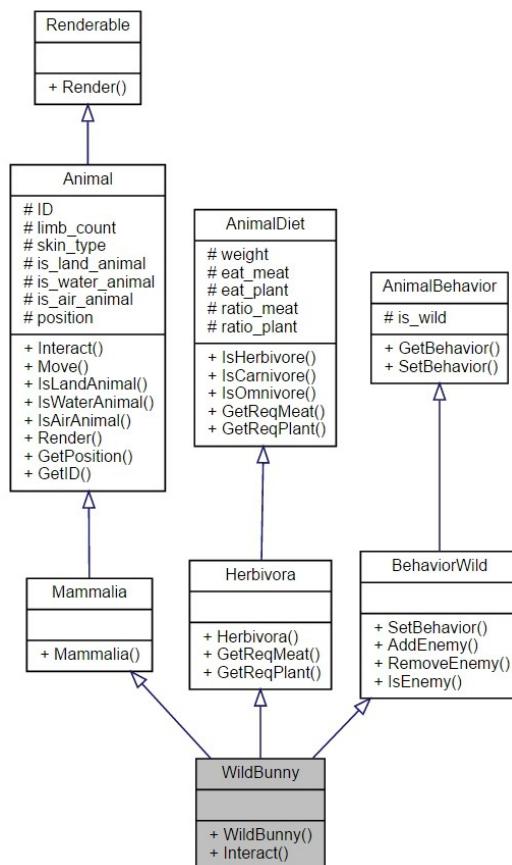
Gambar 45 UML Tiger

2.46 Kelas WaterHabitat



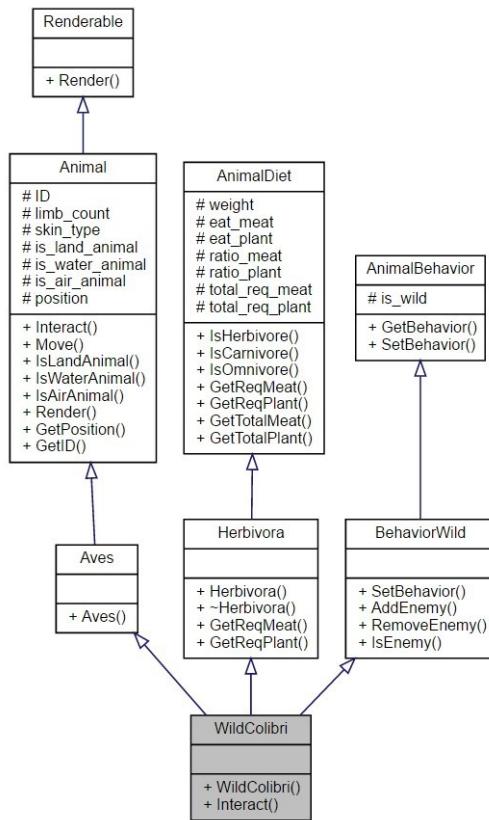
Gambar 46 WaterHabitat

2.47 Kelas WildBunny



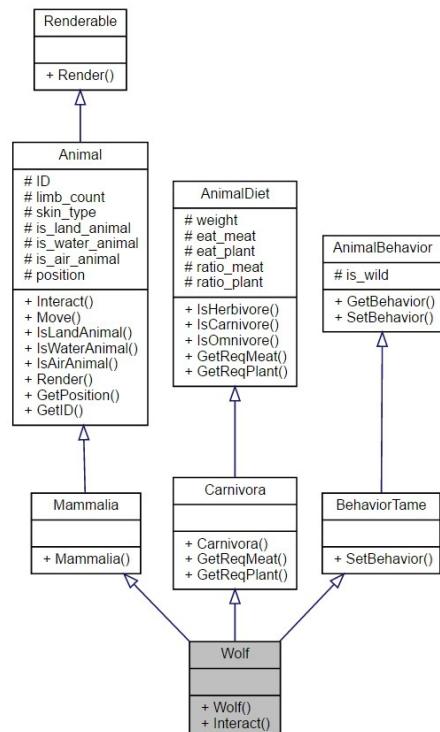
Gambar 47 UML WildBunny

2.48 Kelas WildColibri



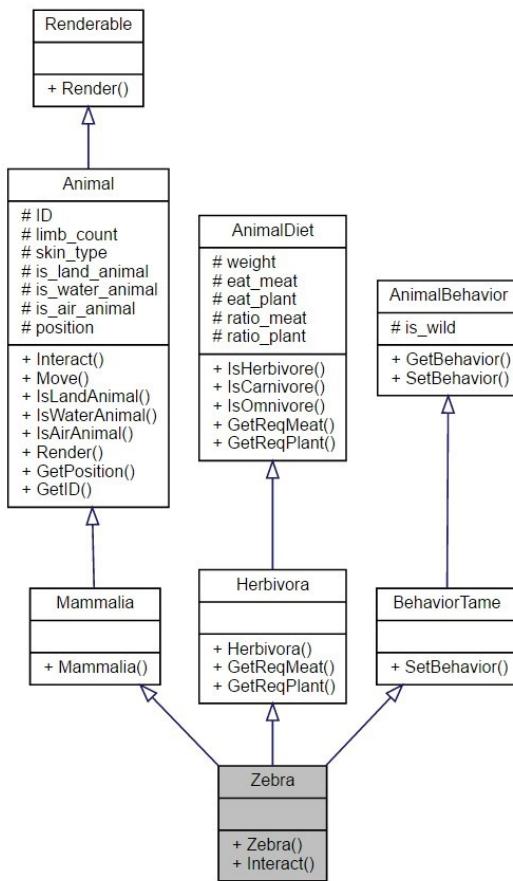
Gambar 48 UML WildColibri

2.49 Kelas Wolf



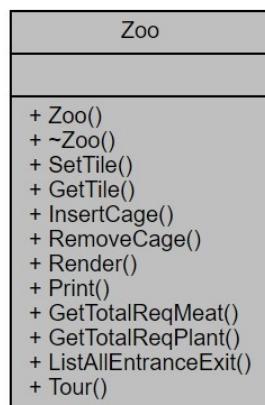
Gambar 49 UML Wolf

2.50 Kelas Zebra



Gambar 50 UML Zebra

2.51 Kelas Zoo



Gambar 51 UML Zoo

BAB III

ULASAN RANCANGAN

3.1 Rancangan Akhir

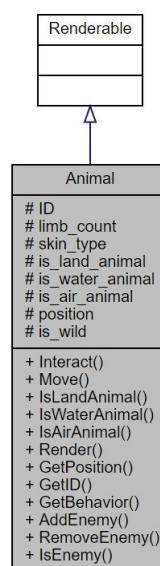
Dalam rancangan kelas yang telah dibuat, kelas Renderable adalah kelas yang diwarisi oleh semua kelas abstrak dan riil. Kelas Renderable mewarisi kelas Person, Animal, dan Cell, kelas tersebut mewarisi atribut berupa suatu lambang yang akan digunakan untuk menandai suatu objek merupakan Person, Animal, atau suatu Cell. Kelas abstrak Animal diwarisi *superclass* hewan yang terdiri dari Aves, Mammalia, Reptilia, dan Pisces.

Kelas riil hewan mewarisi atribut kelas abstrak *superclass*-nya. Atribut tersebut adalah ciri khas dari setiap *superclass*-nya. Selain itu, kelas riil hewan mewarisi kelas abstrak Behavior-nya yang menentukan apakah hewan tersebut jinak atau liar dengan atribut *is_wild* dan kelas abstrak berdasarkan penggolongan makanan hewan tersebut (Herbivora, Carnivora, atau Omnivora).

Kelas Cell adalah induk dari kelas Facility dan Habitat. Kelas habitat mewarisi atribut tipe habitat pada kelas turunannya (LandHabitat, WaterHabitat, dan AirHabitat) yang menjadi ciri kelas-kelas tersebut. Kelas Facility mewarisi atribut pada kelas turunannya (Park, Restaurant, dan Road), atribut tersebut adalah ciri pembeda untuk setiap kelas turunannya.

3.2 Rancangan Awal

Pada rancangan awal kelas, untuk membedakan apakah suatu kelas hewan jinak atau tidak maka digunakan suatu atribut. Sedangkan untuk versi terakhir, untuk membedakan suatu kelas jinak atau tidak, dibuat kelas abstrak Behavior. Kelas ini digunakan untuk menyimpan atribut pembeda hewan jinak dengan liar dan kumpulan hewan apa saja yang tidak dapat disatukan bersama dengan hewan tersebut. Kelas lainnya yang ditambahkan pada versi akhir adalah kelas Person. Kelas Person digunakan untuk mempermudah implementasi metode yang ada dalam kelas Zoo.

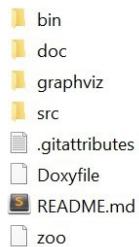


Gambar 52 UML Animal (rancangan awal)

BAB IV

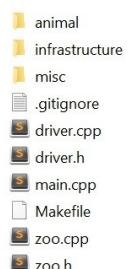
SNAPSHOT STRUKTUR DIREKTORI

4.1 Direktori Root



Gambar 53 Snapshot Root

4.1.1 Direktori src



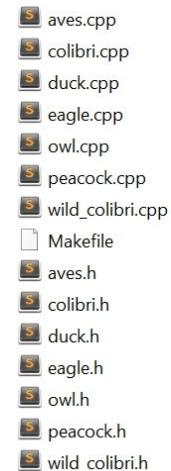
Gambar 54 Snapshot src

4.1.1.1 Direktori animal



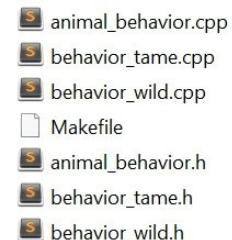
Gambar 55 Snapshot animal

4.1.1.1.1 Direktori aves



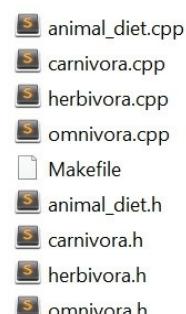
Gambar 56 Snapshot aves

4.1.1.1.2 Direktori behavior



Gambar 57 Snapshot behavior

4.1.1.1.3 Direktori diet



Gambar 58 Snapshot diet

4.1.1.1.4 Direktori mammalia

```
└── mammalia
    ├── elephant.cpp
    ├── giraffe.cpp
    ├── lion.cpp
    ├── mammalia.cpp
    ├── monkey.cpp
    ├── tiger.cpp
    ├── wild_bunny.cpp
    ├── wolf.cpp
    └── zebra.cpp

    └── Makefile
    ├── elephant.h
    ├── giraffe.h
    ├── lion.h
    ├── mammalia.h
    ├── monkey.h
    ├── tiger.h
    ├── wild_bunny.h
    ├── wolf.h
    └── zebra.h
```

Gambar 59 Snapshot mammalia

4.1.1.1.5 Direktori pisces

```
└── pisces
    ├── barracuda.cpp
    ├── clownfish.cpp
    ├── pisces.cpp
    ├── shark.cpp
    └── Makefile

    ├── barracuda.h
    ├── clownfish.h
    ├── pisces.h
    └── shark.h
```

Gambar 60 Snapshot pisces

4.1.1.1.6 Direktori reptilia

```
└── reptilia
    ├── chameleon.cpp
    ├── crocodile.cpp
    ├── iguana.cpp
    ├── komodo.cpp
    ├── python.cpp
    ├── reptilia.cpp
    └── Makefile

    ├── chameleon.h
    ├── crocodile.h
    ├── iguana.h
    ├── komodo.h
    ├── python.h
    └── reptilia.h
```

Gambar 61 Snapshot reptilian

4.1.1.2 Direktori infrastructure

```
└── infrastructure
    ├── facility
    ├── habitat
    ├── cage.cpp
    ├── cage.h
    ├── cell.cpp
    ├── cell.h
    └── Makefile
```

Gambar 62 Snapshot infrastructure

4.1.1.2.1 Direktori facility

```
└── facility
    ├── facility.cpp
    ├── facility.h
    └── Makefile

    ├── park.cpp
    ├── park.h
    ├── restaurant.cpp
    ├── restaurant.h
    ├── road.cpp
    ├── road.h
    ├── road_entrance.cpp
    ├── road_entrance.h
    ├── road_exit.cpp
    └── road_exit.h
```

Gambar 63 Snapshot facility

4.1.1.2.2 Direktori habitat

```
└── habitat
    ├── water_habitat.h
    ├── water_habitat.cpp
    └── Makefile

    ├── land_habitat.h
    ├── land_habitat.cpp
    ├── habitat.h
    ├── habitat.cpp
    ├── air_habitat.h
    └── air_habitat.cpp
```

Gambar 64 Snapshot habitat

4.1.1.3 Direktori misc

```
└── misc
    ├── Makefile
    ├── person.cpp
    ├── person.h
    ├── point.cpp
    ├── point.h
    └── renderable.h
```

Gambar 65 Snapshot misc

BAB V

LAMPIRAN

5.1 Skenario Test

5.1.1 Functional Test

Versi: 3

Skenario	Keterangan	Fakta	Nilai Mhs	Nilai Ass
General				
Inisiasi menu	Menampilkan menu awal Virtual Zoo	OK	A	
Pilih menu	Apabila memilih nomor 1, maka program akan menampilkan peta Virtual Zoo. Apabila memilih nomor 2, maka program akan mengeksekusi metode Tour, yaitu mengizinkan pengunjung untuk berjalan mengelilingi kebun binatang. Apabila memilih nomor 3, maka program akan menampilkan jumlah makanan yang dikonsumsi. Apabila memilih nomor 0, maka akan keluar dari program.	OK	A	
Display Virtual Zoo				
Menampilkan seluruh bagian peta	Besar peta maksimal 20 x 20 karakter, dimulai dari koordinat (0, 0) hingga (20, 20)	OK	A	
Menampilkan sebagian dari peta	Menentukan bagian peta yang ingin ditampilkan, dari koordinat (x1, y1) hingga (x2, y2), dengan besar x1, y1, x2, dan y2 lebih kecil dari 20.	OK	A	
Tour Virtual Zoo				
Pengunjung mengelilingi kebun binatang	Bergerak secara acak hingga menemukan pintu keluar	OK	A	
Pergerakan hewan	Berpindah tempat secara acak di dalam kandang	OK	A	
Jumlah Makanan Dikonsumsi				
Menampilkan konsumsi semua hewan dalam zoo setiap hari	Makanan dapat berbentuk daging dan tumbuhan	OK	A	
Bonus				
Retrieve dari file	Memuat peta dan informasi lainnya dari file eksternal bernama	OK	A	

Tabel 1 Functional Test

	“layout.in”			
--	-------------	--	--	--

5.1.2 Unit Test

Kelas / Keluarga kelas	Method	Kasus	OK/NO	Nilai
Animal	Interact	Pemanggilan metode	OK	
	IsLandAnimal	Pemanggilan metode	OK	
	IsWaterAnimal	Pemanggilan metode	OK	
	IsAirAnimal	Pemanggilan metode	OK	
	IsHerbivore	Pemanggilan metode	OK	
	IsCarnivore	Pemanggilan metode	OK	
	IsOmnivore	Pemanggilan metode	OK	
	GetReqMeat	Pemanggilan metode	OK	
	GetReqPlant	Pemanggilan metode	OK	
	GetPosition	Pemanggilan metode	OK	
	Move(1)	Pemanggilan metode	OK	
	Move(2)	Pemanggilan metode	OK	
	Move(3)	Pemanggilan metode	OK	
	Move(4)	Pemanggilan metode	OK	
	Render	Pemanggilan metode	OK	
	IsEnemy	Khusus hewan liar: parameter metode bukan musuh dari objek	OK	
	IsEnemy	Khusus hewan liar: parameter metode adalah musuh dari objek	OK	
	AddEnemy	Khusus hewan liar: pemanggilan metode	OK	
	RemoveEnemy	Khusus hewan liar: pemanggilan metode	OK	
Park	IsAccessible	Pemanggilan metode	OK	
	GetParkName	Pemanggilan metode	OK	
	Render	Pemanggilan metode	OK	
Restaurant	IsAccessible	Pemanggilan metode	OK	
	GetRestaurantName	Pemanggilan metode	OK	

Kelas / Keluarga kelas	Method	Kasus	OK/NO	Nilai
	Render	Pemanggilan metode	OK	
Road	IsAccessible	Pemanggilan metode	OK	
	IsEntrance	Pemanggilan metode	OK	
	IsExit	Pemanggilan metode	OK	
	Render	Pemanggilan metode	OK	
RoadEntrance	IsAccessible	Pemanggilan metode	OK	
	IsEntrance	Pemanggilan metode	OK	
	IsExit	Pemanggilan metode	OK	
	Render	Pemanggilan metode	OK	
RoadExit	IsAccessible	Pemanggilan metode	OK	
	IsEntrance	Pemanggilan metode	OK	
	IsExit	Pemanggilan metode	OK	
	Render	Pemanggilan metode	OK	
AirHabitat	Render	Pemanggilan metode	OK	
LandHabitat	Render	Pemanggilan metode	OK	
WaterHabitat	Render	Pemanggilan metode	OK	

Tabel 2 Unit Test

5.2 Penghitungan Metriks Perangkat Lunak

No	Metriks	Besarnya	Keterangan
1.	Number of packages	51	Package adalah sub direktori anda
2.	Number of classes	40	Jumlah kelas riil
3.	Number of abstract classes	11	Interface (pada C++)
4.	Afferent Couplings (Ca)	7	Rata-rata jumlah kelas dalam package lain yang memiliki ketergantungan terhadap package yang dihitung
5.	Efferent Couplings (Ce)	3	Rata-rata jumlah kelas dalam package lain yang dimanfaatkan oleh package yang dihitung
6.	Abstractness (A)	11:40	Rasio jumlah kelas abstract dibandingkan dengan kelas riil
7.	Instability (I)	0.3	$I = Ce / (Ce + Ca)$
8.	Package Dependency Cycle	Tidak	Package yang bergantung satu sama lain

9.	Kelas generik	0	Tidak ada
----	---------------	---	-----------

Tabel 3 Penilaian Metriks Perangkat Lunak

5.3 Log Activity

5.3.1 Pembagian peran

5.3.1.1 Deskripsi umum

Secara garis besar, desainer kelas hewan berbentuk taksonomi adalah penulis dengan NIM 13515035, 13515077, dan 13515113. Sedangkan, perilaku hewan dirancang oleh 13515113, lalu preferensi makanan diatur oleh 13515077 dan 13515113. Infrastruktur yang terdiri dari kelas Facility dan habitat, secara garis besar didesain oleh keempat penulis. Kelas Cage, Person, Driver, Renderable, dan Zoo dirancang oleh 13515065. 13515077 juga merancang kelas titik. Keempat penulis juga dapat berlaku sebagai implementator dan tester.

5.3.1.2 Deskripsi rinci

Kelas	File	Designer	Implementator	Tester
AirHabitat	air_habitat.h air_habitat.cpp	13515035 13515065 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Animal	animal.h animal.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113
AnimalBehavior	animal_behavior.h animal_behavior.cpp	13515113	13515113	13515035 13515065 13515077 13515113
AnimalDiet	animal_diet.h animal_diet.cpp	13515077 13515113	13515077 13515113	13515035 13515065 13515077 13515113
Aves	aves.h aves.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Barracuda	barracuda.h barracuda.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
BehaviorTame	behavior_tame.h behavior_tame.cpp	13515113	13515077	13515035 13515065 13515077

Kelas	File	Designer	Implementator	Tester
				13515113
BehaviorWild	behavior_wild.h behavior_wild.cpp	13515113	13515035	13515035 13515065 13515077 13515113
Cage	cage.h cage.cpp	13515065	13515065	13515035 13515065 13515077 13515113
Carnivora	carnivora.h carnivora.cpp	13515077 13515113	13515035	13515035 13515065 13515077 13515113
Cell	cell.h cell.cpp	13515035 13515065 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Chameleon	chameleon.h chameleon.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Clownfish	clownfish.h clownfish.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Colibri	colibri.h colibri.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Crocodile	crocodile.h crocodile.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Driver	driver.h driver.cpp	13515065	13515065	13515035 13515065 13515077 13515113
Duck	duck.h duck.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Eagle	eagle.h eagle.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113

Kelas	File	Designer	Implementator	Tester
Elephant	elephant.h elephant.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Facility	facility.h facility.cpp	13515035 13515065 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Giraffe	giraffe.h giraffe.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Habitat	habitat.h habitat.cpp	13515035 13515065 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Herbivora	herbivora.h herbivora.cpp	13515077 13515113	13515077	13515035 13515065 13515077 13515113
Iguana	iguana.h iguana.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Komodo	komodo.h komodo.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
LandHabitat	land_habitat.h land_habitat.cpp	13515035 13515065 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Lion	lion.h lion.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Mammalia	mammalia.h mammalia.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Monkey	monkey.h monkey.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Omnivora	omnivora.h omnivora.cpp	13515077 13515113	13515113	13515035 13515065

Kelas	File	Designer	Implementator	Tester
				13515077 13515113
Owl	owl.h owl.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Park	park.h park.cpp	13515035 13515065 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Peacock	peacock.h peacock.cpp	13515035 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Person	person.h person.cpp	13515065	13515065	13515035 13515065 13515077 13515113
Pisces	pisces.h pisces.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Point	point.h point.cpp	13515077	13515077	13515035 13515065 13515077 13515113
Python	python.h python.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Renderable	renderable.h renderable.cpp	13515065	13515065	13515035 13515065 13515077 13515113
Reptilia	reptilia.h reptilia.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Restaurant	restaurant.h restaurant.cpp	13515035 13515065 13515077 13515113	13515035	13515035 13515065 13515077 13515113
Road	road.h road.cpp	13515035 13515065 13515077 13515113	13515113	13515035 13515065 13515077 13515113

Kelas	File	Designer	Implementator	Tester
RoadEntrance	road_entrance.h road_entrance.cpp	13515113	13515035	13515035 13515065 13515077 13515113
RoadExit	road_exit.h road_exit.cpp	13515113	13515035	13515035 13515065 13515077 13515113
Shark	shark.h shark.cpp	13515035 13515077 13515113	13515077	13515035 13515065 13515077 13515113
Tiger	tiger.h tiger.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
WaterHabitat	water_habitat.h water_habitat.cpp	13515035 13515065 13515077 13515113	13515077	13515035 13515065 13515077 13515113
WildBunny	wild_bunny.h wild_bunny.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
WildColibri	wild_colibri.h wild_colibri.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Wolf	wolf.h wolf.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Zebra	zebra.h zebra.cpp	13515035 13515077 13515113	13515113	13515035 13515065 13515077 13515113
Zoo	zoo.h zoo.cpp	13515065	13515065	13515035 13515065 13515077 13515113

5.3.1.3 Rincian kegiatan

No	Dari Tgl... Pk...	S.d. Tgl... Pk...	Kegiatan	Hasil	Keterangan
1	20/02/17	20/02/17	Perancangan awal	Kelas VZ01	

	pukul 19.00	pukul 21.00			
2	24/02/17 pukul 14.30	24/02/17 pukul 17.30	Modifikasi rancangan kelas sesuai dengan Deskripsi Tugas Besar versi 2.0	Rancangan kelas yang telah diperbarui	Sesuai dengan Deskripsi Tugas Besar versi 2.0
3	11/03/17 pukul 12.00	11/03/17 pukul 22.00	Modifikasi rancangan kelas hewan dan infrastruktur Melakukan implementasi pada kelas-kelas yang telah dibuat	Rancangan kelas dan implementasinya (VZ03)	Sesuai dengan Change Request versi 3
4	12/03/17 pukul 10.00	12/03/17 pukul 23.00	Melakukan implementasi pada kelas-kelas yang dibuat untuk VZ02 dan VZ03	Rancangan kelas dan implementasi (VZ03 dan VZ02) Dokumentasi softcopy (Doxyfile)	
5	13/03/17 pukul 15.30	13/03/17 pukul 23.00	Melakukan implementasi dan debugging pada kelas-kelas yang dibuat untuk VZ02 dan VZ03	Rancangan kelas dan implementasi (VZ03 dan VZ02) Makefile Dokumentasi Utama VZ03	
6	14/03/17 pukul 18.00	15/03/17 pukul 02.00	Finishing, debugging, dan dokumentasi	Dokumentasi softcopy dan hardcopy	