

VIRTUAL ZOO (VZ.02)

Diajukan untuk memenuhi Tugas Besar I IF2210 Pemrograman Berorientasi Objek

.....	3
.....	4
.....	5
.....	6
2.1 Kelas Animal	6
2.2 Kelas AnimalFood	6
2.3 Kelas Cage	6
2.4 Kelas Cell.....	7
2.5 Kelas Driver	7
2.6 Kelas Person.....	7
2.7 Kelas Point.....	7
2.8 Kelas Zoo	8
.....	9
3.1 Rancangan Akhir	9
3.2 Rancangan Awal	9
.....	10
4.1 Direktori Root	10
4.1.1 Direktori src	10
.....	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

Gambar 1 UML Animal.....	6
Gambar 2 UML AnimalFood.....	6
Gambar 3 UML Cage.....	6
Gambar 4 UML Cell	7
Gambar 5 UML Driver	7
Gambar 6 UML Person.....	7
Gambar 7 UML Point	7
Gambar 8 UML Zoo	8
Gambar 9 Snapshot Root	10
Gambar 10 Snapshot src	10
Gambar 11 Snapshot animal	10
Gambar 12 Snapshot infrastructure.....	10
Gambar 13 Snapshot misc.....	10

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

Virtual Zoo adalah kebun binatang yang terdiri dari sebidang tanah dengan luas 20 x 20 karakter. Pengunjung dapat melakukan 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.

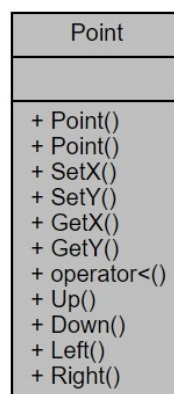
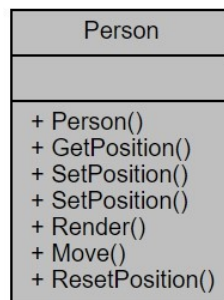
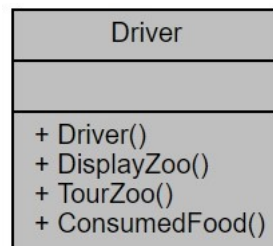
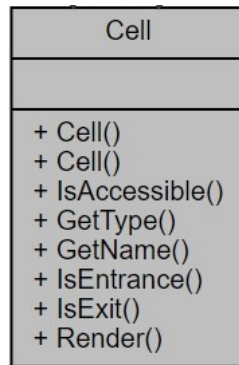
Setiap jenis hewan dapat diklasifikasikan berdasarkan tipe makanan (karnivora / herbivora / omnivora), taksonomi (burung / mamalia / reptil / ikan), tempat tinggal (air / darat / udara), dan perilaku (liar / jinak). Makanan, baik yang berwujud daging ataupun tumbuhan, yang dibutuhkan setiap hewan dapat 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.

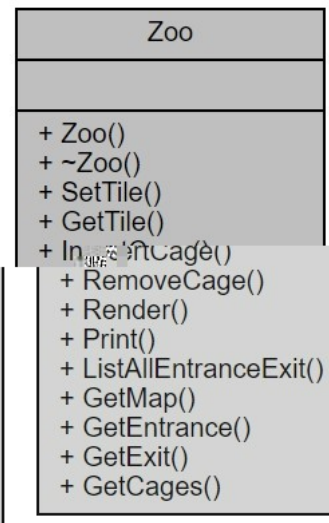
Infrastruktur bangunan bisa berbentuk fasilitas dan habitat. Pengunjung dapat mengakses fasilitas tertentu, tetapi habitat pasti tidak dapat dilewati. Fasilitas bisa berupa taman, restoran, dan jalan. Jalan dapat diklasifikasi sebagai jalan biasa, jalan masuk, dan jalan keluar. Sedangkan, habitat dapat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

Animal
+ Animal() + Interact() + Move() + IsLandAnimal() + IsWaterAnimal() + IsAirAnimal() + IsHerbivore() + IsCarnivore() + IsOmnivore() + GetReqMeat() and 11 more...

AnimalFood
+ AddTotalMeat() + AddTotalPlant() + GetTotalMeat() + GetTotalPlant()

Cage
+ Cage() + AddPoint() + RemovePoint() + AddAnimal() + RemoveAnimal() + MoveAnimal() + GetArea() + GetAnimal()

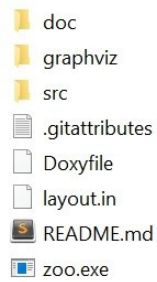




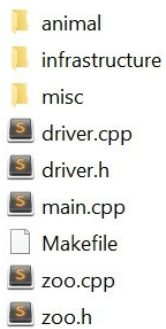
Dalam rancangan kelas VZ02 yang telah dibuat, setiap kelas riil hewan yang ada pada VZ03 diwakilkan dengan satu kelas bernama *Animal* yang memiliki atribut-atribut yang dapat membedakan antar hewan. Pada VZ03, ciri khas tersebut sebagian besar diwariskan dari kelas abstrak. Sedangkan pada VZ02, ciri khas tersebut diberikan oleh atribut-atribut yang ada.

Kelas *Cell* juga mewakili fasilitas (seperti *Park*, *Restaurant*, *Road*, *RoadEntrance*, dan *RoadExit*) dan habitat (seperti *AirHabitat*, *WaterHabitat*, dan *LandHabitat*). Kelas *Cage*, *Driver*, *Person*, *Point*, dan *Zoo* tidak berbeda jauh dengan versi VZ03. Kelas *Renderable* juga ditiadakan karena dapat diintegrasikan ke dalam kelas *Animal* dan *Cell*.

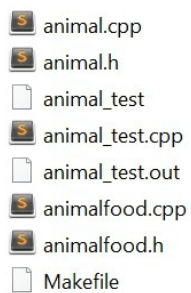
Pada rancangan awal kelas, setiap jenis hewan memiliki kelas riilnya masing-masing. *Road*, *RoadEntrance*, *RoadExit*, *Park*, *Restaurant*, *AirHabitat*, *WaterHabitat*, dan *LandHabitat* juga berdiri sebagai kelas sendiri, tidak tergabung dalam kelas *Cell* seperti pada rancangan akhir. Perubahan dari rancangan awal ke rancangan akhir dilakukan untuk efisiensi penggunaan kelas, terutama karena banyak atribut yang sama digunakan dalam beberapa kelas yang berbeda.



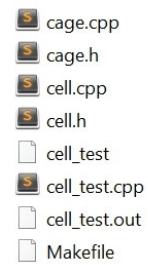
4.1.1 Direktori src



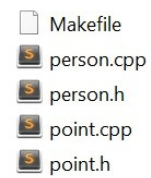
4.1.1.1 Direktori animal



4.1.1.2 Direktori infrastructure



4.1.1.3 Direktori misc



5.1.1 Functional Test

Versi: 2

Inisiasi menu	Menampilkan menu awal Virtual Zoo	OK	A	
Pilih menu	<p>Apabila memilih nomor 1, maka program akan menampilkan peta Virtual Zoo.</p> <p>Apabila memilih nomor 2, maka program akan mengeksekusi metode Tour, yaitu mengizinkan pengunjung untuk berjalan mengelilingi kebun binatang.</p> <p>Apabila memilih nomor 3, maka program akan menampilkan jumlah makanan yang dikonsumsi.</p> <p>Apabila memilih nomor 0, maka akan keluar dari program.</p>	OK	A	
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	
Pengunjung mengelilingi kebun binatang	Bergerak secara acak hingga menemukan pintu keluar	OK	A	
Pergerakan hewan	Berpindah tempat secara acak di dalam kandang	OK	A	
Menampilkan konsumsi semua hewan dalam zoo setiap hari	Makanan dapat berbentuk daging dan tumbuhan	OK	A	
Retrieve dari file	Memuat peta dan informasi lainnya dari file eksternal bernama	OK	A	

	"layout.in"			
--	-------------	--	--	--

5.1.2 Unit Test

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	
	GetID	Pemanggilan metode	OK	
	GetBehavior	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	
	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	

1.	Number of packages	4	Package adalah sub direktori anda
2.	Number of classes	8	Jumlah kelas riil
3.	Number of abstract classes	0	Interface (pada C++)
4.	Afferent Couplings (Ca)	6	Rata-rata jumlah kelas dalam package lain yang memiliki ketergantungan terhadap package yang dihitung
5.	Efferent Couplings (Ce)	8	Rata-rata jumlah kelas dalam package lain yang dimanfaatkan oleh package yang dihitung
6.	Abstractness (A)	0	Rasio jumlah kelas abstract dibandingkan dengan kelas riil

7.	Instability (I)	0.57	$I = C_e / (C_e + C_a)$
8.	Package Dependency Cycle	Tidak	Package yang bergantung satu sama lain
9.	Kelas generik	0	Tidak ada