Dokumentacja projektu z Programowania Obiektowego

Roślinożercy i drapieżniki

Szymon Ciszewski (252730)

Mikołaj Rarok (256306)

https://github.com/Sakue34/CarnivoreHerbivore

Spis treści

Spis treści	
1. Opis zadania symulacji w języku naturalnym	
2. Analiza czasownikowo-rzeczownikowa	
3. Karty CRC	
Pakiet "config"	
Pakiet "map"	
Pakiet "output"	
Pakiet "pawn"	
Pakiet "simulation"	
Pakiet "utility"	
Klasa "App"	
4. Diagram przypadków użycia	
5. Diagram klas	
Pakiet "config"	
Pakiet "map"	
Pakiet "output"	
Pakiet "pawn"	
Pakiet "simulation"	
Pakiet "utility"	16
Klasa "App"	16
6. Diagram obiektów	
7. Diagramy aktywności	

1. Opis zadania symulacji w języku naturalnym.

Roślinożercy i drapieżniki (CarnivoreHerbivore) to prosta symulacja agentowa z wykorzystaniem technik programowania obiektowego.

Program symuluje zachowania dwóch prostych gatunków zwierząt. Roślinożercy poruszają się po mapie, starając się zjeść każdą roślinę w ich zasięgu, oraz uciekają przed drapieżnikami. Drapieżnicy natomiast polują na najbliższego roślinożercę w ich zasięgu wzroku. Obydwa gatunki przechodzą w stan znudzenia jeśli nie widzą pożywienia ani zagrożenia. Poruszają się one wtedy w losowym kierunku, zmieniając go co stały okres.

Rośliny są trzecim rodzajem obiektów, nie poruszają się i stanowią pożywienie dla roślinożerców.

Symulacja rozgrywa się na ograniczonej mapie, poza którą nie może wyjść żaden obiekt. Początkowa pozycja wszystkich obiektów jest losowana w granicach mapy przed rozpoczęciem symulacji.

Wszystkie parametry symulacji, włącznie z jej warunkami początkowymi są wczytywane przy uruchomieniu programu z pliku konfiguracyjnego.

2. Analiza czasownikowo-rzeczownikowa

Na planszy znajdują się trzy rodzaje obiektów: roślinożercy, drapieżnicy oraz rośliny.

Wszystkie obiekty pojawiają się w sposób losowy na początku symulacji, roślinożercy zjadają rośliny oraz uciekają przed drapieżnikami, a drapieżnicy polują na roślinożerców.

Można dostosowywać warunki początkowe symulacji w pliku konfiguracyjnym, początkową ilość roślinożerców i drapieżników.

3. Karty CRC

Pakiet "config"

ConfigProvider		File	ConfigProvider	
Super Classes:		Su	per Classes:	
Sub Classes: DefaultParametersProvider, FileC	onfigProvider	Sul	o Classes:	
Description: Interfejs unifikujący klasy dostarcz	ające parametry symulacji.	De	scription: Klasa dostarczająca początkowe p	arametry symulacji pobierając je z pliku na dysku.
Attributes:		Attr	ibutes:	
Name	Description		Name	Description
Responsibilities:		Res	sponsibilities:	
Name	Collaborator		Name	Collaborator
Unifikowanie klas dostarczających parametrów		Pot	pieranie parametrów symulacji z pliku	
		//		

DefaultParamtersProvider	
Super Classes: ConfigProvider	
Sub Classes:	
Description: Klasa dostarczająca początkowe parametry sy gdy z różnych przyczyn nie udało się odczyta	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
Dostarczanie domyślnych paramterów symulacji	
	//

Pakiet "map"

MapCreator		SimulationMap	
Super Classes:		Super Classes:	
Sub Classes:		Sub Classes:	
	a mapę na której odbywa się symulacja ją początkową liczbą pionków.	Description: Klasa reprezentując symulacja.	a mapę na której odbywa sie
Attributes:		Attributes:	
Name	Description	Name	Description
Responsibilities:	·	pawns	Lista obiektów
		pawnsPosition	Lista pozycji obiektów
Name	Collaborator		
Tworzenie mapy	SimulationMap	Responsibilities:	
		Name	Collaborator
		Przechowywanie obiektów i ich pozycji	Pawn, Position

Pakiet "output"

ProgressOutput		SimpleProgressOutput		TerminalProgressLogger		
Super Classes:		Super Classes: ProgressOutp	ut	Super Classes: SimpleProgressOutput		
Sub Classes: SimpleProgressC	utput	Sub Classes: GUIProgressOu	tput, TerminalProgressLogger	Sub Classes:		
Description: Intersejs unifikująd symulacji.	cy klasy wyświetlające przebieg	Description: Klasa bazowa dla wyświetlania prze		Description: Klasa wyświetlaja pomocą wyjścia		
Attributes:		Attributes:		Attributes:		
Name	Description	Name	Description	Name	Description	
Responsibilities:		Responsibilities:		Responsibilities:		
Name	Collaborator	Name	Collaborator	Name	Collaborator	
Interfejs dla wszystkich metod wyświetlania przebiegu symulacji		Wyświetlanie najbardziej podstawowych informacji o postępie symulacji w konsoli	Simulation	Wyświetlanie przebiegu symulacji w konsoli	Simulation	
symans,		Wyświetlanie podsumowania symulacji w konsoli	SimulationResults		//.	
		GUIProgressOutput		/// SimulationResults		
		Super Classes: SimpleProgr	essOutput	Super Classes:		
		Sub Classes:		Sub Classes:		
		Description: Klasa wyświetlająca przebieg symulacji za pomocą interfejsu graficznego.		Description: Pomocniczna kla		
		pomocą interfej		wyświetlające po	isa tworzona przez klasy stęp po spełnieniu warunków pulacii	
				wyświetlające po zakończenia syn	stęp po spełnieniu warunków	
		pomocą interfej		wyświetlające po zakończenia syn Attributes:	stęp po spełnieniu warunków	
		pomocą interfej Attributes:	su graficznego.	wyświetlające po zakończenia syn Attributes: Name	stęp po spełnieniu warunków	
		pomocą interfej Attributes: Name	su graficznego.	wyświetlające po zakończenia syn Attributes:	stęp po spełnieniu warunków nulacji.	
		pomocą interfej Attributes: Name Responsibilities:	su graficznego. Description	wyświetlające po zakończenia syn Attributes: Name	stęp po spełnieniu warunków nulacji.	

Pakiet "pawn"

Pawn		Plant		Animal	
Super Classes:		Super Classes: Pawn		Super Classes: Pawn	
Sub Classes: Animal, Plant		Sub Classes:		Sub Classes: Carnivore, Herbivore	
Description: Klasa abstrakcyjna unifikuja	ąca wszystkie obiekty (pionki) na mapie.	Description: Klasa reprezentuja	ica roślinę.	Description: Klasa abstrakcyjna u które się poruszają.	nifikująca "zwierzęta", czyli pionki
Attributes:		Attributes:		Attributes:	
Name	Description	Name	Description	Name	Description
energy	Energia życiowa (lub wartość odżywcza)	Responsibilities:		energyConsumptionPerSecond	Ilość energii zużywanej w ciągu
radius	Promień obiektu				sekundy ruchu
velocity	Prędkość	Name	Collaborator	Responsibilities:	
colourString	Kolor	Klasa reprezentująca roślinę w	SimulationLogic	N.I. and a	Callabaratas
Responsibilities:		logice symulacji		Name Name	Collaborator
				Implementacja znudzenia zwierząt Obsługa zużywania energii	
Name	Collaborator			życiowej	
Ujednolicenie obiektów na mapie	SimulationMap			2,515115,	
	l.				ſį.

Herbivore		Carnivore		
Super Classes: Animal		Super Classes: Animal		
Sub Classes:		Sub Classes:		
Description: Klasa reprezentująca	a roślinożercę.	Description: Klasa reprezentuja	ąca drapieżnika	
Attributes:		Attributes:		
Name	Description	Name	Description	
Responsibilities:		Responsibilities:		
Name	Collaborator	Name	Collaborator	
Klasa reprezentująca roślinożercę w logice symulacji	SimulationLogic	Klasa reprezentująca drapieżnika w logice symulacji	SimulationLogic	
	h.			//

Pakiet "simulation"

Simulation		SimulationCreator		SimulationParameters		
Super Classes:		Super Classes:		Super Classes:		
oupor olaboro.		ouper classes.		Super Classes.		
0.1.01		0.1.01		0.1.01		
Sub Classes:		Sub Classes:		Sub Classes:		
Description: Główna klasa sym	nulacji.	Description: Klasa tworząca o	biekt symulacji.	Description: Klasa przechowująca parametry symula	асјі.	
Attributes:		Attributes:		Attributes:		
Name	Description	Name	Description	Name	Description	
Responsibilities:		Responsibilities:	'	mapWidth	Szerokość mapy	
				mapHeight	Wysokość mapy	
Name	Collaborator	Name	Collaborator	speedOfSimulationMultiplier	Mnożnik prędkości symulacji	
Pętla główna symulacji	SimulationLogic, ProgressOutput,	Tworzenie obiektu mapy	MapCreator	startingNumberOfPlants	Początkowa liczba roślin	
	SimulationParameters	Tworzenie obiektu symulacji	Simulation	startingNumberOfHerbivores	Początkowa liczba roślinożerców	
			1	startingNumberOfCarnivores	Początkowa liczba drapieżników	
				herbivoreSpeed	Szybkość roślinożerców	
				carnivoreSpeed	Szybkość drapieżników	
NearestPawnFinder		SimulationLogic		plantNutritionalValue	Wartość odżywcza roślin	
Super Classes:		Super Classes:		herbivoreStartingEnergy	Początkowa energia roślinożerców	
·		Super Classes.		carnivoreStartingEnergy	Początkowa energia drapieżników	
				herbivoreSightRange	Zasięg wzroku roślinożerców	
Sub Classes:		Sub Classes:		carnivoreSightRange	Zasięg wzroku drapieżników	
				baseHerbivoreNutritionalValue	Bazowa wartość odżywcza roślinożerców	
				timeToSetNewRandomHerbivoreWanderDirection	Częstość zmiany kierunku znudzonych roślinożerców	
	na do wyznaczania najbliższego	Description: Klasa obsługująca całą logikę symulacji.		timeToSetNewRandomCarnivoreWanderDirection	Częstość zmiany kierunku znudzonych drapieżników	
pionka danego ty	pu.			useGUI	Czy używać interfejsu graficznego	
Attributes:		Attributes:		timeBetweenProgressOutputInTerminal	Częstośc wyświetlania postępu symulacji w terminalu	
Name	Description	Name	Description	numberOfPawnsToEndSimulation	Liczba obiektów na mapie potrzebna do zakończenia symulacji	
Responsibilities:		Responsibilities:	Босопраст	endSimulationWhenNoHerbivoresLeft	Czy kończyć symulację gdy nie zostaje żaden roślinożerca	
Name	Collaborator	Name	Collaborator	endSimulationWhenNoCarnivoresLeft	Czy kończyć symulację gdy nie zostaje żaden	
Wyszukiwanie najbliższych	SimulationLogic, Posiiton	Obsługa logiki symulacji	SimulationMap, Pawn, Position,	a alliai an Danga Multinliar Of Court Of Dadii	drapieżnik	
obiektów			NearestPawnFinder	collisionRangeMultiplierOfSumOfRadii	Mnożnik granicznego dystansu zachodzenia kolizji między obiektami	
	//		//	Responsibilities:		
				Name	Collaborator	
				Przechowywanie parametrów symulacji w jednym		
				obiekcie		

Pakiet "utility"

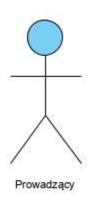
Position	
Super Classes:	
Sub Classes:	
our siassos.	
Description : Klasa przechowuja dostarczająca kilk	ąca pozycję pionków i u metod pomocniczych.
Attributes:	
Name	Description
X	
У	
Responsibilities:	
Name	Collaborator
Enkapsulacja pozycji pionków	Pawn
Generowanie losowej pozycji	MapCreator
Obliczanie odległości między dwoma pozycjami	SimulationLogic

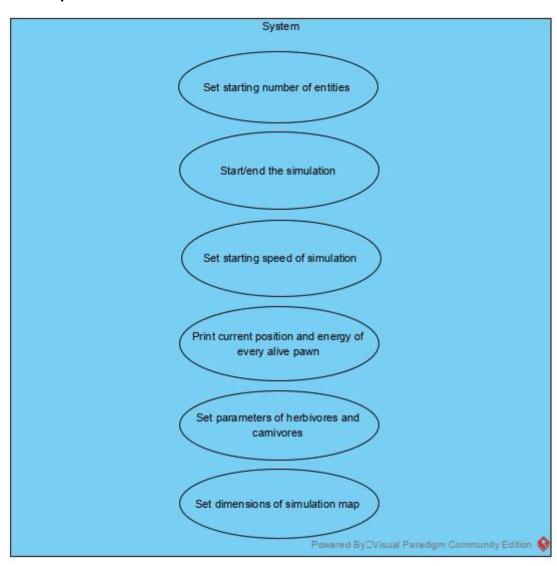
Super Classes:			
ouper classes.			
Sub Classes:			
Description: Matematyczny	wolder utuwany de		
przechowywania prędkości pionków i działań			
	na prigantosor promitori i aziaian		
wektorowých.	na priganosor promioni i azialan		
wektorowych.	na prigaressor promiser i aziaian		
wektorowych.	na prigances, premior i azidan		
wektorowych.	Description		
wektorowych. Attributes:			
wektorowych. Attributes:			
wektorowych. Attributes: Name x			
wektorowych. Attributes: Name X			
wektorowych. Attributes: Name X			
wektorowych. Attributes: Name X y Responsibilities:	Description		

Klasa "App"

Арр	
Super Classes:	
Sub Classes:	
Description: Główna klasa progra	mu z funkcją main.
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
Punkt startowy programu - funkcja main	SimulationCreator, FileConfigProvider, DefaultParametersProvider

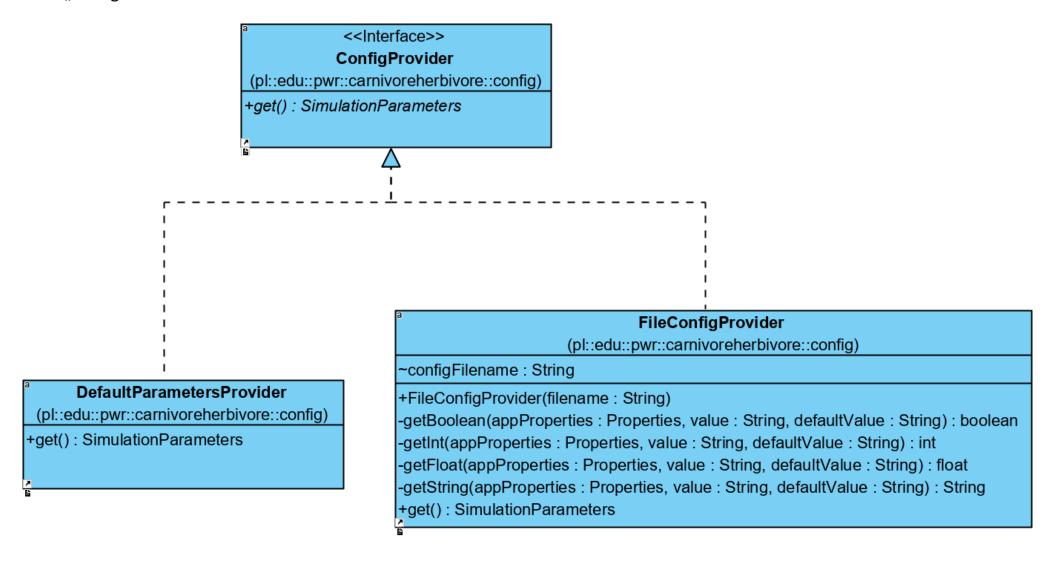
4. Diagram przypadków użycia





5. Diagram klas

Pakiet "config"



Pakiet "map"

SimulationMap

(pl::edu::pwr::carnivoreherbivore::map)

-pawns : Pawn

-pawnsPosition : Map<Pawn, Position>

+getPawns() : List<Pawn>

+getPawnsPositions(): Map<Pawn, Position>

+SimulationMap()

+addPawn(pawn : Pawn, position : Position) : void

+removePawn(pawn : Pawn) : void

MapCreator

(pl::edu::pwr::carnivoreherbivore::map)

-createPlants(simulationMap : SimulationMap, simulationParameters : SimulationParameters) : void

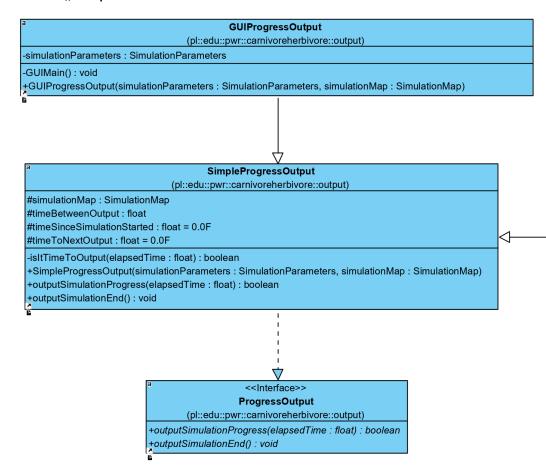
-createHerbivores(simulationMap : SimulationMap, simulationParameters : SimulationParameters) : void

-createCarnivores(simulationMap : SimulationMap, simulationParameters : SimulationParameters) : void

+create(simulationParameters : SimulationParameters) : SimulationMap

E

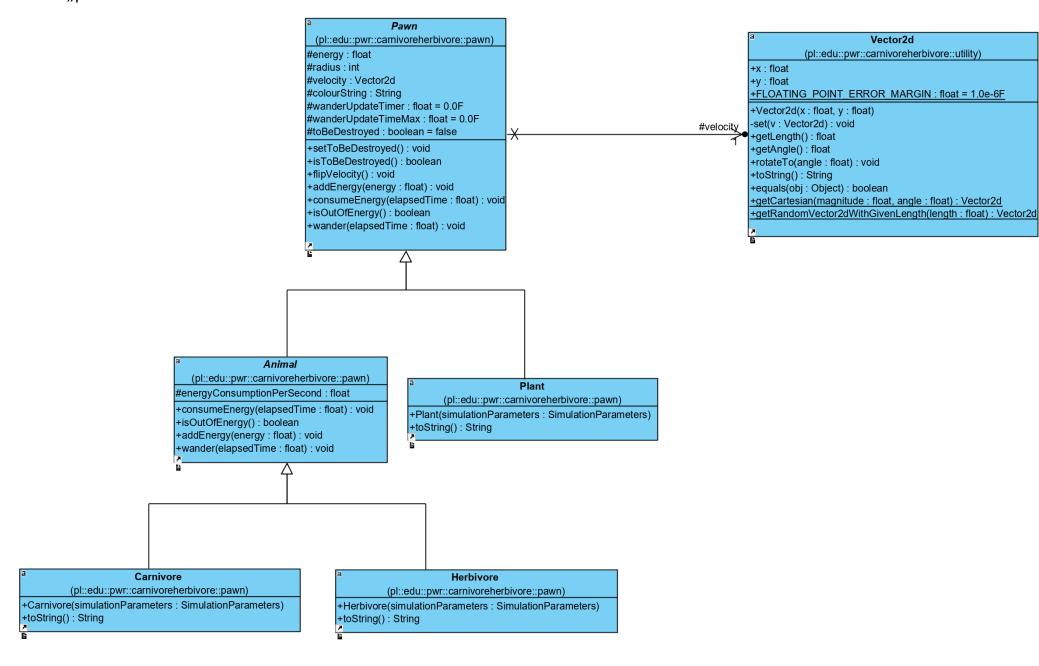
Pakiet "output"



a SimulationResults (pl::edu::pwr::carnivoreherbivore::output) -numberOfCarnivoresLeft : int = 0 -numberOfHerbivoresLeft : int = 0 -numberOfPlantsLeft : int = 0 -numberOfPawnsLeft : int = 0 +SimulationResults(simulationMap : SimulationMap)

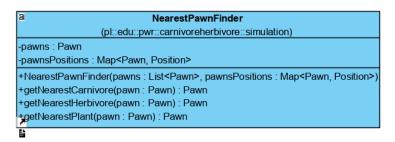
TerminalProgressLogger (pl::edu::pwr::carnivoreherbivore::output) -properTerminalOutput(simulationMap : SimulationMap, elapsedTime : float) : void +outputSimulationProgress(elapsedTime : float) : boolean +TerminalProgressLogger(simulationParameters : SimulationParameters, simulationMap : SimulationMap)

Pakiet "pawn"



Pakiet "simulation"

SimulationLogic (pl::edu::pwr::camivoreherbivore::simulation) -simulationParameters : SimulationParameters -simulationMap : SimulationMap -pawns : Pawn -pawnsPositions : Map<Pawn, Position> +SimulationLogic(simulationParameters : SimulationParameters, simulationMap : SimulationMap) -setPawnVelocityToApproachPosition(pawn:Pawn, pawnPosition:Position, destination:Position): void +updatePawnsWanderTimer(elapsedTime: float): void isWithinSightRange(firstPawn : Pawn, secondPawn : Pawn, sightRange : float) : boolean -updateHerbivoreVelocity(herbivore: Pawn): void -updateCarnivoreVelocity(carnivore: Pawn): void +updatePawnsVelocity(): void +movePawns(elapsedTime: float): void +makePawnsConsumeEnergy(elapsedTime : float) : void +checkPawnsForStarvation(): void +boundPawnsWithinMap(): void -collide(first : Pawn, second : Pawn) : void +checkPawnsForCollision(): void +destroyEatenOrStarvedPawns(): void +shouldSimulationEnd(): boolean



a SimulationCreator
(pl::edu::pwr::camivoreherbivore::simulation)

tcreate(simulationParameters : SimulationParameters) : Simula...

SimulationParameters (pl::edu::pwr::camivoreherbivore::simulation) +mapWidth: int +mapHeight : int +speedOfSimulationMultiplier: float +startingNumberOfPlants: int +startingNumberOfHerbivores: int +startingNumberOfCarnivores: int +herbivoreSpeed : float +camivoreSpeed: float +plantNutritionalValue : float +herbivoreStartingEnergy: float +carnivoreStartingEnergy : float -simulationParameters. +herbivoreEnergyConsumptionPerSecond: float +carnivoreEnergyConsumptionPerSecond : float +plantRadius: int +herbivoreRadius : int +carnivoreRadius: int +herbivoreSightRange : float +carnivoreSightRange: float +baseHerbivoreNutritionalValue : float +timeToSetNewRandomHerbivoreWanderDirection: float +timeToSetNewRandomCamivoreWanderDirection: float +colourStringOfPlant: String +colourStringOfHerbivore: String +colourStringOfCarnivore: String +useGUI: boolean +timeBetweenProgressOutputInTerminal: float +numberOfPawnsToEndSimulation: int +endSimulationWhenNoHerbivoresLeft: boolean +endSimulationWhenNoCarnivoresLeft: boolean +collisionRangeMultiplierOfSumOfRadii: float *SimulationParameters(mapWidth: int, mapHeight: int, spe. -simulationParameters 1 1 Simulation (pl::edu::pwr::carnivoreherbivore::simulation) -simulationParameters : SimulationParameters -simulationMap : SimulationMap -progressOutput : ProgressOutput +Simulation(simulationParameters : SimulationParameters, simulationMap : SimulationMap, progressOutput : ProgressO. +startSimulation(): void -updateSimulationLogic(elapsedTime : float) : boolean outputSimulationProgress(elapsedTime: float): void

Pakiet "utility"

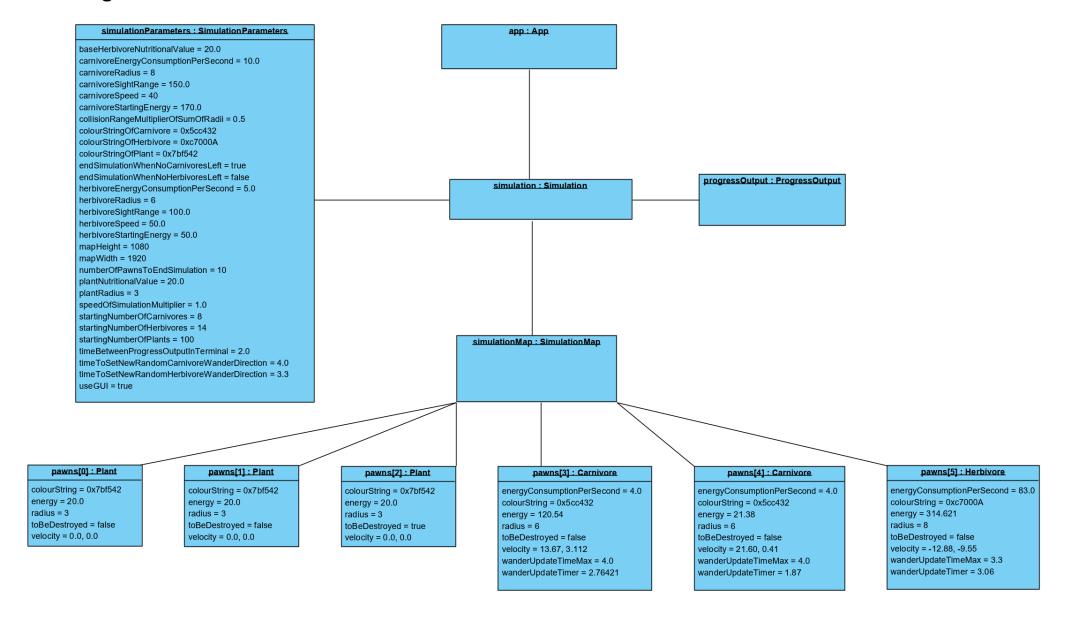
Position
(pl::edu::pwr::carnivoreherbivore::utility)

+x : float
+y : float
+Position(x : float, y : float)
+getRandomPosition(maxX : float, maxY : float) : Position
+toString() : String
+getDistanceBetween(first : Position, second : Position) : float

Klasa "App"

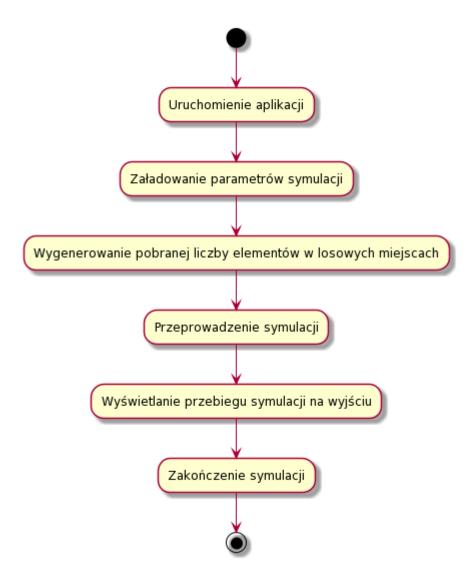
```
App
(pl::edu::pwr::carnivoreherbivore)
-getSimulationParameters() : SimulationParameters
+main(args : String[]) : void
```

6. Diagram obiektów



7. Diagramy aktywności

Diagram aktywności



Cykl symulacji



Zachowanie roślinożercy

