



**СУ “Св. Климент Охридски”,  
ФМИ – Софтуерно инженерство  
Курсов проект по Обектно-ориентирано  
програмиране**

# **Име на проект**

Иванка Златкова Зайчева, Факултетен № 61779

# Съдържание

1. Въведение .....	3
2. Описание на програмния код.....	4-6
3. Използвани технологии.....	7

## Въведение

Програмата God представлява игра, в която ти си God и имаш право да създаваш планети. На всяка планета имаш право да създаваш животни, хора или неопределено същество. Когато програмата се стартира планетата, която вече е създадена е планетата Земята като тя няма обитатели. Докато създавате същества на заден план вече създадените се движат , когато са достатъчно близо един до друг на случаен принцип се определя дали те ще се атакуват или ще създадат поколение.

Всяко същество има следните характеристики:

- Енергия
- Сила
- Координати
- Състояние(яде, спи, търси храна,движи се, мисли или друго )
- Големина
- Тегло
- Име

Новосъздадените обекти получават имена на случаен принцип.

Съществата могат да се атакуват, да се движат или да променят състоянието си. Когато едно същество атакува друго, то намалява енергията второто. Когато енергията на същество е по-малка или равна на нула то умира. Също така имате право да:

- Убиете всички същества от дадена планета
- Унищожите дадена планета
- Получите статистика за всички планети и населението им

# Описание на програмния код

Програмния код съдържа интерфейса IEntity , който съдържа методите :

- `void Attack(IEntity entityAttacked)`
- `void Move(Point2D a)`
- `void DoAction(AEntity entityAttacked),`

както и свойствата:

- `Point2D Point{get;set;}`
- `string Name{get;set;}`
- `double Energy { get; set; }`
- `double Power { get; set; }`
- `double Size { get; set; }`
- `double Weight { get; set; }`
- `Point2D Point { get; set; }`
- `State State { get; set; }`

IEntity се наследява от абстрактния клас AEntity. AEntity имплементира методите и свойствата на интерфейса IEntity. AEntity се наследява от Entity и Animal. Entity изцяло наподобява AEntity.

Animal от своя страна притежава допълнителните методи:

- `public void Eat()`
- `public void SearchingForFood()`
- `public void Sleep()`

Класът Human наследява класа Animal. Human съдържа допълнителните метод:

- `public void Analyze()`

Има два изброими типа данни. Единият е `State`, който съдържа състоянията, в които състезанията могат да са. Другият е `EntityType`, който представлява всички видове същества.

Класът `Point2D` е създаден, за да може да се определя състоянието на създанията, тяхната позиция се определя на случаен принцип след като са създадени. Той съдържа два метода за X и Y.

Класът `RandomG` съдържа два метода, за създаване на случаен принцип на имена и числа:

- `public double RandomNumbers(int minimum, int maximum)`
- `public string RandomName()`

`CommandManager` е клас в който има метод `Read()`, чрез който потребителя дава команди от конзолата. Възможните команди са:

- `add<name_planet>`-потребителят създава нова планета
- `add <name_planet> <entity/animal/human/unknown> <number_creatures>`- създава определен тип същества на дадена планета-определен брой
- `kill <name_planet>` -убива всички същества от дадена планета
- `destroy <name_planet>`-унищожава дадена планета
- `statistic` – извежда статистика за всички планети и всички същества на тях

Класът `Scene` служи за създаване на нови същества, както и съдържа списък от всички създадени планети-той се използва в класа `CommandManager`. Той съдържа метода `CreateEntity(EntityType creature, Planet planet)` чрез който се създават същества и член променливата: `public static List<Planet> AllPlanets`, която съдържа всички създадени планети.

Класът `Planet` представлява самите планети. Той съдържа списък на всички жители на дадена планета, както и метод за създаване на дадени същества на дадена планета и метод, който връща броя на жителите на дадена планета.

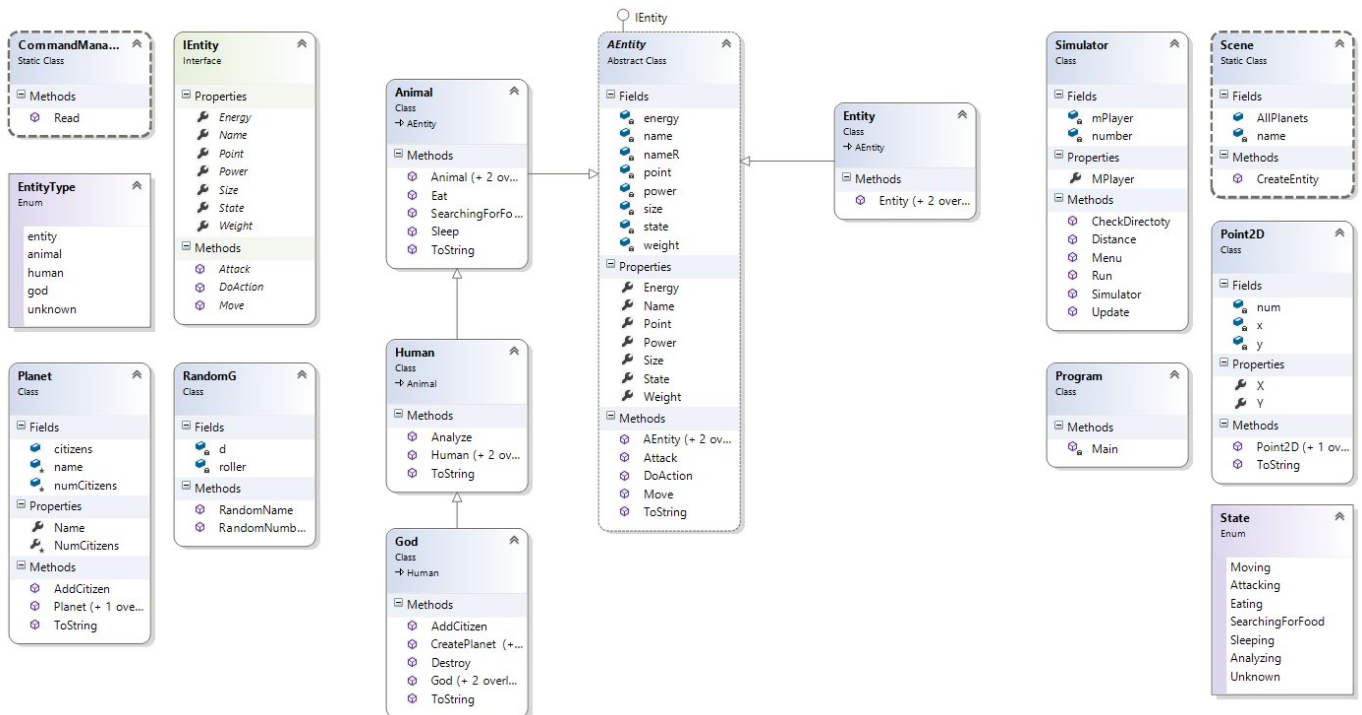
Класът `Simulator` съдържа няколко важни метода:

- `public static void CheckDirectoty()`, който проверява на какво разстояние са съществата едно от друго и когато са на разстояние по-малко или равно на пет те или създават

ново поколение или се атакуват-действието се осъществява на случаен принцип.

- `public static void Update()` – обновява местоположението на всички същества от всички планети
- `public void Menu()` – стартира командите от конзолата
- `public void Run()` – прави отделна нишка за метода `Update()` и стартира `Menu()`

### Class diagram:



## *Използвани технологии*

- Език за програмиране: C#
- Платформа: .Net framework 4.5
- Среда за разработка: Visual Studio 2013