|  |  |
| --- | --- |
| SU_logo.png | **Софийски университет “Св. Климент Охридски”**  **Факултет по математика и информатика** |

**ДОКУМЕНТАЦИЯ**

към проект по „ASP програмиране“

Тема: Симулация на движението на роботи по карти.

|  |  |
| --- | --- |
| **Изготвил:**  ***Антон Дудов, ФН: 71488, ИС курс 3*** | **Под ръководството на:**  ***доц. д-р Павел Павлов*** |

София 2016

**Съдържание:**

1. **Описание на проекта.**
2. **Файлове в проекта и описание за тях.**
3. **DTD (Robosim.dtd).**
4. **Описание на елементите на DTD (Robosim.dtd).**
5. **Описание на атрибутите на DTD (Robosim.dtd).**
6. **Описание на базата от данни (БД).**
7. **Схема на взаимовръзката между таблиците на БД.**
8. **Прехвърляне на информацията от XML към БД.**
9. **Приложение, работещо с XML файловете и БД.**
10. **Описание на проекта.**

Проектът представя структурата на роботи, карти и алгоритми за движението на роботите по съответните карти. Описано е:

* Роботи
* Карти
* Околни среди
* Алгоритми

В „Роботи“ има информация за всеки робот. Геометрията на робота, какви сензори има, разположение на колелата, перки, информация за други подвижни части, максимални скорости, какви среди може да преминава.

В „Карти“ има информация за всяка карта. Пътят до двоичния файл, в който са данните за картата, информация за денивелацията и какви среди(околни среди) се срещат в нея.

В „Околни среди“ има информация за средите(околната среда). Всяка околна среда има име и стойност, която показва с какво „усилие“ робота преминава през нея за единица време. (Например такива среди биха били: суша, въздух(ако трябва да бъде „прескочена“ някоя пропаст или стена) и вода.

В „Алгоритми“ има информация за всеки алгоритъм. Данните за алгоритъма са името, сложността му, дали поддържа множество от целеви точки и дали може да прави разлика между различни околни среди(например- през вода да преминава по-бавнo, отколкото по суша). Използва се за определяне на кой алгоритъм за търсене да бъде пуснат.

1. **Файлове в проекта и описание за тях.**

* RoboSim.dtd – Файл, съдържащ описанието на елементите и атрибутите, които се съдържат в XML документите. Съдържанието му е приложено по-нататък в документацията.
* ValidXMLDocument-X.xml – XML документите, които са валидирани спрямо RoboSim.dtd. В името на файла „Х“ представлява номера на документа и приема стойности от 1 до 20.
* ValidateXML.aspx – Уеб страница базирана на ASP.NET която проверява валидността на XML документите и записва в базата данни информацията съдържаща се във валидните от тях. Страницата визуализира резултатите от валидацията и запазването на информацията.
* XmlToDatabase.cs – файл написн със средствата на езика С#, който извършва същинската валидация и запис в базата от данни.
* AddInformation.aspx – Уеб страница базирана на ASP.NET, която съдържа в себе си форма за въвеждане на информация от потребителя. Освен това тя записва въведената информация в нов XML файл и в базата от данни.

1. **DTD (RoboSim.dtd).**

<?xml version="1.0" encoding="UTF-8"?>  
<!ELEMENT RoboSimulation (SimulationName, SimulationOwner, SimulationOwnerEmail, Environments, Robots, Maps, Algorithms)>  
 <!ELEMENT SimulationName (#PCDATA)>  
 <!ELEMENT SimulationOwner (#PCDATA)>  
 <!ELEMENT SimulationOwnerEmail (#PCDATA)>  
 <!ELEMENT Environments (Environment\*)>  
 <!ELEMENT Environment (TravelCostEnter, TravelCostIn, TravelCostExit, Damage)>  
 <!ATTLIST Environment id ID #REQUIRED  
 name CDATA #REQUIRED>  
 <!ELEMENT TravelCostEnter (#PCDATA)>  
 <!ELEMENT TravelCostIn (#PCDATA)>  
 <!ELEMENT TravelCostExit (#PCDATA)>  
 <!ELEMENT Damage (#PCDATA)>  
 <!ELEMENT Robots (Robot\*)>  
 <!ELEMENT Robot (RobotMeshGrid, Speed, SpeedBack, TurningSpeed, TurningSpeedBack, Wheels?, Sensors?, Rotors?)>  
 <!ATTLIST Robot id ID #REQUIRED  
 environments IDREFS #REQUIRED  
 name CDATA #REQUIRED  
 owner CDATA #REQUIRED>  
 <!ELEMENT RobotMeshGrid (#PCDATA)>  
 <!ELEMENT Speed (#PCDATA)>  
 <!ELEMENT SpeedBack (#PCDATA)>  
 <!ELEMENT TurningSpeed (#PCDATA)>  
 <!ELEMENT TurningSpeedBack (#PCDATA)>   
 <!ELEMENT Wheels (Wheel\*)>   
 <!ELEMENT Wheel (WheelMeshGrid, WheelDiameter, WheelWidth)>  
 <!ATTLIST Wheel driving (ДА|НЕ) #REQUIRED>  
 <!ELEMENT WheelMeshGrid (#PCDATA)>  
 <!ELEMENT WheelDiameter (#PCDATA)>  
 <!ELEMENT WheelWidth (#PCDATA)>  
 <!ELEMENT Sensors (Sensor\*)>  
 <!ELEMENT Sensor (SensorMeshGrid, NumberOfValusPerSecond)>  
 <!ATTLIST Sensor name CDATA #REQUIRED  
 valueType CDATA #REQUIRED>  
 <!ELEMENT SensorMeshGrid (#PCDATA)>  
 <!ELEMENT NumberOfValusPerSecond (#PCDATA)>  
 <!ELEMENT Rotors (Rotor\*)>  
 <!ELEMENT Rotor (RotorMeshGrid, RotorLiftingPower)>  
 <!ELEMENT RotorMeshGrid (#PCDATA)>  
 <!ELEMENT RotorLiftingPower (#PCDATA)>  
 <!ELEMENT Maps (Map\*)>  
 <!ELEMENT Map (MapData, Denivelation)>  
 <!ATTLIST Map id ID #REQUIRED  
 environments IDREFS #REQUIRED>  
 <!ELEMENT MapData (#PCDATA)>  
 <!ELEMENT Denivelation (#PCDATA)>  
 <!ELEMENT Algorithms (Algorithm\*)>  
 <!ELEMENT Algorithm (Complexity, Depth)>  
 <!ATTLIST Algorithm id ID #REQUIRED  
 name CDATA #REQUIRED   
 diffEnvironments (ДА|НЕ) #REQUIRED  
 multipleDestPoints (ДА|НЕ) #REQUIRED>  
 <!ELEMENT Complexity (#PCDATA)>  
 <!ELEMENT Depth (#PCDATA)>

1. **Описание на елементите на DTD (RoboSim.dtd).**
2. RoboSimulation – основен елемент на всеки XML документ(кореновият елемент).
3. SimulationName – съдържа информация за името на симулацията.
4. SimulationOwner – съдържа информация за собственика на симулацията.
5. SimulationOwnerEmail – съдържа информация за пощата на собственика на симулацията.
6. Environments – обединява всички околни среди.
7. Environment – съдържа информация за конкретна околна среда.
8. TravelCostEnter – съдържа информация за „цената“ за влизане в конкретната околна среда.
9. TravelCostIn – съдържа информация за „цената“ за преминаване на единица разстояние в конкретната околна среда.
10. TravelCostOut – съдържа информация за „цената“ за излизане от конкретната околната среда.
11. Damage – съдържа информация за това какво количество щети ще поеме робота, ако прекара единица време в тази конкретна околна среда.
12. Robots – обединява всички роботи.
13. Robot – съдържа информация за конкретен робот.
14. RobotMeshGrid – съдържа адреса на файла, в който е информацията за геометрията на робота.
15. Speed – съдържа информация за скоростта на робота при движение в права линия.
16. SpeedBack – съдържа информация за скоростта на робота при движение в права линия назад.
17. TurningSpeed – съдържа информация за скоростта на робота при завиване.
18. TurningSpeedBack – съдържа информация за скоростта на робота при завиване назад.
19. Wheels – обединява всички колела на робота.
20. Wheel – съдържа информация за конкретно колело.
21. WheelMeshGrid – съдържа адреса на файла, в който е информацията за геометрията на колелото.
22. WheelDiameter – съдържа информация за диаметъра на колелото. С нея ще се пресметне какво разстояние ще премине робота за едно завъртане на колелото.
23. WheelWidth – съдържа информация за ширината на колелото.
24. Sensors – обединява всички сензори на робота.
25. Sensor – съдържа информация за конкретен сензор.
26. SensorMeshGrid – съдържа адреса на файла, в който е информацията за геометрията на сензора.
27. NumberOfValusPerSecond– съдържа информация за това колко измервания и съответно върнати стойности може да отчете сензора за една секунда.
28. Rotors – обединява всички перки на робота.
29. Rotor– съдържа информация за конкретна перка.
30. RotorMeshGrid - съдържа адреса на файла, в който е информацията за геометрията на перката.
31. RotorLiftingPower – съдържа информация за подемната сила на перката. С нея ще се пресметне какво разстояние ще измине за единица оборот на перката.
32. Maps – обединява всички карти.
33. Map – съдържа информация за конкретна карта.
34. MapData – съдържа адреса на файла, в който е информацията за самата карта.
35. Denivelation – съдържа инфрмация за максималната денивелация по картата в градуси.
36. Algorithms – обединява всички алгоритми за търсене по картите.
37. Algorithm – съдържа информация за конкретен алгоритъм.
38. Complexity – съдържа информация за времевата сложност, за която може да бъде намерен път между робота и целта му.
39. Depth – съдържа информация за това на каква дълбочина максимално да се пуска алгоритъма. Т.е. ако е много голяма картата може на етапи да се търси пътя от робота до целта му.
40. **Описание на атрибутите на DTD (RoboSim.dtd).**
41. name – атрибут на елемента Environment. Показва името на околната среда.
42. id – атрибут на елемента Environment. Показва идентификационният номер на всяка от околните среди.
43. id – атрибут на елемента Robot. Показ ва идентификационният номер на всеки от роботите.
44. environments – атрибут на елемента Robot. Показва идентификационните номера на околните среди, които конкретния робот може да преминава.
45. name – атрибут на елемента Robot. Показва името на робота.
46. Owner – атрибут на елемента Robot. Показва собственика на робота.
47. driving – атрибут на елемента Wheel. Показва дали конкретното колело на робота е задвижващо или не. (ДА|НЕ) стойности.
48. name – атрибут на елемента Sensor. Показва името на робота.
49. valueType – атрибут на елемента Sensor. Показва каква величина измерва сензора(например температура, разстояние до обект и т.н.).
50. id – атрибут на елемента Map. Показва идентификационният номер на всяка от картите.
51. environments – атрибут на елемента Map. Показва идентификационните номера на околните среди, които конкретната ката съдържа.
52. id – атрибут на елемента Algorithm. Показва идентификационният номер на всеки от алгоритмите.
53. name – атрибут на елемента Algorithm. Показва името на алгоритъма.
54. diffEnvironments – атрибут на елемента Algorithm. Показва дали алгоритъма поддържа различаване на околните среди. (ДА|НЕ) стойности.
55. multipleDestPoints – атрибут на елемента Algorithm. Показва дали алгоритъма поддържа откриване на максимален път до множество целеви точки. (ДА|НЕ) стойности.

Всички атрибути са задължителни.