ОЩЕ ЗА АДАПТИРАНЕТО НА ИНТЕРАКТИВНИ ИГРОВИ ЕЛЕМЕНТИ В ТРИИЗМЕРНА СРЕДА ЗА ЦЕЛИТЕ НА ОБУЧЕНИЕТО

Ивайло И. Буров

Резюме: *Тази статия е продължение на статията „Разработване и адаптиране на интерактивни игрови елементи в триизмерна среда за образователни цели“ и споделя опит, свързан с адаптирането на съществуващи интерактивни решения в 3D среда за образователни цели. Представеният материал може да бъде полезен за разработчици, учители, докторанти и студенти, които са приели предизвикателството да работят в област, изискваща опит в диапазон от 3D моделиране и анимация до програмиране в избрана среда.*

Ключови думи: *интерактивен, виртуален 3d, модел, адаптация, обучение, игри в обучението*

MORE FOR ADAPTATION OF INTERACTIVE GAME ELEMENTS IN A THREE-DIMENSIONAL ENVIRONMENT FOR EDUCATIONAL PURPOSES

Ivailo I. Burov

ABSTRACT*: This article is a continuation of the article "Development and adaptation of interactive game elements in a three-dimensional environment for educational purposes" and shares experiences related to the adaptation of existing interactive solutions in 3D environment for educational purposes. The presented material can be useful for developers, teachers, PhD students and students who have accepted the challenge of working in an area requiring experience in a range from 3D modeling and animation to programming in a chosen environment.*

KEYWORDS*: interactive, virtual, 3d, model, adaptation, education, games in education*

Интегриране на шаблон с отворен код за приключенски игри GOAT (Godot Open Adventure Template, създаден за игровия двигател Godot.

Този шаблон е създаден от Miskatonicstudio, като поставената цел е улесненото създаване на приключенски игри. Поради тази причина в съществуващия проект е налична само камера от първо лице (FPS), което налага самостоятелно интегриране на допълнителна камера от трето лице за нуждите на игри свързани с обучението. При кореспонденция, авторът предвижда подобна функционалност в бъдеще, но на този етап не се забелязва публикувано развитие. При адаптация за игри с образователна цел първоначално се забелязват само положителните страни заложени в този проект, но в хода на разработката възникват много функционални липси, които за съжаление се доближават до програмния дизайн на проекта. Тук е необходимо да се отвори скоба за плюсовете и минусите при използване на проекти с отворен код и платените (като тези при Unity игровия двигател, а и при Unreal). Платените разширения обикновено предлагат по-разширена функционалност с целите за комерсиализация, но са със затворен код и ако възникне необходимост от нова функционалност, то е необходимо достатъчно клиенти да се обърнат към автора с молба за реализирането и. За разлика от този недостатък при разширенията с отворен код всеки може да добавя собствен код за разширяване на функционалността. Ако авторът на платено разширение се откаже от поддръжката му, то при потребителски проект, ползващ това разширение възникват проблеми и често е необходимо намиране на друго решение и много труд

за преработка на проекта. При проектите с отворен код почти винаги се намират разработчици, които продължават съществуването на успешен проект, чиято поддръжка е спряна от автора му. Също така всеки има възможност да направи сам това според познанията си в областта.

Тук ще бъдат разгледани както положителните страни на проекта GOAT, така и откритите недостатъци при използването му в интерактивни учебни игри, а също и откритите липсващи функционалности при практическото му адаптиране в учебна игра с реален сценарий.

Положителните функционалности на GOAT са свързани с разработените възможности за интерактивно взаимодействие с 3d модели, наречени от автора интерактивни елементи и интерактивни екрани, разделени на три типа:

- **Нормални**: могат да се активират многократно, напр. врата, която се отваря и затваря;
- **За еднократно ползване**: могат да се активират само веднъж, напр. скала, която може да бъде избутана или храна, която може да бъде консумирана;
- **Инвентар**: активирането му премахва интерактивния елемент от обкръжението и го добавя като елемент от инвентара. *Може да бъде за еднократно или многократно използване, което означава че всъщност съществуват* ***две главни групи: нормални и инвентар***.

Приложението на такива интерактивни елементи е подходящо при решаване на задачи, свързани с откриване на зададени обекти или артефакти и прилагането им по изисквано местоназначение.

За целта е изградена инвентарна система за събиране на интерактивни елементи и взаимодействие с тях. Примери от автора са представени на фигури 1 и 2.

Фигура 1. Активиране на генератор в GOAT, представен като нормален интерактивен елемент.

Фигура 2. Взаимодействие с инвентарен тип интерактивен елемент в GOAT.

Интерактивните екрани представляват симулатори на реален обект в самата игра, явяваща се също симулация. Някои примери за такива интерактивни обекти са: настолен компютър (фигура 3), лаптоп, смартфон, калкулатор или др., предвидени от авторите на сюжета.

Фигура 3. Взаимодействие с интерактивен екран в GOAT.

Интерактивните екрани в GOAT са триизмерни интерактивни модели, върху чиято двуизмерна текстура, назначена за екран се проектира двуизмерно изображение. Чрез такъв подход става възможна симулация на виртуално работещи компютри или смарт устройства, като в най-разширения случай върху екрана може да се проектира дори готова 2D игра, която играчът ще играе, само че на виртуално устройство в същинската игра - техника известна като „**игра в играта**“ или „**мини игра**“ - много подходяща при създаване на игри, насочени към обучението.

При необходимост от изграждане на някаква последователност, се въвеждат глобални променливи, а активирането на интерактивния елемент е възможно само при установено състояние на тази променлива, в противен случай е предвидена подсказка. Например ако желаем да включим дистанционното, то първо трябва да поставим батериите в него, а ако желаем да пуснем виртуалния персонален компютър първо трябва да включим захранващия генератор. С това се постига определено ниво на уникалност, но разбира се за сметка на универсалността и лекотата при създаване на съдържание.

Забелязани недостатъци

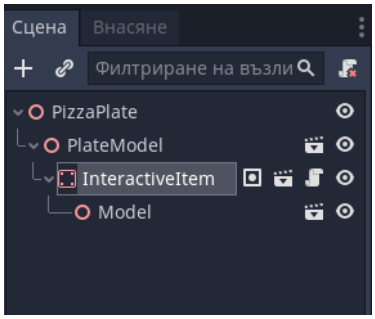
Въпреки доброто предоставяне на интерактивни възможности, трябва да бъде отбелязана необходимостта от отделно кодиране на всеки интерактивен елемент, което изисква постоянно съдействие на ИТ специалист, запознат с игровата разработка, програмирането свързано с избрания игрови двигател и конкретно с използвания интерактивен GOAT шаблон.

- За създаването на интерактивен елемент е необходимо да се спазва определено разположение в структурата на зададените за целта папки от файловата система, да се програмират определени събития, свързани със селекция или активация **на всеки** интерактивен обект, като понякога и се анимират програмно ако необходимо. Това прави разработката бавна.

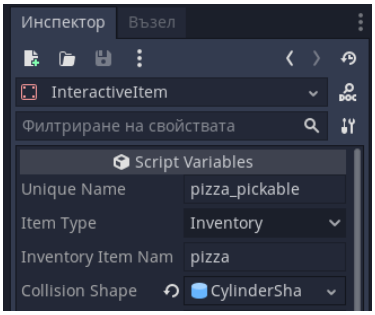
- Използването на твърдо вградени в кода имена на интерактивните елементи също се отчита като недостатък. Това изключва възможността за използване на един интерактивен елемент няколко пъти и изисква дублирането му за такава цел.

- Няма връзка между интерактивните елементи извън инвентарната система и в нея. На практика интерактивните елементи, които аватарът среща в обкръжението са различни от тези в инвентарната система. Могат да ползват един същи 3d модел, а също и различни такива, но на практика това са два различни интерактивни обекта и програмният дизайн е построен така, че единият елемент „не знае“ нищо за съществуването на другия. Те не споделят общи свойства и местонахождение в триизмерното пространство. Това налага дублиране на подобни свойства или преработка на програмния дизайн. *С цел запазването на програмния дизайн при адаптирането на подобни интерактивни елементи като конкретно решение послужи кодирането на шаблони, които могат да се прикачват към интерактивните елементи за избягване на този недостатък.*

- В GOAT един триизмерен модел се превръща в интерактивен, ако към него се прикрепи т.н. InteractiveItem (интерактивен артикул, явяващ се възел в сцената), което е лесно и удобно по принцип (Фигура 4).

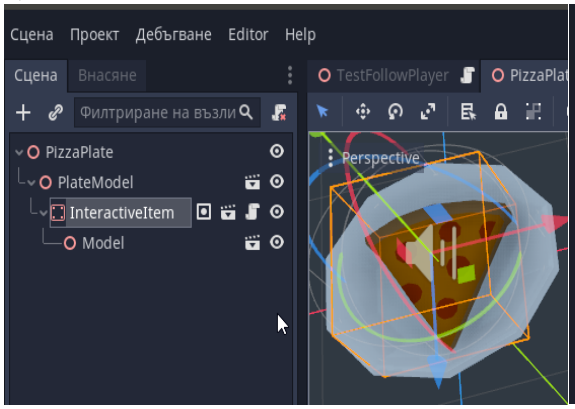


Фигура 4.
Интерактивен артикул като възел в сцената.

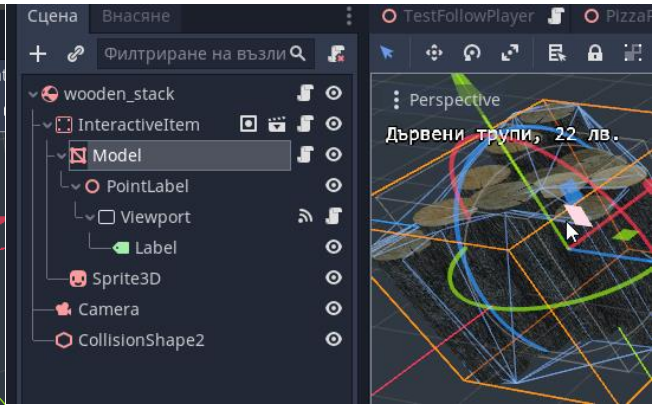


Фигура 5.
Уникално и инвентарно име на артикула.

В интерактивния артикул трябва да се назначи уникално име на интерактивния елемент, а също и съответно име в инвентарната система (Фигура 5). По дизайн 3d моделът е предвиден като дъщерен на интерактивния артикул (InteractiveItem, Фигура 6). Този вариант обаче изключва работата с физична симулация. Ако интерактивният елемент, например дървени трупи или руда трябва да реагира с обкръжението спрямо физичните закони, представената структура не работи. За целта е необходимо създаването на отделна колизия спрямо твърдо тяло (RigidBody), относно 3d модела, която не допуска интерактивният артикул да бъде представен като родител (Фигура 7).



Фигура 6.
Моделът е дъщерен на интерактивния артикул. Използва се колизията на артикула.



Фигура 7.
Моделът отново е дъщерен на интерактивния артикул но артикулът се явява дъщерен на твърдото тяло (wooden_stack) с отделна колизия.

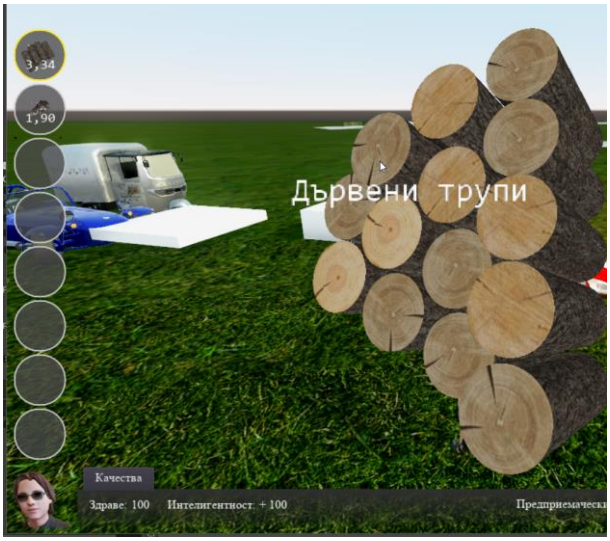
Това води до невъзможност за назначаване на уникално име на интерактивния артикул в главната сцена, освен ако не бъде локално представен, а това утежнява сцената и *налага допълнително дефиниране уникално име към главния възел*. При неспазване на уникалността на имената, действията спрямо един интерактивен елемент се прилагат към всички, защото GOAT работи с уникалните имена. *За избягването на този според автора на статията недостатък, в създадения шаблон за разширяване на функционалността се включва допълнително отчитане на колизия с лъч спрямо 3d модела*.

- Липсва функционалност за прикачване на допълнителни свойства към инвентарните обекти, примерно като брой, цена и др. Липсата на функционалност за отчитане на брой прави невъзможно събиране на множество ресурси от еднакъв тип, което много ограничава

приложението в учебни игри с развит сценарий. При кореспонденция, авторът на GOAT изразява виждане за бъдещо внедряване на такава функционалност, но на този етап е необходимо собствено решение. По дизайн инвентарните елементи се съхраняват в кода посредством масив от данни. Ако е използван например Dictionary (речник в GDScript - аналог на map в езици като C++), то разширяването на функционалността в тази насока не би била проблем. Оказа се, че преобразуването на масива в речник изисква голям обем преработка на съществуващия код и за тази цел *към използвания масив бе добавен речник, синхронизиран с него по индекс. По този начин не се променя значително оригиналната структура на GOAT, при увеличена функционалност.*

Крайният резултат с модификация за добавяне на допълнителна функционалност е демонстриран на фигури 8, 9 и 10.

Фигура 8. Изграждане на възможност за събиране на множество ресурси от един вид.



Фигура 9. Отчитане на допълнителни параметри като брой и цена в инвентарната система.

Фигура 10. Реализация на интерактивните елементи като физическо тяло (Rigid Body) за взаимодействие с обкръжението по физични закони.

Интернет ресурси, използвани при разработката

1. GODOT, https://godotengine.org/
2. GOAT (Godot Open Adventure Template), https://github.com/miskatonicstudio/goat

доц. д-р Ивайло Иванов Буров
Шуменски университет "Епископ Константин Преславски"
Педагогически факултет
катедра .Предучилищна и начална училищна педагогика
имейл: i.burov@shu.bg