

**Sławomir ISKIERKA** 

---

*ORCID: 0000-0002-2834-8696. Prof. PCz, dr hab. inż., Politechnika Częstochowska,  
ul. H. Dąbrowskiego 69, 42-200 Częstochowa; e-mail: iskierka@el.pcz.czyst.pl*

---

## **TWORZENIE APLIKACJI INTERAKTYWNYCH W WYBRANYCH ŚRODOWISKACH 3D**

### **CREATING INTERACTIVE APPLICATIONS IN SELECTED 3D ENVIRONMENTS**

**Słowa kluczowe:** programowanie 3D, interakcja, projekty graficzne.

**Keywords:** 3D programming, interaction, graphic designs.

#### **Streszczenie**

Przedstawiono zagadnienia związane z grafiką komputerową 3D i tworzeniem projektów graficznych 3D. Opisano wybrane środowiska do projektowania i programowania 3D. Pokazano możliwości wykorzystywania wybranych środowisk graficznych 2D i 3D do tworzenia aplikacji interaktywnych.

#### **Abstract**

Issues related to 3D computer graphics and creating 3D graphic designs were presented. Selected environments for 3D design and programming were described. The possibilities of using selected 2D and 3D graphic environments to create interactive applications were shown.

#### **Wstęp**

W procesie tworzenia aplikacji interaktywnych wykorzystywanych jest wiele narzędzi oraz środowisk graficznych i programistycznych. W zależności od rodzaju aplikacji oraz jej przeznaczenia można podzielić pracę na kilka charakterystycznych etapów związanych z projektowaniem wstępnym, tworzeniem prototypu, tworzeniem elementów aplikacji, importowaniem elementów aplikacji

do zintegrowanego środowiska do tworzenia trójwymiarowych gier komputerowych lub innych aplikacji interaktywnych, kodowaniem, testowaniem i budowaniem aplikacji. Na każdym z tych etapów korzysta się z odpowiednich narzędzi i środowisk graficznych oraz programistycznych do uzyskania oczekiwanego efektu. Wśród wielu takich środowisk na uwagę zasługują programy z grupy programów open source. Są to Inkscape, Gimp, Blender, bezpłatna wersja programu Unity. Pozwalają one na pracę związaną z tworzeniem elementów aplikacji interaktywnej począwszy od tworzenia obiektów wektorowych, obróbkę plików rastrowych, modelowania 3D oraz animacji. Użytkownicy wykorzystują również środowisko Unity będące zestawem narzędzi do tworzenia wieloplatformowych gier i aplikacji interaktywnych, bazujących na najpopularniejszym na świecie silniku 3D i 2D.

## **Grafika komputerowa 3D i wybrane środowiska graficzne 2D i 3D**

Grafika komputerowa trójwymiarowa to grafika wykorzystująca trzy wymiary do reprezentacji danych geometrycznych (często kartezjańskich), które są przechowywane w komputerze w celu wykonywania obliczeń i renderowania obrazów dwuwymiarowych. Obrazy takie można przechowywać w celu wyświetlenia w odpowiednim momencie albo wyświetlać na bieżąco<sup>1</sup>. Ta popularna definicja grafiki trójwymiarowej uwzględnia jej najważniejsze cechy: dane są reprezentowane w trójwymiarowym układzie współrzędnych, ostatecznie grafika jest renderowana jako obraz dwuwymiarowy (np. na ekranie monitora), grafikę można wyświetlać na bieżąco, co oznacza, że gdy dane są zmieniane w ramach animacji albo też z powodu działań użytkownika, następuje ich aktualizacja bez widocznego opóźnienia. W grafice 3D wykorzystuje się trójwymiarowe komputerowe modele testowe<sup>2</sup>. Najczęściej używane modele testowe w grafice 3D to:

- model czajnika, który został stworzony w 1975 roku przez informatyka Martina Newellana Uniwersytecie Utah. Za źródło danych posłużył prawdziwy czajnik firmy Melitta kupiony wcześniej przez żonę Newella,
- Suzanne – jest to wbudowany prymityw w programie Blender, który jest alternatywą popularnego czajnika z Utah. Obiekt w kształcie głowy małpki wykorzystywany jest do testowania materiałów, oświetlenia, animacji. Autorem trójwymiarowego modelu jest Willem-Paul van Overbruggen (2002 r.),
- The Stanford Bunny to testowy model grafiki 3D opracowany przez Grega Turka i Marca Levoya w 1994 roku na Uniwersytecie Stanforda. Siatka

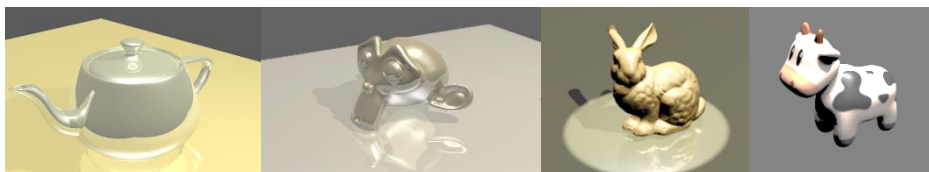
---

<sup>1</sup> T. Parasi, *Aplikacje 3D. Przewodnik po HTML5, WebGL i CSS3*, Gliwice 2014.

<sup>2</sup> [https://en.wikipedia.org/wiki/List\\_of\\_common\\_3D\\_test\\_models](https://en.wikipedia.org/wiki/List_of_common_3D_test_models) (dostęp: 5.04.2020 r.).

modelu składa się z 69 451 trójkątów. Model uzyskano przez skanowanie 3D ceramicznej figurki królika.

Jednym z najczęściej używanych modeli testowych w grafice komputerowej jest Spot. Jest to trójwymiarowy komputerowy model testowy opracowany przez Keenan Crane w 2012 roku. Siatka modelu cętkowanej krówki składa się z 5856 trójkątów oraz 2930 wierzchołków.



**Rys. 1. Najczęściej używane modele testowe w grafice 3D: teapot, Suzanne, The Stanford Bunny, Spot**

Źródło: opracowanie własne na podstawie [https://en.wikipedia.org/wiki/List\\_of\\_common\\_3D\\_test\\_models](https://en.wikipedia.org/wiki/List_of_common_3D_test_models) (dostęp: 05.04.2020).

W procesie tworzenia aplikacji interaktywnych wykorzystuje się równolegle wiele środowisk graficznych i programistycznych. Do tworzenia elementów aplikacji (gry) można wykorzystać programy do grafiki wektorowej i rastrowej. Najczęściej użytkownicy wybierają środowiska darmowe, takie jak program Inkscape oraz Gimp. Do modelowania 3D i animacji można wykorzystać program Blender oraz 3ds Max. Do tworzenia zaawansowanych aplikacji (gier) bardzo często użytkownicy wybierają silnik gier komputerowych Unity.

Blender jest darmowym programem open source do tworzenia grafiki i animacji 3D. Jest dostępny w wersjach dla systemów Windows, Linux i Mac OS. Blender jest wciąż rozwijany przez społeczność użytkowników, a prace te są koordynowane przez organizację non profit o nazwie Blender Foundation. Jest ona kierowana przez Tona Roosendaala, jednego z twórców Blendera<sup>3</sup>. W obecnej wersji zawiera ogromną gamę funkcji, narzędzi rzeźbiarskich, narzędzi symulacyjnych, animacyjnych, narzędzi do tworzenia efektów wizualnych i wiele innych. Obecnie Blender jest używany w wielu studiach animacyjnych, a także przez twórców gier, artystów i hobbystów<sup>4</sup>.

Ekran Blendera jest podzielony na prostokątne obszary, nazywane są one oknami (ang. *window*, a czasami także „obszarem” – *area*). Na domyślnym ekranie Blendera widać pięć okien. Okno widoku (*View 3D*) pokazuje trójwymiarową „scenę”, a na niej model. Okno struktury (*Outliner*) zawiera strukturę

<sup>3</sup> <https://www.blender.org/foundation/> (dostęp: 5.04.2020 r.).

<sup>4</sup> B. Simons, *Blender. Praktyczny przewodnik po modelowaniu, rzeźbieniu i renderowaniu*, Gliwice 2014.

sceny, przedstawioną w postaci symbolicznej. Okno właściwości (*Properties*) to obszar zawierający przyciski, pola numeryczne, przełączniki, za pomocą których użytkownik może zmienić właściwości elementów sceny. Okno informacyjne (*Info*) to menu główne programu oraz miejsce na komunikaty. Okno osi czasu (*Timeline*) wykorzystuje się w procesie animacji<sup>5</sup>.

Do tworzenia modeli 3D w programie Blender wykorzystuje się często krzywe. Blender umożliwia utworzenie kilku rodzajów krzywych: Bezier, NURBS, Path i innych. Krzywa Bezierra pozwala na utworzenie ścieżki do wyciągania określonych kształtów. Przypomina więc ona pod kilkoma względami nić, którą użytkownik układa w określony sposób. Jeśli nie ustalimy szerokości krzywej bądź też nie zostanie ona wytloczona, nie będzie można jej dostrzec po zrenderowaniu sceny.

Blender oferuje poza standardowymi metodami edycji obiektu również dodatkowe narzędzia, występujące pod nazwą Modifiers – modyfikatory. Mimo że każde narzędzie Modifier (modyfikator) służy do innych celów, to mają one kilka cech wspólnych.

W programach Blender oraz 3ds Max można tworzyć projekty elementów aplikacji modelując wyłącznie w środowisku 3D lub też wykorzystując projekty utworzone w programach do grafiki 2D. Korzysta się więc często z możliwości wykonania projektów w programach 2D np. Inkscape oraz Illustrator i przeniesienia ich (wyeksportowania) do programów Blender oraz 3ds Max. Do programu Blender można zaimportować projekty systemu identyfikacji wizualnej, które utworzono wcześniej w programie Inkscape w formacie zapisu grafiki wektorowej SVG (ang. *Scalable Vector Graphics* – skalowalna grafika wektorowa). W programie 3ds Max nie ma takiej możliwości, natomiast korzysta się z możliwości importu z programu Illustrator.

Program Inkscape jest rozpowszechniany na otwartej licencji GNU GPL. Jest edytorem dwuwymiarowej grafiki wektorowej o dużych możliwościach, zbliżonych do tych oferowanych przez jego komercyjne odpowiedniki. Udostępniany jest dla platform systemowych Windows, Linux, Mac OS X. Ogromną zaletą programu jest zastosowanie jako macierzystego formatu zapisu standaryzowanego, otwartego formatu zapisu grafiki wektorowej SVG<sup>6</sup>. Chociaż program korzysta ze standardu SVG będącego dialektem metajęzyka XML, aby stworzyć grafikę, użytkownik nie musi mieć wiedzy na temat tego formatu. W bardziej zaawansowanych projektach graficznych znajomość składni języka SVG może być jednak bardzo pomocna. Inkscape służy przede wszystkim do tworzenia logo, ikon, wykresów, diagramów, schematów i wielu innych projektów w grafice wektorowej.

---

<sup>5</sup> K. Kukło, J. Kolmaga, *Blender. Kompendium*, Gliwice 2007.

<sup>6</sup> K. Cieśla, *Inkscape. Podstawowa obsługa program*, Gliwice 2013.

Adobe Illustrator CC to rozbudowany program do tworzenia kompozycji wektorowych na potrzeby każdego projektu graficznego. Aplikacja oferuje zestaw zaawansowanych i precyzyjnych narzędzi do tworzenia profesjonalnych projektów zarówno grafikom, artystom, jak również twórcom stron internetowych. Omówione programy mają ogromne możliwości związane z tworzeniem elementów graficznych 2D i 3D.



**Rys. 2. Programy 2D wykorzystywane przy tworzeniu elementów 3D**

Źródło: opracowanie własne na podstawie <https://freedesignfile.com/238238-tattoo-ornaments-design-material-vector-04/> (dostęp: 5.04.2020 r.)

Środowisko Unity jest jednym z najczęściej używanych oraz najbardziej cenionych pakietów pozwalających na projektowanie gier komputerowych oraz aplikacji interaktywnych. Unity jest zintegrowanym środowiskiem do tworzenia animacji i gier. Jest wykorzystywane przez użytkowników od wielu lat i pracują z nim zarówno hobbysci jak i duże firmy. Za pomocą Unity tworzone są gry oraz interaktywne aplikacje dla przeglądarek internetowych, komputerów stacjonarnych, urządzeń przenośnych oraz konsol.

W środowisku Unity istotne jest rozróżnienie przestrzeni modelu i przestrzeni świata. Wszystkie współrzędne obiektów w przestrzeni 3D odnoszą się do punktu zerowego. Aby zdefiniować współrzędne danego obiektu w relacji do innego używa się przestrzeni modelu (zwanej przestrzenią obiektu). Oznacza to, że przy użyciu przestrzeni modelu można wyznaczać odległości od innych obiektów, wykorzystując współrzędne obiektu nadrzędnego jako nowy punkt zerowy dla każdego z jego obiektów podrzędnych.

Wykrywanie kolizji jest w silnikach gier zagadnieniem bardzo ważnym. Jest to metoda analizy utworzonego świata 3D pod kątem detekcji kolizji zachodzących między obiektami<sup>7</sup>. Poprzez przypisanie obiektowi komponentu zderzacza użytkownik może w rzeczywistości utworzyć wokół niego niewidzialną powierzchnię. W większości przypadków odzwierciedla ona kształt obiektu i wykrywa dowolne kolizje z innymi zderzaczami, co powoduje, że silnik gry odpowiednio na nie reaguje.

<sup>7</sup> W. Goldstone, *Projektowanie gier w środowisku Unity*, Gliwice 2014.

W środowisku Unity istnieją dwa zasadnicze rodzaje zderzaczy (ang. *collider*): podstawowe (ang. *primitives*) i siatkowe (ang. *meshes*). Kształty podstawowe w środowisku 3D są prostymi geometrycznymi obiektami, takimi jak skrzynki (ang. *box*), kule (ang. *sphere*) i kapsuły (ang. *capsule*).

Idea pracy w środowisku Unity opiera się na wykorzystaniu komponentów (ang. *components*). Komponent składa się ze zmiennych (ang. *variables*), które opisują jego właściwości lub zawierają ustawienia pozwalające nim sterować. Dzięki modyfikacji zmiennych użytkownik może mieć pełną kontrolę nad tym, jaki wpływ wywiera komponent na dany obiekt. Komponenty mogą występować pod różnymi postaciami. Mogą definiować zachowanie obiektu poprzez określanie wyglądu oraz wpływać na inne aspekty jego funkcjonowania w aplikacji. Dołączanie komponentów pozwala użytkownikowi na używanie określonych składników silnika gry w danym obiekcie.

Najczęściej używane komponenty związane z tworzeniem aplikacji są już wbudowane do systemu Unity. Są to na przykład komponent fizyki bryły sztywnej, a także elementy, takie jak światła, kamery, emiterzy cząstek. W środowisku Unity wykorzystuje się również prefabrykaty. Użytkownik może zastosować obiekty aplikacji, a także przechowywać je w postaci zasobów możliwych do ponownego użycia w różnych częściach aplikacji. Zasoby te mogą być konkretyzowane w dowolnym momencie.

## Wybrane zagadnienia dotyczące technik projektowania i programowania aplikacji interaktywnych

Istnieje wiele technik animacji i wiele koncepcji modelowania 3D<sup>8</sup>. Do popularnych technik animacji można zaliczyć:

- programowe modyfikowanie właściwości obiektów wizualnych przy użyciu pętli wykonawczej. Tę technikę wykorzystuje się w tworzeniu prostych animacji, takich jak obiekty obracające się wokół jednej osi. Technika jest również przydatna, gdy można wyrazić pozycję, orientację lub jakąś inną cechę obiektu jako funkcję zmiennej czasu,
- wykorzystanie międzyklatek (ang. *tween*) w celu uzyskania płynnego przejścia od jednej do drugiej wartości cechy. Ta technika jest wykorzystywana do tworzenia prostych jednorazowych efektów (np. przesuwania obiektów z jednego miejsca w inne po prostej linii),
- wykorzystanie klatek kluczowych. W tej sytuacji struktury danych w klatkach kluczowych reprezentują indywidualne wartości na osi czasu, a me-

---

<sup>8</sup> T. Parasi, *Aplikacje 3D...*

chanizm oblicza (interpoluje) wartości pośrednie w celu uzyskania płynnego efektu. Technika klatek kluczowych jest wykorzystywana do tworzenia prostego animowania przesunięć, obrotów i skalowania oraz animowania właściwości, takich jak kolory materiałów. Klatki kluczowe pozwalają na opracowywanie kilku przejść, w odróżnieniu od międzyklatek, które umożliwiają tworzenie pojedynczych przejść od jednej wartości do innej,

- animowanie obiektów po linii ścieżek – technika ta umożliwia tworzenie złożonych i realistycznych ruchów na podstawie wzorów i wcześniej utworzonych danych ścieżkowych,

- wykorzystanie celów morfingu (ang. *morph target*) do zdeformowania geometrii poprzez zmieszanie kilku różnych kształtów. Używa się takiej techniki do animowania mimiki oraz tworzenia bardzo prostych animacji postaci,

- zastosowanie skinningu w celu deformowania geometrii na podstawie animacji znajdującego się pod spodem szkieletu. Jest to preferowany i popularny sposób animowania postaci i innych złożonych kształtów.

W tworzonych aplikacjach bardzo często wykorzystuje się jedną albo też kilka technik animacji. Nie ma ścisłych reguł określających użycie danej techniki.

Dużą rolę w modelowaniu 3D odgrywa umiejętność tworzenia i wykorzystywania obiektów złożonych. Obiekty złożone powstają na skutek połączenia ze sobą dwóch lub większej liczby obiektów niezależnych. Znajdują one zastosowanie w zaawansowanych technikach modelowania oraz przy animowaniu pewnych efektów specjalnych, jak na przykład morfing<sup>9</sup>. Obiekty złożone dają bardzo dużo możliwości modelowania; za ich pomocą można bowiem deformować, przycinać, łączyć, usuwać, a nawet wytłaczać obiekty. Za ich pomocą można także tworzyć uporządkowane lub chaotyczne szyki obiektów powielonych, a także obiekty tektoniczne bazujące na liniach konturów.

W grafice 3D wykonuje się modelowanie obiektów za pomocą podstawowych operacji Boole'a, określanych także mianem operacji logicznych. Algebra Boole'a jest działem matematyki, którego twórcą był brytyjski profesor matematyki i logiki, George Boole, żyjący w latach 1815–1864. Algebrę Boole'a pierwotnie wykorzystywano do analizy zadań logicznych na zbiorach złożonych z dwóch elementów. W grafice 3D operacje Boole'a odnoszą się do zbiorów złożonych z płaszczyzn elementarnych lub splajnow. Operacje Boole'a kształtują dany obiekt poprzez dodawanie lub odejmowanie objętości oddziałujących na siebie obiektów. Operacje Boole'a działają tylko i wyłącznie na dwóch obiektach. Wynik końcowy operacji logicznej uzależniony jest od typu operacji oraz od usytuowania obiektów względem siebie. Najczęściej wykorzystywane typy operacji Boole'a to: dodawanie, odejmowanie, część wspólna.

---

<sup>9</sup> M. Matossian, *Po prostu 3ds max4*, Gliwice 2002.

Programy do tworzenia i modyfikowania grafiki 3D umożliwiają modelowanie za pomocą następujących metod: modelowanie bazujące na wielobokach lub siatkach, modelowanie bazujące na splajnach, modelowanie parametryczne, modelowanie za pomocą powierzchni sklejaných, modelowanie obiektów typu NURBS<sup>10</sup>. Każda z metod modelowania polega na innym manipulowaniu obiektami i ich strukturą. Niektóre typy obiektów łatwiej modelować za pomocą określonych metod, trudniej za pomocą innych.

Modelowanie bazujące na siatkach to metoda modelowania używana szczególnie w przypadku tworzenia obiektów, które są z natury zgeometryzowane. Przykładami takich obiektów są: budynki, proste postacie (np. postacie tworzone na użytek gier).

Modelowanie bazujące na splajnach to metoda modelowania, która doskonale się sprawdza w przypadku tworzenia obiektów posiadających profil lub kształt, który może zostać „wyciągnięty” lub wytłoczony. Przykładem takich obiektów są banany, butelki, słuchawki telefoniczne, kieliszki czy talerze.

Modelowanie parametryczne to metoda modelowania polegająca na wykorzystywaniu obiektów o predefiniowanych atrybutach, takich jak szerokość czy wysokość lub też modelowanie obiektów za pomocą modyfikatorów (które również mają charakter parametryczny). Obiekt jest uważany za parametryczny wówczas, jeśli w każdej chwili można powrócić do jego poprzedniej wersji i zmienić jego atrybuty. Oznacza to dokładną kontrolę nad obiektami przez cały czas ich edycji.

Wykorzystuje się także edycję wzorowaną<sup>11</sup>. Pierwszym krokiem prowadzącym do powstania modeli trójwymiarowych, jak postacie, budynki czy zwierzęta jest narysowanie ich na zwykłej kartce papieru. Uzyskany szkic przechodzi kolejne etapy obróbki, aż na końcu przyjmuje formę obrazu 2D zapisanego jako plik graficzny. Najczęściej przedstawiona na nim twarz bądź sylwetka ukazana jest dwójako – z profilu i widoku z przodu. Gwarantuje to przekazanie grafikowi komputerowemu niezbędnych informacji dotyczących wysokości i szerokości opracowywanego modelu, co pozwoli mu stworzyć wzorowany na ilustracji trójwymiarowy obiekt. W tym celu niezbędne jest uzyskanie w strefie 3D podglądu obrazu, stanowiącego swego rodzaju matrycę dla trójwymiarowych obiektów<sup>12</sup>.

Coraz częściej stosowana jest metoda modelowania określana jako rzeźbienie obiektów 3D. Wykorzystuje się ją, ponieważ pozwala tworzyć bardzo szczegółowe i realistyczne modele. W trójwymiarowej grafice komputerowej rzeźbienie jest metodą operowania kształtem obiektu w sposób podobny do tego, w jaki

---

<sup>10</sup> M. Peterson, *3D Studio Max dla każdego*, Gliwice 2000.

<sup>11</sup> K. Kukło, J. Kolmaga, *Blender...*

<sup>12</sup> T. Mullen, *Blender. Mistrzowskie animacje 3D*, Gliwice 2010.

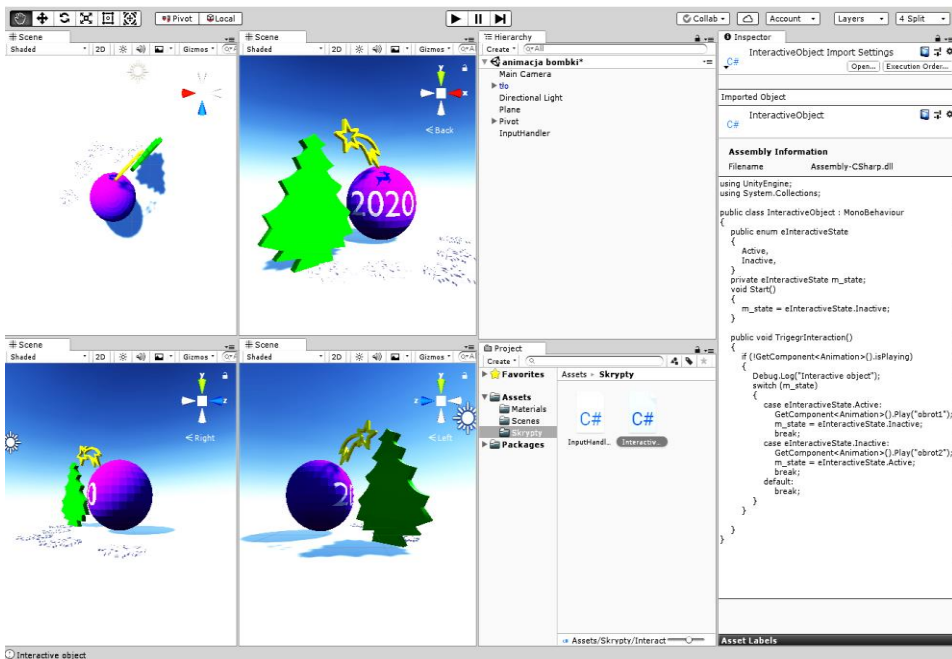


rzeźbiarz kształtuje glinę lub inny materiał plastyczny. Można w ten sposób modelować szczegółowe formy złożone. Wielu użytkowników twierdzi, że jest on bardziej naturalny niż bezpośrednie manipulowanie wierzchołkami i wielokątami siatek. Dlatego metody rzeźbiarskie są coraz popularniejsze również wśród zawodowych grafików komputerowych.

W środowisku Unity tworzy się skrypty, które uważane są za kluczowy składnik wykorzystywany w procesie tworzenia aplikacji interaktywnych.

Na rys. 3 przedstawiono środowisko Unity z obiektami zaimportowanymi z programu Blender oraz z elementami zaprojektowanymi w programie Inkscape.

Widoczny również jest skrypt napisany w języku C#.



**Rys. 3. Środowisko Unity. Obiekty zaimportowane z programu Blender z elementami zaprojektowanymi w programie Inkscape**

Źródło: opracowanie własne.

Aby użytkownik mógł dodawać do gry kolejne interaktywne składniki, wykorzystuje się możliwość pisania skryptów, które są rozpoznawane przez środowisko Unity jako komponenty. Skrypty mogą poszerzać lub modyfikować istniejącą funkcjonalność, możliwą do uzyskania w Unity, albo generować odpowiedni sposób zachowania się obiektu przy użyciu dostępnych klas skryptowych. Skrypty tworzy się w języku C Sharp i w środowisku Visual Studio.

## Podsumowanie

Tworzenie aplikacji interaktywnych jest wieloetapowym procesem, na który składają się: pomysł, rodzaj projektu graficznego, narzędzia i środowiska graficzne i programistyczne. Ostateczny efekt w procesie tworzenia aplikacji interaktywnych zależy między innymi od zestawu zastosowanych programów graficznych, które można wykorzystać. Zestaw Inkscape – Blender pozwala na pracę w otwartym formacie zapisu grafiki wektorowej SVG, który umożliwia import projektu do programu Blender. W programie 3ds Max użytkownik może wykorzystać pliki programu Adobe Illustrator CC. Wybór zależy od rodzaju tworzonego projektu i jego wykorzystania. Kluczową sprawą jest także użycie odpowiedniego środowiska programistycznego.

## Bibliografia

- Cieśla K., *Inkscape. Podstawowa obsługa program*, Gliwice 2013.  
Goldstone W., *Projektowanie gier w środowisku Unity*, Gliwice 2014.  
Kuklo K., Kolmaga J., *Blender. Kompendium*, Gliwice 2007.  
Matossian M., *Po prostu 3ds max4*, Gliwice 2002.  
Mullen T., *Blender. Mistrzowskie animacje 3D*, Gliwice 2010.  
Parasi T., *Aplikacje 3D. Przewodnik po HTML5, WebGL i CSS3*, Gliwice 2014.  
Peterson M., *3D Studio Max dla każdego*, Gliwice 2000.  
Simons B., *Blender. Praktyczny przewodnik po modelowaniu, rzeźbieniu i renderowaniu*, Gliwice 2014.  
Thorn A., *Praktyczne tworzenie gier Unity i Blender*, Gliwice 2015.

## Netografia

- [https://en.wikipedia.org/wiki/List\\_of\\_common\\_3D\\_test\\_models](https://en.wikipedia.org/wiki/List_of_common_3D_test_models)  
<https://freedesignfile.com/238238-tattoo-ornaments-design-material-vector-04/>  
<https://www.blender.org/foundation/>