

Создание прототипа ГИС-приложения с использованием Blender и Godot Engine

Development of GIS-application prototype using Blender and Godot Engine software

Бочкарев Л.М.

Студент, ФГБОУ ВО «МИРЭА - Российский технологический университет», г. Москва
e-mail: bochkarev.l.m@edu.mirea.ru

Bochkarev L.M.

Student, MIREA - Russian Technological University, Moscow
e-mail: bochkarev.l.m@edu.mirea.ru

Аннотация

В статье показан способ создания интерактивной трехмерной модели здания, используя ПО «Blender-3D» и «Godot Engine», показаны преимущества использования «Godot Engine» над «Unity» для создания интерактивной сцены на основе полученной модели. Показанный метод может быть полезен при решении различных задач в области трехмерного моделирования в программе «Blender-3D», при разработке приложений дополненной реальности и при создании интерактивных экскурсий.

Ключевые слова: трехмерное геоинформационное моделирование, трехмерное моделирование, трехмерная модель, Blender, Godot Engine.

Abstract

The article shows the method of creation of interactive three-dimensional model of building, with the means «Blender-3D» and «Godot Engine» software. The benefits of using «Godot Engine» over «Unity» when creating interactive scene based on previously made model are shown as well. The shown method is applicable for solving different three-dimensional modeling tasks in «Blender-3D», for developing augmented reality applications and for creation of interactive excursions.

Keywords: three-dimensional geoinformation modeling, three-dimensional modeling, three-dimensional model, Blender, Godot Engine.

Введение

В соответствии с используемой в РТУ МИРЭА оригинальной методической системой проведения занятий [1, 2, 3], на занятиях в курсе «Основы трехмерного моделирования» используется практико-ориентированное обучение (курсовой проект, в рамках которого создается прототип ГИС-приложения), студенты привлекаются к научно-исследовательской деятельности.

При выполнении курсового проекта для создания прототипа ГИС-приложения студентам предлагается использовать Unity3D [4]. Данный подход имеет следующие недостатки:

- относительно высокие системные требования;
- большой «вес» самого ПО (4 Гб);
- дорогая подписка.

Для решения перечисленных выше проблем при создании прототипа ГИС-приложения предлагается использовать платформу «Godot Engine». Она имеет ряд отличий от «Unity3D», некоторые из которых перечислены в табл. 1.

Сравнение Godot Engine и Unity

Характеристика	Godot Engine	Unity
Лицензия	Лицензия MIT	Проприетарная
Язык программирования	GScript, C#, C++	C#, C++
Минимальные системные требования		
ОС	Windows 7 SP1+, macOS 10.11+, Linux (какая-либо дистрибуция с ядром 2.6.18+)	Windows 7 SP1+, 8, 10, только 64-разрядные версии; Mac OS X 10.12+; Ubuntu 16.04, 18.04; CentOS 7
Процессор	64-битный двухъядерный процессор, тактовая частота 2.0 ГГц+	Любой 4-ядерный процессор начиная с 2011–2012 годов
Оперативная память	4 ГБ	4 ГБ
Графический процессор	с поддержкой OpenGL ES 3.0+	Видеокарта уровня GT1030 или встроенная графика уровня Intel HD Graphics 610
Свободное место	1 ГБ	4 ГБ

Создание прототипа ГИС-системы с помощью 3DF Zephyr/Blender3D/Godot 3D-моделирование зданий

Для создания 3D модели здания используются материалы съемки, полученные с помощью беспилотного летательного аппарата. Снимки обрабатываются с применением фотосканера 3DF Zephyr, где генерируется текстурированный меш, который затем импортируется в Blender для воспроизведения более точной геометрии зданий (рис. 1–2).

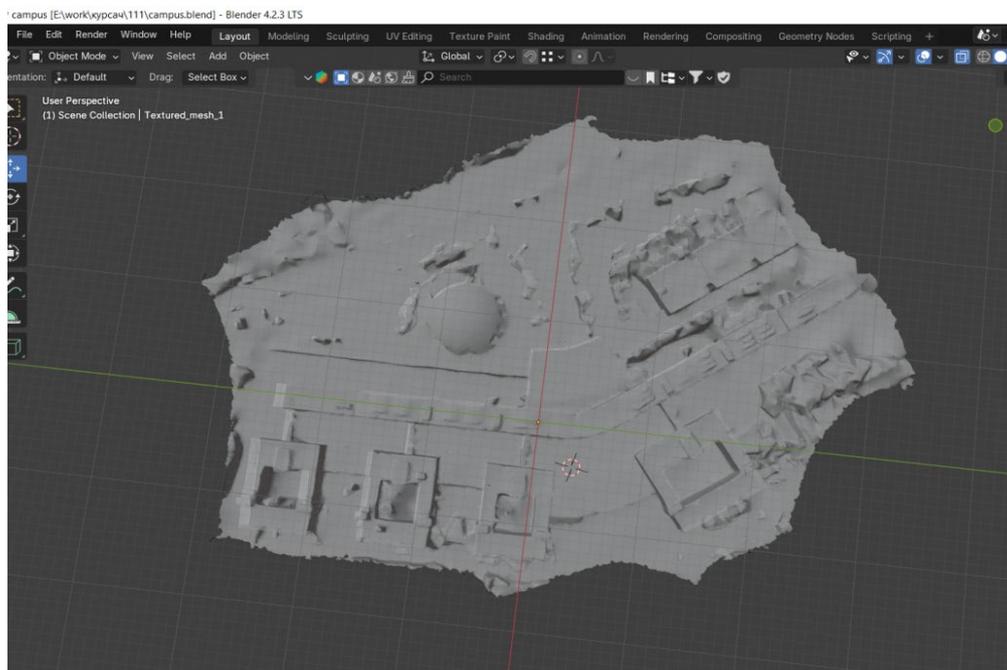


Рис. 1. Меш, импортированный в Blender 3D из 3DF Zephyr Free

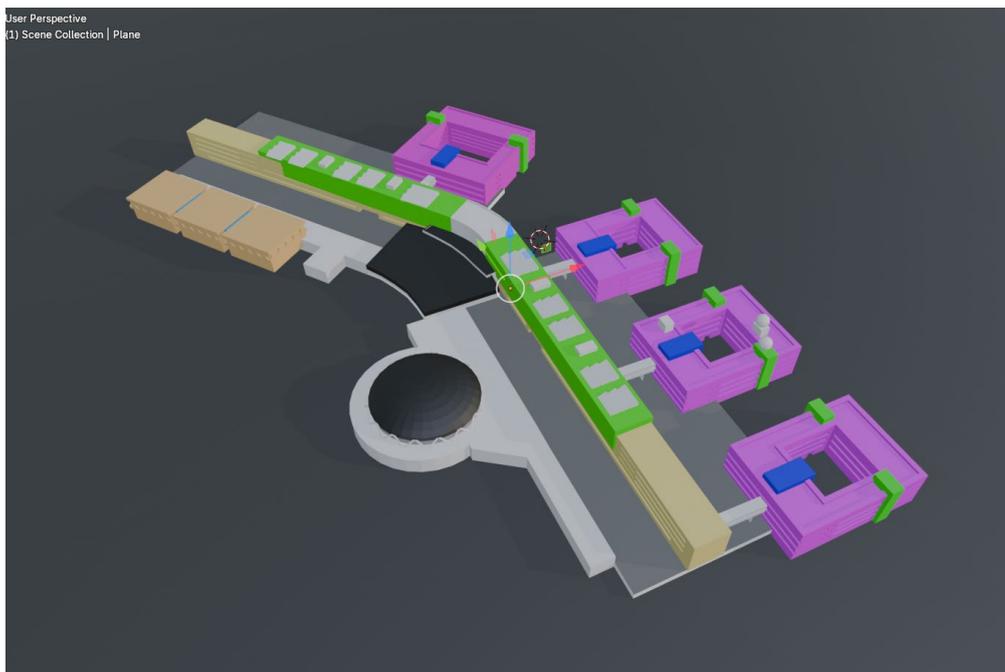


Рис. 2. 3D-модель зданий, созданная в редакторе Blender 3D на основе данных из 3DF Zephyr

После воссоздания геометрии требуется придать модели цвет и текстуру (рис. 3).

В настоящей работе для этого был использован бесплатный аддон Blender Kit [5], который позволяет загружать модели и текстуры из открытых библиотек. Также добавлены объекты обстановки (деревья и др.).

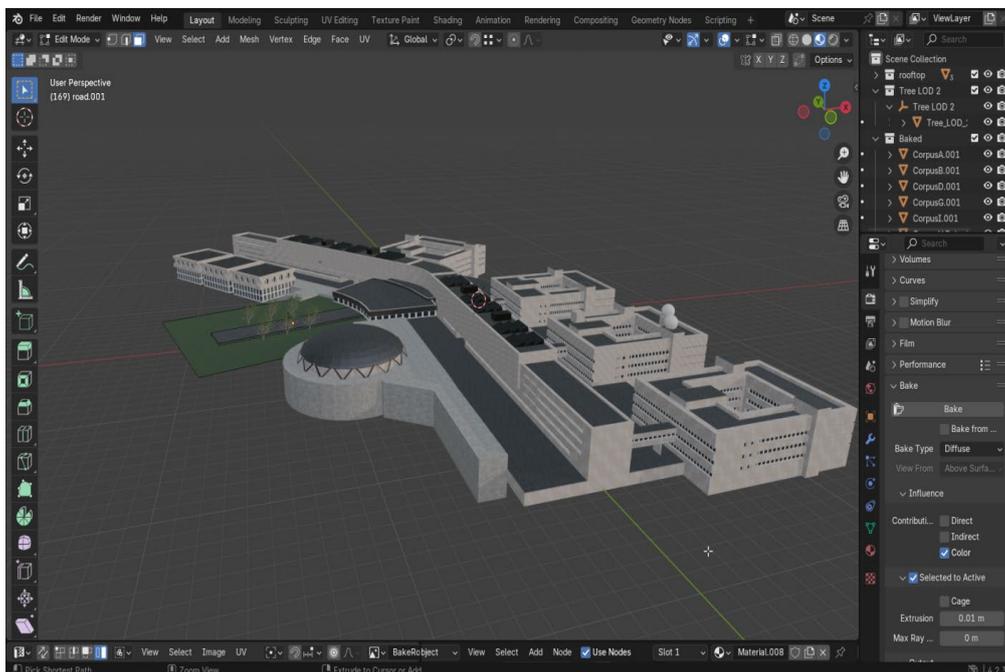


Рис. 3. Текстурированная модель зданий

Создание интерактивной среды в Godot Engine

Godot Engine — открытый кроссплатформенный двухмерный и трёхмерный игровой движок под лицензией MIT, который разрабатывается сообществом Godot Engine Community [6]. Среда позволяет разработчикам создавать игры с нуля, не пользуясь более никакими инструментами. Процесс программирования также не требует внешних инструментов.

В данной работе игровой движок использовался как среда для создания симулятора облёта кампуса на беспилотном летательном аппарате с функцией пояснений для некоторых

объектов и видом от третьего лица. Сохраненная сцена из Blender была импортирована из формата.blend (рис. 4).

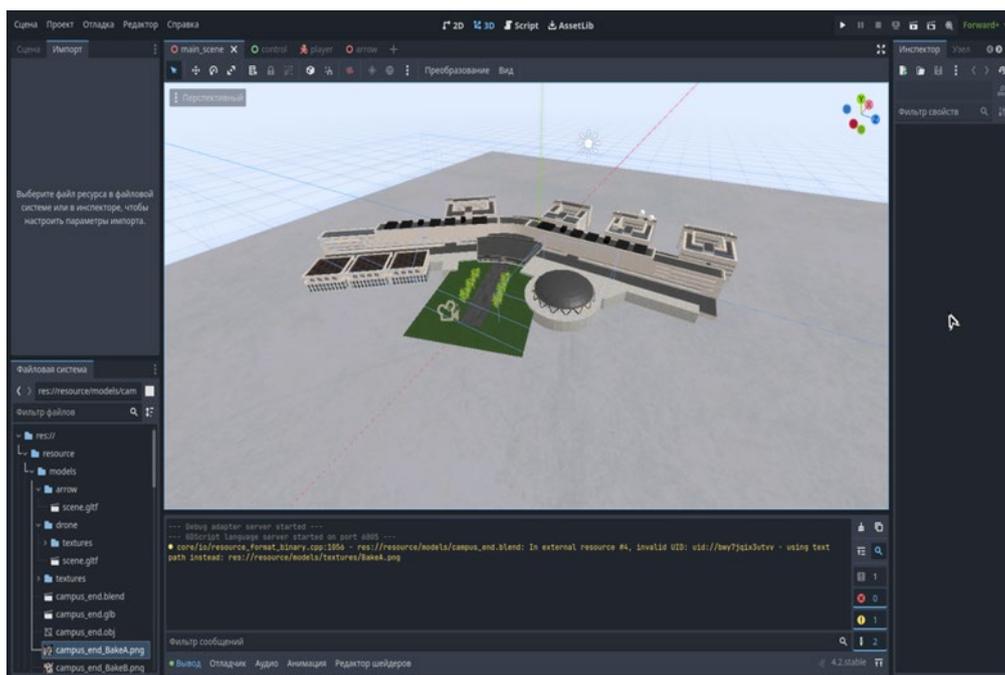


Рис. 4. Импортированная сцена в Godot

После импорта была создана отдельная сцена под названием «player» для создания «персонажа» пользователя. Модель БПЛА (рис. 5) была загружена из открытой библиотеки 3D моделей Sketchfab [7, 8].

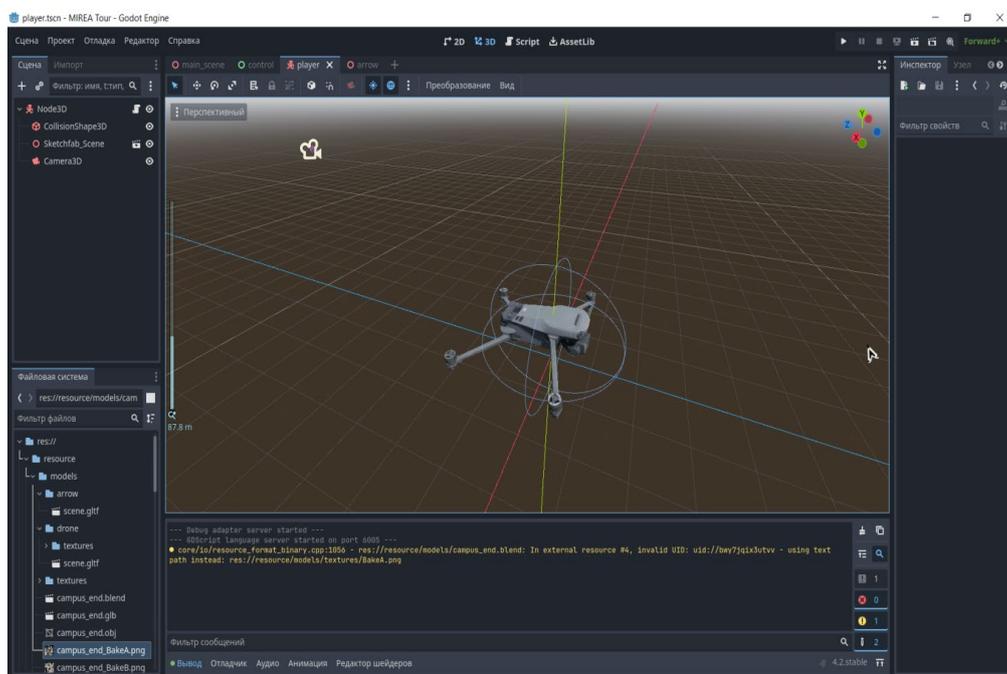


Рис. 5. Сцена с «персонажем» пользователя – беспилотником

Далее был создан пользовательский интерфейс для описаний корпусов (рис. 6). Для этого отдельно была создана сцена под названием «control» одноименного класса, в которой создавался весь интерфейс в режиме 2D, а также был написан скрипт для выгрузки информации о корпусах из заранее созданного JSON-файла.

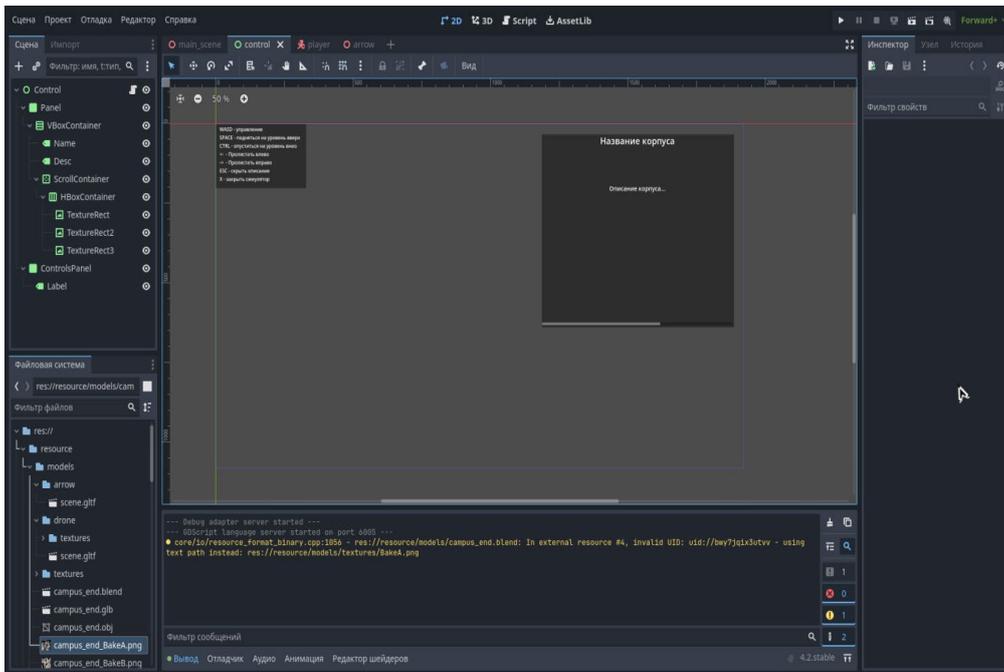


Рис. 6. Настройка пользовательского интерфейса в редакторе

Структура информации, изложенной в JSON файле (рис. 7), представляет из себя словарь, в котором ключами являются названия корпусов на английском языке (например, информация про корпус И имеет ключ «CorpusI»). Данные, связанные с ключом, представлены в виде массива из строк, в котором нулевой элемент – название корпуса на русском языке, первый элемент – описание корпуса, второй, третий и четвертый – ссылки на изображения в папке с ресурсами проекта. Итоговый вид JSON-файла представлен на рис. 9.

```

1  {"CorpusI": ["Корпус И",
2  "В этом корпусе располагается библиотека РТУ МИРЭА, музей РТУ МИРЭА, а также кафедра иностранных языков. Здесь часто можно встретить студентов, которые играют в настоль",
3  "res://resource/photos/CorpusI/photo1.jpg",
4  "res://resource/photos/CorpusI/photo2.jpg",
5  "res://resource/photos/CorpusI/photo3.jpg"
6  ]},
7  {"Turtle": ["Черепашка",
8  "Конгресс-центр или конференц-зал РТУ МИРЭА, который в простонародии называют 'Черепашка'. Долгое время находился на ремонте, но его открытие планируется уже в начале 20",
9  "res://resource/photos/Turtle/photo2.jpg",
10 "res://resource/photos/Turtle/photo3.png"
11 ]},
12 {"CorpusV": ["Корпус В",
13 "В этом корпусе находится Институт перспективных технологий и индустриального программирования! (ИПИП). Корпоративный цвет этого Института - желтый, а животное-талисма",
14 "res://resource/photos/CorpusV/photo1.jpg",
15 "res://resource/photos/CorpusV/photo2.jpg",
16 "res://resource/photos/CorpusV/photo3.jpg"
17 ]},
18 {"CorpusB": ["Корпус В",
19 "В этом корпусе базируется Институт радиоэлектроники и информатики. Корпоративный цвет этого Института - фиолетовый, а тоемное животное летучая мышь! Студенты этого Ин",
20 "res://resource/photos/CorpusB/photo1.jpg",
21 "res://resource/photos/CorpusB/photo2.jpg",
22 "res://resource/photos/CorpusB/photo3.jpg"
23 ]},
24 {"CorpusD": ["Корпус Д",
25 "Корпус Д является ректорским корпусом. Здесь находятся кабинеты большей части представителей администрации Университета, а также зал заседаний, в котором нередко прохо",
26 "res://resource/photos/CorpusD/photo1.jpg",
27 "res://resource/photos/CorpusD/photo2.jpg",
28 "res://resource/photos/CorpusD/photo3.jpg"
29 ]},
30 {"CorpusG": ["Корпус Г",
31 "В этом корпусе находится Институт искусственного интеллекта! (ИИИ или ИИВ - от 'жибретника'). Корпоративный цвет этого Института - зеленый, а талисман - робот! Студен",
32 "res://resource/photos/CorpusG/photo3.jpg"
33 ]},
34 {"CorpusI": ["Корпус И",
35 "res://resource/photos/CorpusI/photo1.jpg",
36 "res://resource/photos/CorpusI/photo2.jpg",
37 "res://resource/photos/CorpusI/photo3.jpg"
38 ]}

```

Рис. 7. JSON-файл с информацией

В главной сцене для каждого корпуса были добавлены объекты класса Area3D, которые реагируют на ввод и вывод объектов из их зоны обнаружения (рис. 8).

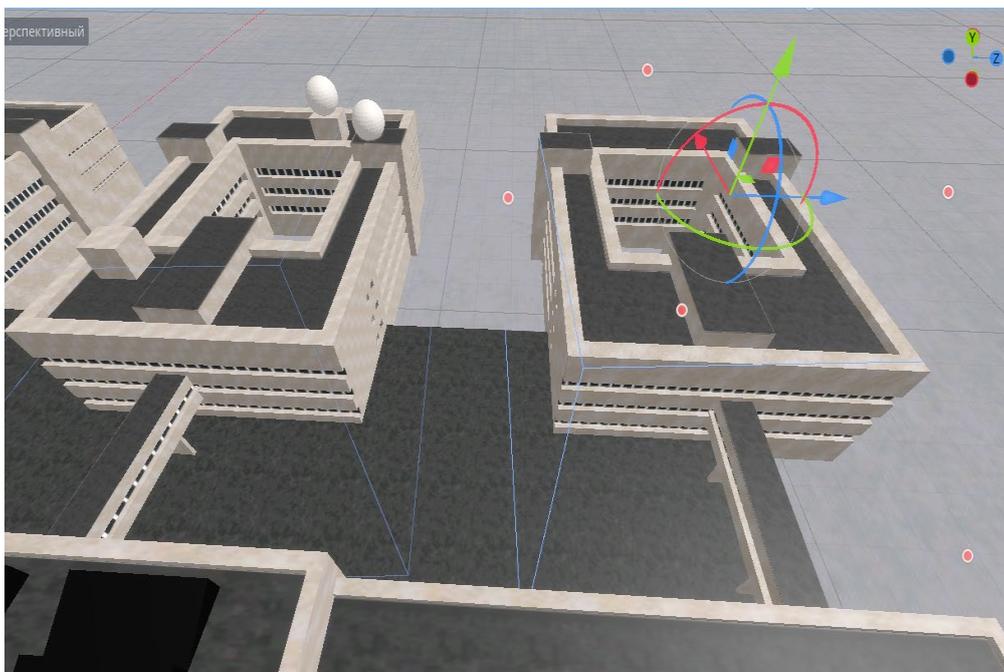


Рис. 8. Объекты класса Area3D (отображены синими рамками)

Вместе с зонами обнаружения были также добавлены модели стрелок, которые становятся видимыми для пользователя после взаимодействия с соответствующим объектом Area3D (рис. 9).

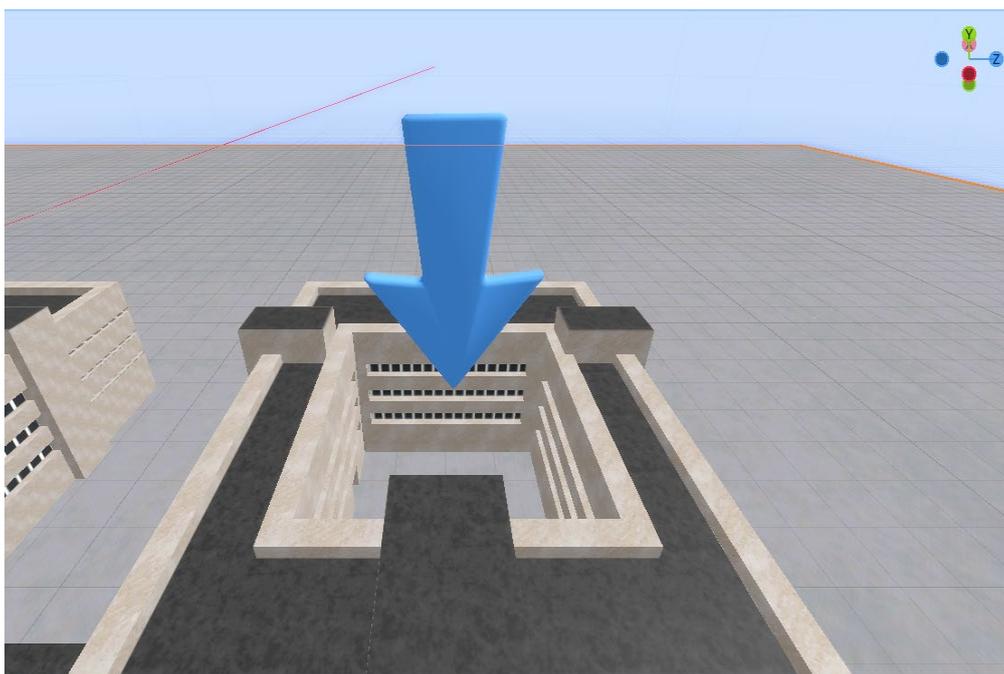


Рис. 9. Отображение стрелки

Далее к каждому объекту класса Area3D был прикреплен скрипт, реализующий основной функционал обновления интерфейса и загрузки информации о соответствующем корпусе. Код скрипта был написан на языке GDScript и представлен на рис. 10.

Работающее приложение показано на рис. 11.

```

1 extends Area3D #Наследование от класса Area3D
2
3 const filePath = "res://resource/text.json" #Путь до JSON файла с информацией
4
5 var nameCorp = "Название" #Переменная для поля с названием корпуса
6 var desc = "Описание" #Переменная для поля с описанием
7 var links = ["Ссылка1", "Ссылка2", "Ссылка3"] #Массив ссылок на изображения
8
9 func _ready(): #Функция, вызываемая после загрузки объекта
10     > "$../Arrow".set_visible(false) #Отключение отображения стрелки
11     > "$../Control"._on_pressed() #Функция, скрывающая интерфейс
12     > monitoring = true #Включение мониторинга объектов
13
14 func _on_body_entered(body): #Ф-я, вызыв. после обнаружения объекта
15     > "$../Arrow".set_visible(true) #Включение отображения стрелки
16     > var dataFile = FileAccess.open(filePath,FileAccess.READ) #Открытие JSON файла
17     > var parseResult = JSON.parse_string(dataFile.get_as_text()) #Считывание информации
18     > dataFile.close() #Закрытие файла
19     > nameCorp = parseResult[name][0] #Загрузка информации соответствующих полей
20     > desc = parseResult[name][1]
21     > links[0] = parseResult[name][2]
22     > links[1] = parseResult[name][3]
23     > links[2] = parseResult[name][4]
24     > "$../Control".updatePanel(nameCorp, desc, links) #Обновление интерфейса
25
26 func _on_body_exited(body): #Выход объекта
27     > "$../Arrow".set_visible(false) #Отключение отображения стрелки

```

Рис. 10. Скрипт зоны обнаружения

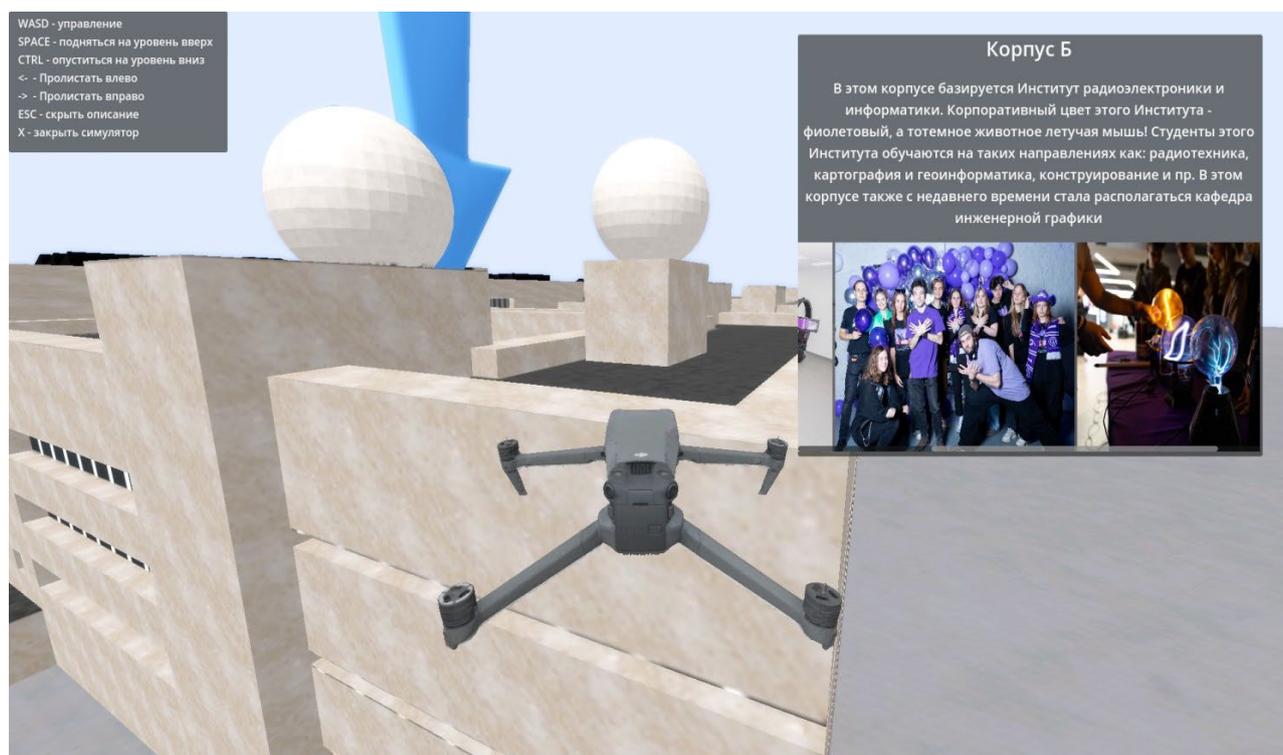


Рис. 11. Снимок экрана работающего приложения

Выводы

В данной работе показаны преимущества использования ПО «Blender3D» и игрового движка «Godot Engine» при создании интерактивных информационных систем, в частности геоинформационных систем. Предлагаемый подход позволяет создавать ГИС-модели зданий или сооружений с внедрением интерактивных элементов. Полученные таким образом продукты могут быть использованы при создании виртуальных экскурсий и 3D карт.

Литература

1. Вышнепольский В.И., Бойков А.А., Егиазарян К.Т., Кадыкова Н.С. Методическая система проведения занятий на кафедре «Инженерная графика» РТУ МИРЭА // Геометрия и графика. 2023. №. 1. С. 23-34. DOI: 10.12737/2308-4898-2023-11-1-23-34.
2. Вышнепольский В.И., Бойков А.А., Ефремов А.В., Кадыкова Н.С. Организация практико-ориентированного обучения на кафедре «Инженерная графика» РТУ МИРЭА // Геометрия и графика. 2023. №. 1. С. 35-43. DOI: 10.12737/2308-4898-2023-11-1-35-43.
3. Вышнепольский В.И., Бойков А.А., Егиазарян К.Т., Ефремов А.В. Научноисследовательская работа на кафедре «Инженерная графика» РТУ МИРЭА // Геометрия и графика. 2023. №. 1. С. 70-85. DOI: 10.12737/2308-4898-2023-11-1-70-85.
4. Шевченко О.Т. Создание интерактивной трехмерной ГИС-модели окрестности маяка // Журнал естественнонаучных исследований. — 2024. — Т. 7, №. 2. — С. 7–12.
5. BlenderKit. — Текст: электронный // BlenderKit: [сайт]. — URL: <https://www.blenderkit.com/> (дата обращения: 15.05.2025).
6. Руководство по работе с Godot Engine. — Текст: электронный // Godot Engine: [сайт]. — URL: <https://docs.godotengine.org> (дата обращения: 15.05.2025).
7. The leading platform for 3D & AR on the web. – Текст: электронный // Sketchfab: [сайт]. — URL: <https://sketchfab.com> (дата обращения: 15.05.2025).
8. DJI Mavic3 Classic DJI drone. – Текст: электронный // Sketchfab: [сайт]. — URL: <https://sketchfab.com/3d-models/dji-mavic3-classic-dji-drone-492473d005e54d628e0c0124871a98c6> (дата обращения: 15.05.2025).
9. The leading platform for 3D & AR on the web // Sketchfab [site]. URL: <https://sketchfab.com> (date of access: 15.05.2025).
10. DJI Mavic3 Classic DJI drone // Sketchfab [site]. URL: <https://sketchfab.com/3d-models/dji-mavic3-classic-dji-drone-492473d005e54d628e0c0124871a98c6> (date of access: 15.05.2025).