МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ «ВИТЕБСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ П.М. МАШЕРОВА»

Факультет математики и информационных технологий

Кафедра прикладного и системного программирования

Отчет о прохождении преддипломной практики

**РАЗРАБОТКА VR ПРИЛОЖЕНИЯ В СРЕДЕ UNITY**

Студент гр. 44 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Д. Царенко

подпись, дата

Руководитель от вуза,  
старший преподаватель кафедры ПиСП \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Новый

подпись, дата

Руководитель от предприятия,  
инженер-программист кафедры ПиСП \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.Ю. Пудовкин

подпись, дата

Витебск, 2024

**РЕФЕРАТ**

Отчет по практике 18 с., 6 источников, 7 рисунков

UNITY, VR, 3D-МОДЕЛЬ, NODEJS

Объект исследования – VR приложение для изучения органов человека.

Предмет исследования – написание VR приложения для изучения и взаимодействия с органами человека.

Цель практики – исследование процесса создания VR приложений на движке Unity, создание VR приложения для изучения органов.

Методы исследования: анализ, синтез, изучение литературы, практическая реализация.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc159939270)

[1. ВОЗМОЖНОСТИ И ОСОБЕННОСТИ ПРИЛОЖЕНИЯ 6](#_Toc159939271)

[1.1 Краткая характеристика средства реализации 6](#_Toc159939272)

[1.2 Возможности и особенности приложения 7](#_Toc159939273)

[2. ВЗАИМОДЕЙСТВИЕ ОРГАНОВ С ПОЛЬЗОВАТЕЛЕМ 9](#_Toc159939274)

[2.1 Виды взаимодействия с виртуальными предметами 9](#_Toc159939275)

[2.2 Реализация взаимодействия органов с пользователем 10](#_Toc159939276)

[2.3 Добавление звуков при взаимодействии с органами 13](#_Toc159939277)

[3. ДИПЛОМНАЯ СОСТАВЛЯЮЩАЯ 15](#_Toc159939278)

[ЗАКЛЮЧЕНИЕ 17](#_Toc159939279)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 18](#_Toc159939280)

# ВВЕДЕНИЕ

Данный отчет подготовлен по результатам прохождения преддипломной практики на кафедре прикладного и системного программирования ВГУ имени П.М. Машерова. В процессе прохождения преддипломной практики были изучены возможности Unity для разработки VR приложений, а также разработано приложения для изучения органов человека. В рамках выполнения дипломной составляющей практики были изучены основные возможности платформы NodeJS для разработки серверной части приложения для мониторинга активности пользователей.

Виртуальная реальность представляет собой некое подобие окружающего нас мира, искусственно созданного с помощью технических средств и представленного в цифровой форме. Создаваемые эффекты проецируются на сознание человека и позволяют испытывать ощущения, максимально приближенные к реальным [1].

Отличительные особенности виртуальной реальности:

* Формирует виртуальный мир;
* Создает эффект присутствия в ином пространстве;
* Обеспечивает максимальный уровень правдоподобия происходящего.

VR приложения предлагают широкий спектр возможностей, позволяющих пользователям погрузиться в иммерсивные (иммерсивность (от англ. immersive — «присутствие, погружение») — это способ восприятия, создающий эффект погружения в искусственно созданную среду [1]) виртуальные миры. VR приложения могут быть использованы в образовательных целях для создания интерактивных и реалистичных сред для обучения студентов. Они могут предлагать виртуальные лаборатории, исторические события, экскурсии по музеям, медицинское обучение и многое другое.

Объектом исследования являются VR приложения. Предметом исследования является разработка VR приложения для взаимодействия и изучения органов человека. Таким образом целью прохождения практики является исследование возможностей Unity для разработки VR приложений и практическая реализация VR приложения для взаимодействия с органами человека.

Для достижения поставленной цели необходимо было выполнить следующие задачи:

1. Изучить теоретические основы создания VR приложений на движке Unity.
2. Подготовка 3D-моделей органов и человека.
3. Организация взаимодействия пользователя с органами.

# ВОЗМОЖНОСТИ И ОСОБЕННОСТИ ПРИЛОЖЕНИЯ

## **Краткая характеристика средства реализации**

Unity – это кроссплатформенный и интегрированный игровой движок и среда разработки, разработанный компанией Unity Technologies. Он предоставляет разработчикам мощные инструменты для создания игр, виртуальной реальности (VR), дополненной реальности (AR) и других интерактивных приложений [2].

Основные характеристики и возможности Unity:

* Визуальный редактор: Unity предоставляет интуитивно понятный визуальный редактор, который позволяет разработчикам создавать сцены, объекты, анимацию, эффекты и другие элементы игры без необходимости писать код. Визуальный редактор также поддерживает функцию перетаскивания и масштабирования, что упрощает процесс создания объектов и компонентов;
* Скриптинг и программирование: Unity поддерживает несколько языков программирования, включая C#, UnityScript (базируется на JavaScript) и др. Разработчики могут использовать эти языки для создания скриптов, контролирующих поведение объектов, реализации игровой логики, обработки пользовательского ввода и многого другого;
* Рендеринг и графика: Unity оснащен мощным движком рендеринга, который обеспечивает высококачественную графику и визуализацию. Он поддерживает различные техники освещения, шейдеры, эффекты частиц, постобработку и другие возможности для создания уникального визуального стиля;
* Физика и коллизии: Unity включает встроенную систему физики, которая позволяет имитировать реалистическое поведение объектов в приложении. Разработчики могут определить физические свойства объектов, такие как масса, трение и силы, и использовать физические движки для достижения реалистичной симуляции столкновений, гравитации и других физических эффектов;
* Мультиплатформенность: Unity позволяет разрабатывать приложения для различных платформ, включая компьютеры, мобильные устройства, а также виртуальную реальность (VR) и дополненную реальность (AR).

Unity предоставляет мощные инструменты для разработки VR приложений, что особенно полезны для разработки нашего приложения. Unity поддерживает интеграцию с VR устройствами, позволяет использовать сенсоры виртуальной реальности, такие как трекеры положения и контроллеры, предоставляет инструменты и рекомендации для оптимизации производительности VR-приложений.

## **1.2 Возможности и особенности приложения**

VR (виртуальная реальность) приложение для взаимодействия с органами человека может быть очень полезным инструментом для образования, медицинского обучения и исследований.

Разрабатываемое VR приложение предоставляет пользователю возможность исследовать и взаимодействовать с органами человеческого тела в виртуальном окружении. Пользователь может получить уникальный опыт, который расширяет понимание анатомии и функционирования человеческого организма.

Приложение может быть полезным для различных пользователей, включая студентов медицинских и биологических факультетов, преподавателей, исследователей, врачей и прочих специалистов в сфере здравоохранения. Оно помогает улучшить понимание анатомии и функционирования органов человека.

Основные особенности приложения:

1. Виртуальное окружение: приложение предлагает иммерсивное виртуальное окружение, в котором пользователь может свободно перемещаться и исследовать органы человека. Это создает реалистичную и увлекательную среду для изучения анатомии;
2. 3D модели органов: приложение содержит 3D модели различных органов человеческого тела, включая печень, легкие, почки, мозг и другие. Пользователь может подбирать, вращать и рассматривать органы со всех сторон, чтобы получить полное представление о их структуре и местоположении;
3. Интерактивное взаимодействие: пользователь может взаимодействовать с органами, используя контроллеры VR. Это помогает лучше понять функционирование органов и их взаимосвязь;
4. Информационные аннотации: в приложении предоставляется подробная информация о каждом органе, его функциях и расположении. Пользователь может получить доступ к текстовым аннотациям, чтобы расширить свои знания об анатомии и медицине;
5. Окружение: приложение содержит несколько комнат: основная, в которой расположен макет человека с органами, а также смотровая, в которой органы представлены в увеличенном масштабе.

Основная комната содержит следующие компоненты:

1. Модель человека с органами внутри. Органы можно доставать из человека, рассматривать их и помещать обратно.
2. Платформа для размещения органов. При размещении органа на платформе он начинает вращаться, а рядом с платформой всплывает меню с подробным описанием органа.
3. Меню для взаимодействия с пользователем.

Смотровая комната содержит следующие элементы:

1. Платформа для размещения увеличенных моделей органов.
2. Меню выбора органов.

# 2. ВЗАИМОДЕЙСТВИЕ ОРГАНОВ С ПОЛЬЗОВАТЕЛЕМ

## **2.1 Виды взаимодействия с виртуальными предметами**

Виртуальный предмет – это объект или предмет, созданный и существующий в виртуальной среде, такой как виртуальная реальность (VR) или компьютерная графика. В отличие от реальных предметов, виртуальные предметы существуют только в цифровой форме и представляются с помощью данных и моделей.

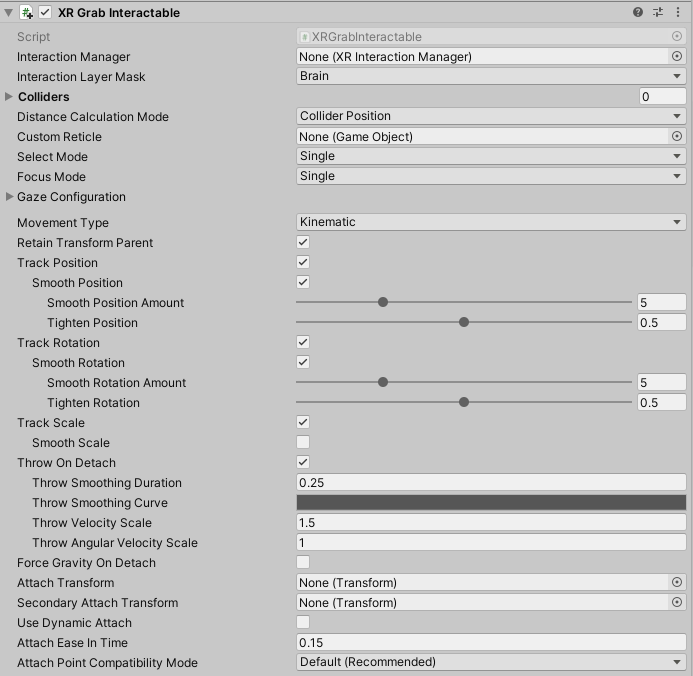
VR приложения могут предоставлять различные способы взаимодействия с виртуальными предметами, что делает опыт более реалистичным и иммерсивным. Основные способы взаимодействия с предметами в VR:

1. Охватывающие контроллеры: виртуальная реальность может использовать специальные контроллеры, которые позволяют пользователю схватывать, перемещать и выполнять различные действия с виртуальными предметами;
2. Жесты и движения тела: некоторые VR приложения позволяют взаимодействовать с предметами с помощью жестов и движений тела. Например, имеется возможность использовать руки и пальцы для схватывания, бросания или вращения предметов. Технологии отслеживания движений могут обнаруживать и интерпретировать жесты, чтобы предметы вел себя соответствующим образом;
3. Физическое моделирование: виртуальные предметы в VR приложениях могут иметь физические свойства, такие как масса, размер, форма и текучесть. Имеется возможность схватывать предметы, бросать их, сталкивать и взаимодействовать с ними, и они будут реагировать соответствующим образом;
4. Виртуальные интерфейсы: в VR приложениях могут быть виртуальные интерфейсы, такие как панели инструментов, кнопки или меню, которые позволяют манипулировать предметами или выполнять определенные действия. Например, можно использовать виртуальное меню для выбора предметов или для перемещения между сценами.

Это лишь некоторые примеры способов взаимодействия с предметами в VR. Технологии VR постоянно развиваются, и новые инновации позволяют создавать все более реалистичные и полноценные виртуальные миры, где пользователи могут взаимодействовать с предметами так же, как в реальном мире.

## **2.2 Реализация взаимодействия органов с пользователем**

Взаимодействие органов с пользователем будет происходить с помощью VR контроллеров. Для этого используется компонент XR Grab Interactable [3].

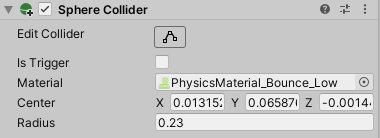


*Рисунок 2.1 – компонент XR Grab Interactable*

Основными настройками данного компонента являются:

1. Interaction Layer Mask – задает маску взаимодействия с органом. Будет использоваться в сокетах для ограничения позиций органов;
2. Smooth Position и Smooth Rotation – позволяет вращать органы, которые находятся в руке;
3. Attach Transform – при необходимости позволяет установить точку соприкасания органа с контроллером пользователя.

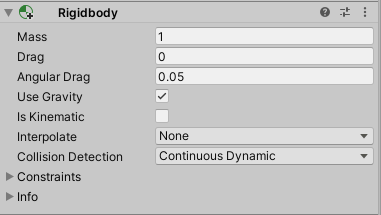
Для взаимодействия органов с окружающим миром используется компонент Collider.



*Рисунок 2.2 – компонент Collider*

Данный компонент позволяет задавать физические размеры органа (сама модель органа с окружающим миром не взаимодействует). Существует множество видов коллайдеров: сферический, кубический и т.д. Также присутствует возможность комбинировать несколько видов коллайдеров для одного органа.

Следующим важным компонентом является Rigidbody.

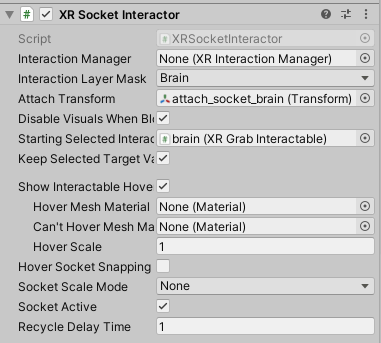


*Рисунок 2.3 – компонент Rigidbody*

Компонент Rigidbody отвечает за физические характеристики органа. Данный компонент позволяет регулировать массу предмета, применение к нему гравитации, а также тип коллизии с окружающим миром.

Так как органы в теле человека находятся на определенных позициях, то в приложении должна быть реализована возможность размещать органы в строго определенном месте. Для этого используется специальный объект – сокет. В VR приложениях сокеты могут использоваться для точного размещения моделей в определенных местах или на определенных поверхностях.

Для создания сокета служит компонент XR Socket Interactor [4].

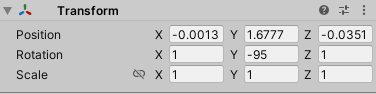


*Рисунок 2.4 – компонент XR Socket Interactor*

Основными настройками данного компонента являются следующие: Interaction Layer Mask – используется для задания органа, для которого данный сокет предназначен, Attach Transform – задает точку, в которую будет помещен орган.

Также необходимым компонентом для сокета является Collider, для которого флажок Is Trigger ставится в значение «истина». В этом случае Collider задает зону, при попадании в которую орган будет помещен в сокет.

Компонент Transform задает ориентацию (углы поворота вдоль осей и т.д.) органа в сокете.



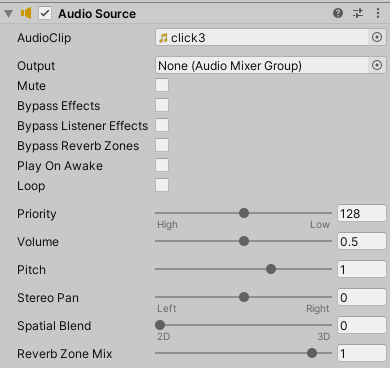
*Рисунок 2.5 – компонент Transform*

## **2.3 Добавление звуков при взаимодействии с органами**

В Unity звуки используются для создания аудиоэффектов и музыки.

Первым шагом является импортирование звуковых файлов в проект Unity. Поддерживаемые форматы включают WAV, MP3, OGG и другие. В нашем случае нужны звуки для поднятия и отпускания органа.

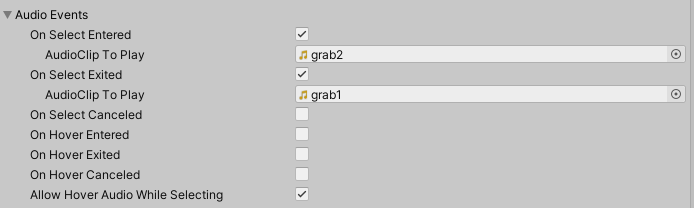
В Unity для воспроизведения звуковых файлов используются компоненты звуковые источники (Audio Source).



*Рисунок 2.6 – компонент Audio Source*

В компоненте Audio Source указывается источник звука (файл с расширением mp3), а также можно настроить различные параметры звука, такие как громкость (Volume), темп (Pitch), пространственный звук (Spatial Blend) и т.д. Также можно указать, должен ли звук повторяться (Loop), как долго он должен длиться и другие параметры воспроизведения. Также можно задать параметр Play On Awake, при включении которого звук будет проигрываться при входе в приложении (используется для фоновой музыки).

В VR проекте присутствует объект XR Rig, который представляет собой пользователя в пространстве. У его дочерних объектов LeftHand Controller и RightHand Controller (левый и правый контроллеры соответственно) присутствует компонент XR Ray Interactor, в котором можно указать источник звука при поднятии или отпускании предмета.



*Рисунок 2.7 – компонент Audio Events для контроллеров*

# 3. ДИПЛОМНАЯ СОСТАВЛЯЮЩАЯ

Node.js представляет среду выполнения кода на JavaScript, которая построена на основе движка JavaScript Chrome V8, который позволяет транслировать вызовы на языке JavaScript в машинный код. Node.js прежде всего предназначен для создания серверных приложений на языке JavaScript [5].

Особенностью Node.js является то, что Node.js использует один (основной) поток, который получает все запросы и управляет ими через очередь запросов (таким образом, Node.js является однопоточным сервером). Внутри этого потока выполняется так называемый цикл событий (event loop), который представляет собой цикл, который непрерывно проверяет запросы из очереди событий и обрабатывает события ввода и вывода.

При написании серверной части приложения, предназначенного для отслеживания входа и выхода пользователей из операционной системы, будем использовать фреймворк для NodeJS – NestJS.

NestJS – это фреймворк для создания эффективных, масштабируемых серверных приложений на Node.js. Он использует прогрессивный JavaScript, хотя Nest создан с помощью TypeScript и полностью его поддерживает (но все же позволяет разработчикам писать код на чистом Javascript) и сочетает в себе элементы ООП (объектно-ориентированного программирования), ФП (функционального программирования) и ФРП (функционально-реактивного программирования) [6].

Особенности NestJS:

1. Использование TypeScript: NestJS полностью основан на TypeScript, что позволяет разработчикам использовать преимущества статической типизации, улучшенной поддержки IDE и других возможностей языка TypeScript. Это способствует созданию более надежного и чистого кода, а также облегчает процесс разработки и поддержки приложений;
2. Модульная архитектура: NestJS поддерживает модульную архитектуру, которая позволяет разбить приложение на независимые и переиспользуемые модули. Это упрощает организацию кода, улучшает его читаемость и поддержку. Каждый модуль может содержать свои контроллеры, провайдеры и другие компоненты, что способствует логическому разделению функциональности приложения;
3. Расширяемость: NestJS предоставляет возможность интеграции с другими библиотеками и фреймворками, такими как Express.js, Fastify и другими. Это позволяет разработчикам использовать существующий экосистемный инструментарий и библиотеки, что облегчает разработку и ускоряет процесс создания приложений.

При разработке backend части приложения для мониторинга активности пользователей будет использоваться TypeORM (ORM (Object-Relational Mapping) – это технология, которая обеспечивает связь и взаимодействие между приложением и базой данных; позволяет разработчикам работать с данными в базе данных, используя объекты и методы, а не низкоуровневые запросы).

Основные используемые классы:

1. Entity (Сущность) – это класс, который представляет таблицу в базе данных, и его свойства соответствуют полям этой таблицы. Сущности обычно декорируются декоратором @Entity() из пакета TypeORM;
2. Service (Сервис) – класс, который содержит бизнес-логику и выполняет определенные операции или функции в приложении. Сервисы являются основным местом для обработки бизнес-логики и взаимодействия с другими компонентами, такими как репозитории, внешние API или другие сервисы. Используют декоратор @Service();
3. Controller (Контроллер) - отвечает за обработку входящих HTTP-запросов и взаимодействие с клиентами. Контроллеры определяют маршруты, по которым приложение будет принимать запросы, и содержат методы, которые обрабатывают эти запросы.

Таким образом, были рассмотрены отличительные особенности NodeJS, в частности, его фреймворка – NestJS.

# ЗАКЛЮЧЕНИЕ

За время прохождения преддипломной практики были изучены возможности Unity для создания VR приложений. Также было реализовано приложение для изучения и взаимодействия с органами человека. В дипломной части практики были рассмотрены возможности платформы NodeJS для реализации серверной части приложения.

Все задачи, поставленные во время прохождения практики были выполнены.

В результате выполнения программы практики было разработано VR приложения для изучения органов человека. Данное приложение может быть полезным для различных пользователей, включая студентов медицинских и биологических факультетов, преподавателей, исследователей, врачей и прочих специалистов в сфере здравоохранения. Оно помогает улучшить понимание анатомии и функционирования органов человека.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Виртуальная реальность Virtual Reality (VR) – [Электронный ресурс] - Режим доступа: <https://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1> %82%D1%8C%D1%8F:%D0%92%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F\_%D1%80%D0%B5%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%8C\_(VR,\_Virtual\_Reality)#.D0.98.D0.BC.D0.BC.D0.B5.D1.80.D1.81.D0.B8.D0.B2.D0.BD.D1.8B.D0.B5\_.D1.82.D0.B5.D1.85.D0.BD.D0.BE.D0.BB.D0.BE.D0.B3.D0.B8.D0.B8 – Дата доступа: 25.02.2024
2. Движок Unity – особенности, преимущества и недостатки – [Электронный ресурс] – Режим доступа: <https://cubiq.ru/dvizhok-unity/> - Дата доступа: 25.02.2024
3. Grabbable Objects – [Электронный ресурс] – Режим доступа: [https://learn.unity.com/tutorial/grabbable-objects-1?uv=2020.3&courseId=60183276edbc2 a](https://learn.unity.com/tutorial/grabbable-objects-1?uv=2020.3&courseId=60183276edbc2%20a)2e6c4c7dae&projectId=60183335edbc2a2e6c4c7dcb# - Дата доступа: 25.02.2024
4. Sockets – [Электронный ресурс] – Режим доступа: <https://learn.unity.com/tutorial/sockets-g-1?uv=2020.3&courseId=60183276edbc2a2e6c4c> 7dae&projectId=60183335edbc2a2e6c4c7dcb – Дата доступа: 25.02.2024
5. Введение в NodeJS – [Электронный ресурс] – Режим доступа: <https://metanit.com/web/nodejs/1.1.php> - Дата доступа: 25.02.2024
6. Введение в NestJS – [Электронный ресурс] – Режим доступа: <https://nestjs.ru/> - Дата доступа: 25.02.2024