



ДИПЛОМНА РАБОТА

Тема:

„Визуализация на артикули във виртуална реалност“

РАЗРАБОТИЛ:

Име: Димитър Пламенов Иванов

Спец.: “Компютърни науки”

Фак. №: 2209013779

ОКС: Бакалавър

Форма на обучение: Редовна

НАУЧЕН РЪКОВОДИТЕЛ:

Име: доц. д-р Емилиян Георгиев
Петков

Катедра: Компютърни системи и
технологии

***Великотърновски университет “Св. Св.Кирил и Методий”
Факултет „Математика и информатика“
Катедра “Компютърни системи и технологии”***

ЗАДАНИЕ НА ДИПЛОМНА РАБОТА

Дипломант: Димитър Пламенов Иванов

Специалност: Компютърни науки, **Курс:** 2209013779, **Фак. №:** 2209013779

ОКС: бакалавър

Тема на дипломната работа: „Визуализация на артикули във виртуална реалност”.

Изследователски проблем: Изследване на подходи за създаване на виртуална реалност и визуализация на артикули във виртуална реалност.

Цел на дипломната работа: Изследване на възможности и реализация на виртуална среда с включени обекти, с които потребителят може да взаимодейства.

Задачи на дипломната работа:

- 1) Да се изследват различните начини, възможности и технологии за създаване на тримерни обекти;
- 2) Да се изследват възможностите на технологиите и програмите за създаване на среда във виртуална реалност;
- 3) Да се изследват възможностите на човешко-машинни интерфейси за взаимодействие на потребителя с виртуалната реалност и обектите.

Основна литература за проучване:

1. Емилиян Петков. Основи на компютърната графика. Фабер, Велико Търново, 2013;
2. William R. Sherman. Understanding Virtual Reality. Morgan Kaufmann. California. 2018;
3. John F. Hughes. Computer graphics principles and practice. Addison-Wesley Professional. Providence, Rhode Island. 2013.

Препоръчителен обем на разработката: от 50 до 60 страници

Краен срок на предаване на дипломния проект: 06.09.2024 г.

Дипломант:.....

/Димитър Иванов/

Научен ръководител:.....

/доц. д-р Емилиян Петков/

Дата на възлагането: 02.02.2024 г.

Гр. Велико Търново

**ДИПЛОМНАТА РАБОТА Е ПРЕДАДЕНА В СРОК
И ОТГОВАРЯ НА ИЗИСКВАНИЯТА ЗА ДОПУСКАНЕ ДО ЗАЩИТА**

Дата на предаването: 02.09.2024 г.

Гр. Велико Търново

Научен ръководител:

/доц. д-р Емилиян Петков/

СЪДЪРЖАНИЕ

УВОД.....	10
ПЪРВА ГЛАВА	13
Основи на виртуалната реалност и тримерното моделиране	13
1.1.1. Координатни системи и 3D пространство	13
1.1.2. Полигони, върхове и мрежи	14
1.1.3. Текстуриране и материали	16
1.1.4. Осветление и засенчване	18
1.2. Техники за моделиране	20
1.2.1. Прimitives моделиране	20
1.2.2. Моделиране на полигони.....	21
1.2.3. NURBS (Non-Uniform Rational B-Splines) моделиране	21
1.2.4. Моделиране на повърхнини с подразделяне	21
1.2.5. Скулптурство. Дигитално извайване.....	22
1.2.6. Процедурно моделиране.....	22
1.3. Основи на виртуалната реалност	23
1.3.1. Определение за виртуална реалност и кратка история	23
1.3.2. Видове VR системи	24
1.3.2.1. Напълно потапяща VR системи	24
1.3.2.2. Полупотапящи VR системи	24
1.3.2.3. Непотапящи VR системи	24
1.3.2.4. Разширена реалност (Augmented Reality, AR) и смесена реалност (Mixed Reality, MR)	24
1.3.3. Основни компоненти на VR системите.....	25
1.3.3.1. Хардуерни компоненти	25
1.3.3.2. Софтуерни компоненти.....	25
1.4. Човешко-машинни интерфейси, използвани в системите за виртуална реалност.....	26
1.4.1. Дисплей, монтирани на главата (HMD)	26
1.4.2. Системи за следене на движението	27
1.4.2.1. Проследяване отвън-навътре.....	27
1.4.2.2. Проследяване отвътре-навън.....	27
1.4.3. Хаптични устройства и обратна връзка	28
1.4.4. Входни устройства	28

1.4.4.1.	Ръчни контролери	28
1.4.4.2.	Ръкавици	28
1.4.4.3.	Проследяване на очите	28
1.5.	Ергономичност на човешкото тяло, сетива и проблеми	29
1.5.1.	Визуално възприятие във VR.....	29
1.5.1.1.	Стереоскопия	29
1.5.1.2.	Полезрение (FOV)	30
1.5.1.3.	Разделителна способност и плътност на пикселите	30
1.5.1.4.	Честота на опресняване и плавност на движенията на обектите	30
1.5.2.	Пространствен звук	30
1.5.3.	Проблеми със загубата на равновесие, дезориентацията и комфорта	31
1.5.3.1.	Причини за нежелани странични ефекти от работата с VR	32
1.5.3.2.	Решения за справяне с такива проблеми	32
1.5.3.3.	Ергономични съображения при работа със системи за виртуална реалност... ..	32
1.5.3.4.	Продължителност на работа със система за виртуална реалност	32
1.5.3.5.	Съображения за достъпност от хора с увреждания	33
1.6.	Рендиране на графика в реално време за виртуална реалност	33
1.6.1.	Стъпки за обработка на графика и графични сцени	33
1.6.1.1.	Етап на начално процесиране на сцената (Application stage).....	34
1.6.1.2.	Обработка на геометрията (Geometry).....	35
1.6.1.3.	Растеризация	35
1.6.1.4.	Обработка на пиксели	35
1.6.1.5.	Пост-процесиране	35
1.6.1.6.	Специфични за VR съображения.....	35
1.6.2.	Техники за оптимизация	35
1.6.3.	Съображения, свързани с честотата на опреснение и времезабавянето	37
1.6.3.1.	Изисквания за броя на кадрите	37
1.6.3.2.	Съображения, свързани с времезабавянето	37
1.6.3.3.	Балансиране на качеството и производителността.....	37
1.7.	Създаване на 3D обекти и взаимодействие във VR	38
1.7.1.	Оптимизиране на 3D моделите за VR	38
1.7.1.1.	Оптимизиране на броя на полигоните	38
1.7.1.2.	Оптимизиране на текстурите	38

1.7.1.3.	Прилагане на вариращо ниво на детайлност(Level Of Detail, LOD)	38
1.7.1.4.	Изрязване на невидими обекти (Culling).....	38
1.7.1.5.	Предварително обработване („Изпичане“, Baking) на осветление в светлинни карти.....	39
1.7.2.	Ниво на детайлност (LOD) в среди за виртуална реалност	39
1.7.3.	Естествени потребителски интерфейси и разпознаване на жестове	39
1.8.	Присъствие и привличане на вниманието на потребителите във VR среда.....	40
1.8.1.	Фактори, влияещи върху присъствието	40
1.8.1.1.	Визуална достоверност	40
1.8.1.2.	Качество на звука	41
1.8.1.3.	Интерактивност	41
1.8.1.4.	Полезрение (Field Of View, FOV).....	41
1.8.1.5.	Честота на кадрите и закъснение.....	41
1.8.1.6.	Последователност на виртуалния свят	41
1.8.1.7.	Аватар и синхронизация с действията на потребителя	42
1.8.2.	Техники за подобряване задържането на вниманието на потребителя при работа във виртуалния свят	42
1.8.2.1.	Фотореалистично визуализиране	42
1.8.2.2.	Дизайн на пространствения звук	42
1.8.2.3.	Интерактивни обекти.....	43
1.8.2.4.	Референтни мащаби	43
1.8.2.5.	Разказвателни елементи	43
1.9.	Нови тенденции във VR за визуализация на мебели и архитектура	43
1.9.1.	Развитие на хаптичните устройства.....	43
1.9.2.	Облачно базирано процесизиране	44
1.9.3.	Фотограмметрия	44
1.9.4.	Сътрудничество с много потребители	44
ВТОРА ГЛАВА.....	45	
Преглед, анализ и сравнение на технологиите за създаване на виртуална реалност и тримерно моделиране	45	
2.1.	Хардуерни платформи за виртуална реалност	45
2.1.1.	Преглед на хардуерните опции за VR	46
2.1.1.1.	Системи за виртуална реалност от висок клас	46
2.1.1.1.1.	Oculus Rift S.....	47

2.1.1.1.2. HTC Vive Pro	48
2.1.1.1.3. HP Reverb G2.....	49
2.1.1.2. Самостоятелни VR хедсети	50
2.1.1.2.1. Meta Quest 2.....	50
2.1.1.2.2. Apple vision pro	51
2.1.1.3. VR системи, базирани на мобилни телефони	52
2.1.1.3.1. Google Cardboard	52
2.1.1.3.2. Samsung Gear VR.....	53
2.1.2. Сравнение на стандартен Хедсет за виртуална реалност с VR системи за мобилен телефон	54
2.1.3. Обосновка за избора на мобилен телефон и VR очила	56
2.2. Софтуер за разработване на VR приложения	57
2.2.1. Преглед на популярни 3D енджини за разработка на игри	57
2.2.1.1. Unity Engine.....	57
2.2.1.2. Unreal Engine.....	58
2.2.1.3. CryEngine	58
2.2.1.4. Анализ на Unity Engine.....	59
2.2.1.5. Сравнение На Unity Engine с Unreal Engine.....	59
2.2.1.6. Обосновка за избора на Unity със C#.....	61
2.2.2. Преглед на софтуери за 3D моделиране	62
2.2.2.1. CAD софтуер.....	62
2.2.2.2. Софтуер за BIM	63
2.2.2.2.1. Revit	63
2.2.2.2.2. ArchiCAD	63
2.2.2.3. Софтуер за 3D моделиране с общо предназначение	64
2.2.2.3.1. Blender	64
2.2.2.3.2. 3ds Max.....	64
2.2.2.4. Анализ на Blender.....	65
2.2.2.5. Сравнение на Blender с 3ds Max и AutoCAD.....	65
2.2.2.6. Обосновка на избора на Blender	67
2.3. Допълнителни технологии и езици за разработване на VR приложения	68
2.3.1. SDK и API за VR.....	68
2.3.1.1. OpenVR/SteamVR.....	68
2.3.1.2. Oculus SDK.....	68

2.3.1.3.	Google VR SDK.....	69
2.3.2.	3D файлови формати и стандарти	69
2.3.2.1.	FBX (Filmbox)	69
2.3.2.2.	glTF (GL Transmission Format)	69
2.3.2.3.	OBJ.....	70
2.3.2.4.	COLLADA	70
2.3.3.	Инструменти за 360-градусови снимки и видео	70
2.3.3.1.	360-градусови камери.....	70
2.3.3.2.	Софтуер за фотограметрия	70
2.3.3.3.	Софтуер за съшиване на изображения	71
2.3.4.	Инструменти за проектиране, специфични за VR.....	71
2.3.4.1.	Gravity Sketch	71
2.3.4.2.	Medium.....	72
2.4.	Прилики и разлики между видеоигрите и симулаторите	72
2.4.1.	Общи елементи в игрите и симулаторите.....	73
2.4.2.	Уникални аспекти на симулаторите	73
2.4.3.	Преодоляване на разликата между развлекателните и професионалните инструменти.....	74
ТРЕТА ГЛАВА		76
Проектиране и разработка на прототип на виртуална среда и обекти		76
3.1.	Описание на проблема и цели.....	76
3.1.1.	Описание на проблема, който се решава	76
3.1.2.	Основни цели и задачи на прототипа	77
3.2.	Фаза на проектиране	78
3.2.1.	Концептуален проект	78
3.2.2.	Дизайн на потребителското изживяване (UX)	79
3.2.3.	Дизайн на интерфейса	79
3.3.	Ключови решения и задачи	80
3.3.1.	Избрани технологии и инструменти.....	80
3.3.2.	Принципи и подходи за проектиране	81
3.4.	Средата за разработка Unity	82
3.4.1.	Преглед на Unity Hub	83
3.4.2.	Инсталиране на Unity	84
3.4.3.	Преглед на Microsoft Visual Studio и пакета Game development with Unity	85

3.4.4.	Преглед на C#	87
3.4.5.	Преглед на елементите на потребителския интерфейс на Unity	87
3.5.	Програмата за моделиране Blender	90
3.5.1.	Инсталиране на Blender	90
3.5.2.	Създаване на нов проект	91
3.5.3.	Основни елементи на Blender	91
3.6.	Структура на проекта	94
3.6.1.	Създаване и настройка на проект	94
3.6.2.	Проектиране на сцени	94
3.7.	Подробности на прототипа	95
3.7.1.	Дизайн на стаята – под, стени, таван	95
3.7.2.	Разполагане на предмети и мебели	96
3.7.3.	Създаване на контролер за потребителя	98
3.7.4.	Осветление	99
3.7.5.	Динамични текстури	99
3.7.6.	Аудио	100
3.7.7.	Движение на потребителя	100
3.8.	Тестване на приложението и получени резултати	101
3.9.	Затруднения по време на разработване на прототипа	103
3.9.1.	Несъвместимост с Oculus SDK	103
3.9.2.	Несъвместимост с игрови контролер за PlayStation 4	104
3.9.3.	Режим на работа на потребителя при реализирана с Google VR SDK функционалност	104
3.9.4.	Липса на функционалност за регулиране на височината	105
3.9.5.	Липса на функционалност за управление на менюта	106
3.10.	Бъдещи подобрения	106
ЗАКЛЮЧЕНИЕ	108
ИЗПОЛЗВАНА ЛИТЕРАТУРА	110

УВОД

Във времена на бърз технологичен напредък начинът, по който се визуализира и взаимодейства с 3D обекти особено в области като интериорния дизайн и архитектурата, тепърва започва да се усъвършенства. Традиционните методи за разглеждане на рендери на обекти или помещения на двуизмерни екрани често не успяват да предадат напълно пространствените връзки и интерактивния потенциал на обекти като мебели, подове, стени и тавани. Технологията за виртуална реалност (Virtual Reality, VR) предлага обещаващо решение на това предизвикателство, като се представя иновативен подход за визуализиране и взаимодействие с тези елементи в напълно поглъщащо триизмерно пространство.

Настоящата среда за визуализация на триизмерни обекти на двуизмерни екрани има няколко ограничения. Липсата на истинско пространствено възприятие, трудностите при оценката на мащаба и пропорциите и невъзможността за естествено взаимодействие с обектите са само някои от пречките, пред които са изправени дизайнерите, клиентите и крайните потребители. Използвайки възможности на VR, може да се трансформира изживяването, като се подобри разбирането, ангажираността и процесите на вземане на решения, свързани с интериорния и архитектурния дизайн.

Интегрирането на VR технологията в 3D визуализацията на обекти може значително да подобри начина, по който се възприемат и взаимодейства с мебелите и архитектурните елементи, предлагайки по-интуитивно, ангажиращо и ефективно преживяване, в сравнение с традиционните 2D визуализации. Чрез създаването на напълно поглъщаща, интерактивна среда за представяне на обекти е добре да се разчита на естествените способности на човешкия мозък за пространствено разсъждение, което потенциално може да доведе до революция в начина, по който възприемаме и планираме пространства.

Дизайнерите и архитектите се стремят да предадат идеите си по-ефективно на клиентите и заинтересованите страни. VR предоставя възможност да се освободят от ограниченията на традиционните методи за визуализация и да се изследват нови измерения на пространственото представяне. Проектантските фирми биха могли да използват тази технология за създаване на по-интересни и ефективни презентации, а клиентите – за по-добро разбиране и принос към процеса на проектиране. Освен това производителите на мебели и търговците на дребно биха могли да намерят нови начини за представяне на своите продукти, като се направи процеса на избор по-интерактивен и информативен за клиентите. Разработчиците и агентите на недвижими имоти биха могли да използват тази технология, за да се осигурят интерактивни обиколки на имотите, като потенциално се увеличи ангажираността на клиентите и се улесни процеса на избор на имот. Особено актуално е в строителните индустрии, където точната визуализация и комуникация са необходими. Като се изследват иновативни начини за взаимодействие с 3D обекти във виртуалното пространство, може да се подобри удовлетвореността на клиентите и потенциално да се намали скъпоструващите промени по време на строителната фаза.

Проучва се потенциалът на VR технологията за революционизиране на визуализацията на мебели и архитектурни елементи и да се разработи прототипно приложение, което да демонстрира тази концепция.

За да се постигнат целите, ще се разгледат основите на VR и 3D моделирането и ще се анализират и сравнят технологиите за създаване на виртуална реалност и 3D модели и разработи прототип. По-подробно:

- 1) Ще се направи цялостен преглед на технологията и техниките за VR и за 3D моделиране, свързани с визуализацията на мебели и архитектура. Ще се разгледат фундаменталните концепции, хардуерните изисквания и софтуерните инструменти, които обикновено се използват при разработването на VR за пространствен дизайн.

- 2) Ще се направи обстоен преглед на съществуващите платформи за виртуална реалност и софтуер за 3D моделиране, подходящи за визуализация на архитектурен и интериорен дизайн. Ще бъдат сравнени техните характеристики, лекота за ползване, съвместимост и пригодност за създаване на интерактивни среди с динамични текстури. Този анализ ще послужи за избора ни на инструменти за разработване на прототипи и ще даде представа за текущото състояние на VR технологиите в дизайнерската индустрия.
- 3) Ще се проектира и реализира прототип на VR приложение за визуализиране и взаимодействие с мебели и архитектурни елементи. В процеса на разработване се включва създаване на 3D модели на мебели, стени, подове и тавани, прилагане на текстури, проектиране на потребителски интерфейс и включване на интерактивни функции за подобряване на визуализацията.

Прототипът, разработен в рамките на това изследване, може да послужи като трамплин за бъдещи иновации в областта на пространствения дизайн и визуализацията, повлиявайки на начина, по който се взаимодейства с физическите пространства в различни индустрии.

Макар че техническите аспекти на VR и 3D моделирането са сложни, целта е да се представи резултат по начин, който демонстрира практическите ползи и потенциалните приложения на тази технология в процеса на проектиране. По този начин се вдъхновява практическото приложение, което може да бъде от полза за проектантите, клиентите и крайните потребители в различни дисциплини, свързани с пространственото проектиране и визуализация.

ПЪРВА ГЛАВА

Основи на виртуалната реалност и тримерното моделиране

В сферата на виртуалната реалност и 3D визуализацията доброто разбиране на основите на 3D графиката и моделирането е необходимо за работния процес. Този раздел поставя основите на разбирането на това как се създават, манипулират и визуализират триизмерни обекти и среди в цифровите пространства.

Ще се разгледат основните концепции, които формират гръбнака на 3D графиката като координатните системи, определящи виртуалното пространство. Ще се опишат градивните елементи на 3D моделите: полигони, върхове и мрежи. Текстурирането и материалите придават на обектите визуални характеристики, а с техниките осветление и засенчване се придава дълбочина и реализъм на 3D сцените.

Ще се направи преглед на различни техники за 3D моделиране, подготвяйки нашата основа за по-напреднали тематика в областта на визуализацията на виртуалната реалност. Тези основни познания са от съществено значение за разбирането на техническите аспекти на създаването на реални и интерактивни виртуални изживявания особено в контекста на визуализацията на мебели и архитектурни елементи.

1.1.1. Координатни системи и 3D пространство

В 3D графиката и виртуалната реалност разбирането на координатните системи е от основно значение. Те осигуряват рамката за позициониране и манипулиране на обекти в триизмерна виртуална среда.

Триизмерно пространство в компютърната графика е математически модел, който представя физическо пространство с помощта на три перпендикулярни оси. Това пространство позволява прецизно позициониране и ориентация на обекти, камери и светлини във виртуална среда [2].

Декартова координатна система е най-често използваната координатна система в тримерната графиката. Тя се състои от три перпендикулярни оси:

- X – Представява движението наляво-надясно;
- Y – Представява движение нагоре-надолу;
- Z – Представява движение напред-назад.

Всяка точка в триизмерното пространство се определя от позицията ѝ по тези три оси, изразена като тримерен координатен вектор във вида (x, y, z) . Например точката $(2, 3, -1)$ се намира на 2 единици по оста X, на 3 единици по оста Y и на 1 единица в отрицателна посока на оста Z.

В повечето 3D софтуери началната точка $(0, 0, 0)$ е в центъра на сцената, но това варира в зависимост от използваното приложение или игрови двигател [2].

Макар че декартовата система е най-разпространена, други координатни системи са полезни в специфични случаи като полярната, полезна за кръгови или радиални разположения, чиито точки се определят от радиус и ъгъл, сферична, която е подобна на полярната, но в 3D пространството и UV координатна система, използвана при изобразяването на текстури, тоест на двумерни изображенията увити около тримерни обекти [2].

Разбирането на тези координатни системи служи да се разбере точното поставяне и манипулиране на обектите в средите на виртуалната реалност особено когато става въпрос за поставяне на мебели и архитектурна визуализация.

1.1.2. Полигони, върхове и мрежи

Полигоните, върховете и мрежите са основните градивни елементи на 3D моделите в компютърната графика и виртуалната реалност. Разбирането на тези елементи е служи за ефективното създаване и манипулиране на 3D обекти.

Върх е точка в 3D пространството, определена от нейните координати X, Y и Z. Върховете са най-основните елементи на 3D геометрията и служат като ъгли или крайни точки на ръбове в 3D моделите.

Ръб е отсечка от линия, свързваща два върха. Ръбовете определят границите на лицата и допринасят за цялостната форма на 3D обекта.

Лице е плоска повърхност, ограничена от ръбове. В 3D моделирането лицата обикновено са с триъгълна или четириъгълна форма, въпреки че съществуват и по-сложни многоъгълници.

Полигон е затворена форма, образувана чрез свързване на множество върхове с ръбове [2]. Най-често срещаните видове многоъгълници в 3D графиката са:

- **Триъгълник** – Най-простият и най-широко използван тип многоъгълник, определен от три върха;
- **Четириъгълник** – Предпочитани от някои графични дизайнери заради лесната им манипулация;
- **N-ъгълник** – Полигони с повече от четири страни, които обикновено се избягват в работната среда, защото са трудни за оптимизация.

Мрежа (Mesh, Meshes) е съвкупност от върхове, ръбове и лица, които определят формата на 3D обект. Мрежите са прости като на куб или много сложни като подробен модел на герой от игра. Качеството и сложността на мрежата оказва значително влияние върху производителността на рендиране и визуалната достоверност във VR приложенията [2].

Топологията на една мрежа се отнася до начина, по който полигоните са свързани и се движат по повърхността на 3D модела. Добрата топология е от значение за:

- плавни деформации в анимирани модели;
- Ефективно текстуриране при рисуване и по-важно разположението като текстурно съотнасяне – texture mapping;

- Оптимизирана производителност при рендиране.

В контекста на визуализацията на мебели и архитектура във VR разбирането на тези концепции помага за създаване на ефективни и визуално привлекателни 3D модели. Оптимизирането на гъстотата на мрежата (броят на полигоните) е от полза за поддържане на гладка работа във VR среди в реално време, като същевременно да се запази определено визуално качество.

1.1.3. Текстуриране и материали

Текстурирането и материалите са ключови елементи в 3D графиката. То придава на обектите техните визуални характеристики и ги прави да изглеждат реалистични или стилизирани според избрания стил от разработчиците. Този процес е особено важен във VR за визуализация на мебели и архитектура, тъй като оказва значително влияние върху възприемането на виртуалните обекти от потребителя.

UV мапинг е съпоставянето на текстури към геометрия. Това е процес на проектиране на 2D изображение върху повърхността на 3D модел. Терминът "UV" се отнася до координатите на двумерното пространство на текстурата, като U координатата представлява хоризонталната ос, а V координатата представлява вертикалната ос [2].

UV мапингът позволява на дизайнерите да контролират прецизно начина на прилагане на текстурите върху 3D обекти, като се гарантира, че моделите и детайлите се появяват правилно върху повърхността на модела. Например при направата на модел на глобус на земята. Процесът е валиден и в обратна посока. Все едно се създава земна карта.

Различните видове текстури работят заедно, за да се създаде крайния вид на 3D обекта [2]:

- 1. Дифузна (цветова) карта** – Основният цвят или цветови шаблон на повърхността на обекта;

- 2. Карта на нормалните вектори (Normal map)** – Симулират се детайли на повърхността, без да се добавя допълнителна геометрия, като се създава илюзия за дълбочина и сложност;
- 3. Карта на отраженията (Specular map)** – Определя се грапавостта или гладкостта на различни части на повърхността;
- 4. Карта на грапавостта** – Контролират се детайлите на повърхността на лицето, като се влияе на начина, по който светлината се разпръсква по повърхността;
- 5. Карта на околните засенчвания (Ambient Occlusion)** – Добавят се фини сенки в области, до които светлината достига по-рядко, като се подобрява възприятието за дълбочина;
- 6. Карта на разместванията** – Променя се геометрията на модела, за да се създадат реални детайли на повърхността.

Материалите в 3D графиката определят начина, по който даден обект взаимодейства със светлината чрез техните свойства. Ключовите такива включват:

- **Отразяваща способност** – Колко светлина се отразява от повърхността;
- **Прозрачност** – Степента, в която светлината преминава през обект;
- **Пречупване** – Как светлината се огъва, когато преминава през обект;
- **Подповърхностно разсейване** – Проникваща и разсейваща способност на светлината в полупрозрачни материали.

Шейдърите са програми, които определят начина, по който 3D енджинът (софтуер, който позволява генериране на триизмерни графики в реално време) визуализира материалите. Те комбинират информацията за текстурите с изчисленията за осветлението, за да се създаде крайният вид на обекта. Усъвършенстваните шейдъри могат да симулират сложни материали

като дърво, метал или плат, което служи за реалистичната визуализация на мебели във VR [2].

Във VR приложенията за визуализация на мебели и архитектура ефективното използване на текстури и материали е от значение за създаването на убедителна и поглъщаща среда. Това позволява точно да се интерпретира вида и усещането на различните повърхности, като се подобрява цялостното изживяване и се дава възможност за по-добро вземане на решения в процесите на проектиране.

1.1.4. Осветление и засенчване

Осветлението и засенчването са ключови елементи в 3D графиките и VR, които допринасят значително за реализма и атмосферата на дадена сцена. Те играят жизненоважна роля за това как потребителите възприемат дълбочината, формата и материалните свойства на виртуалните обекти включително мебели и архитектурни елементи. Има няколко вида светлинни източници [2]:

- **Точкови** – Те излъчват светлина във всички посоки от една точка в пространството. Полезни са за симулиране на електрически крушки без външно тяло или малки източници на светлина.
- **Насочени** – Те излъчват паралелни светлинни лъчи в определена посока дори и от безкрайно отдалечен източник. Използват се за симулиране на слънчева светлина.
- **Конусовидни** – Те излъчват светлина под формата на конус от една точка. Полезни са за фокусирани светлини, идващи от лампи или фенерчета.
- **Плоски** – Те излъчват светлина от определен двуизмерен обект, като създават меки и реалистични сенки. Те са скъпи от гледна точка на изчисленията, но се осигурява много естествено изглеждащо осветление.

Моделите на засенчване са математически подходи, използвани за изчисляване на начина, по който светлината взаимодейства с повърхностите в 3D графиките. Те са от решаващо значение за създаването на реалистични или стилизирани изображения на обекти във виртуална среда. Изборът на модел на засенчване оказва значително влияние както върху визуалното качество, така и върху изчислителните изисквания на дадена 3D сцена [2]. Основните модели на засенчване са:

- **Плоско засенчване** – Най-простата и бързо изчислима форма на засенчване, при която всеки полигон на обекта получава един цвят въз основа на ориентацията му към източника на светлина и нормалният му вектор. Това създава ясно изразена повърхност, позната от старите видео игри;
- **Засенчване на Gouraud (Гуро)** – Този метод интерполира цветовете по повърхността на полигоните, като създава по-плавно преливане от плоското засенчване. Изчислително ефективен е, но се получават артефакти върху извити повърхности;
- **Засенчване на Phong (Фонг)** – С този модел се изчислява цвета за всеки пиксел, а не за всеки връх, което води до по-точни светлосенки и по-плавен външен вид от засенчването на Gouraud.

Глобалното осветяване са усъвършенствани техники за осветяване, които отчитат както директното осветяване от източниците на светлина, така и индиректното осветяване от светлината, която се отразява от други повърхности в сцената [2]. Такива методи са:

- **Трасиране на лъчи** – Симулира се пътят на светлинните лъчи, докато те се отразяват в сцената, създавайки изключително реалистично осветление, отражения и сенки. Макар че е изчислително интензивно, трасирането на лъчи в реално време става все по-възможно за приложенията за виртуална реалност.

- **Радиометричност по метода на броимите елементи (Radiosity)** – Изчислява се преноса на дифузна светлина между повърхностите в сцената, като се създава меко и естествено осветление. Особено полезно е в архитектурната визуализация.
- **Оклузия на околната среда (Ambient Occlusion)** – Приближава се до начина, по който светлината се излъчва в реална среда, като се затъмняват областите, в които има вероятност светлината да бъде закрита, като например ъглите, пукнатините и областите под обектите.

Във VR приложенията за мебели и архитектурна визуализация ефективното осветление и засенчване са от решаващо значение за създаване на усещане за присъствие и точното възприемане на формата, текстурата и пространствените връзки на обектите. Техниките за осветяване в реално време трябва да балансират между визуалното качество и производителността, за да се поддържа плавно скорост на кадрите, необходима за комфортно VR изживяване.

1.2. Техники за моделиране

3D моделирането е процес на създаване на триизмерни обекти или повърхности в цифрова среда, приближаващи се до реалните обекти. При 3D моделирането се използват различни техники, всяка от които има своите позитиви и приложения. Описани са основните техники за моделиране [2].

1.2.1. Прimitives моделиране

- Използват се основни 3D форми – кубове, сфери, цилиндри и други като градивни елементи;
- Формите могат да се комбинират, изваждат или пресичат, за да се създадат по-сложни форми;

- Идеален метод за моделиране на твърди повърхности и архитектурни елементи;
- Бърз и ефективен е за създаване на прости обекти или базови мрежи.

1.2.2. Моделиране на полигони

- Включва се манипулиране на върховете, ръбовете и лицата на мрежата;
- Предлага високо ниво на контрол и детайлност;
- Подходящ е за органични форми и сложни структури;
- Може да се раздели на части за по-гладки повърхности;
- Разпространено е при моделиране на герои и създаване на подробни и комплексни обекти.

1.2.3. NURBS (Non-Uniform Rational B-Splines) моделиране

- Използват се математически дефинирани криви и повърхности;
- Създават се гладки и точни форми без видими полигони;
- Идеален е за дизайни в производството и моделиране на автомобили;
- Позволява лесно мащабиране без загуба на детайл;
- Трудно е за текстуриране и анимиране в сравнение с полигоновите модели.

1.2.4. Моделиране на повърхнини с подразделяне

- Съчетават се аспекти от моделирането с полигони и NURBS моделирането;
- Започва се с модел с малко детайл (малко на брой полигони), който след това се разделя на части за постигане на гладкост;
- Позволява както широко дефиниране на формата, така и работа с фини детайли;
- Използва се широко във филмовата индустрия и в производството на тежки игри.

1.2.5. Скулптурство. Дигитално извайване

- Наподобява традиционните техники за скулптурно майсторство в цифрова среда;
- Използват се виртуални инструменти, базирани на реални инструменти – четки за вдълбаване, изпъкване или друг вид манипулиране на повърхности;
- Отлично е за създаване на органични форми с висок детайл;
- Често се използва в комбинация с ретопология (процес на оптимизация на мрежата от полигони) за оптимизиране на моделите.

1.2.6. Процедурно моделиране

- Генерира се модел въз основа на правила или алгоритми;
- Полезно е за създаване на мащабни среди или повтарящи се структури;
- Позволява бързи итерации и вариации;
- Обикновено се използва в архитектурната и визуализация за земята.

Примитивното моделиране и моделирането с полигони често се използват за създаване на основни форми на мебели и архитектурни елементи, NURBS за проектиране на прецизни, извити мебелни елементи. Моделирането на повърхнини с подразделяне помага за създаването на гладки, органични форми, като същевременно се запазва контролът върху цялостната форма. Дигиталното извайване се използва за добавяне на фин детайл към мебели или архитектурни орнаменти. Процедурното моделиране е особено полезно за генериране на мащабни архитектурни среди или повтарящи се елементи като прозорци или плочки поради многобройността на тези елементи.

Изборът на техника за моделиране често зависи от специфичните изисквания на проекта, желаното ниво на детайлност и предвидената употреба на модела във виртуалната среда. По-сложни обекти във VR приложенията се

създават с помощта на комбинация от тези техники, за да се постигне най-добрият баланс между визуално качество и производителност.

1.3. Основи на виртуалната реалност

1.3.1. Определение за виртуална реалност и кратка история

Виртуалната реалност (Virtual Reality, VR) е компютърно генерирана симулация на триизмерна среда, с която човек взаимодейства по привидно реален или физически начин, като се използват специални периферни устройства, като например хедсет с вътрешен екран, аудио колони и ръкавици или джойстици, снабдени със сензори [3].

Концепцията за VR води началото си от 60-те години на миналия век. В първите разработки се включва завладяващи вниманието симулатори с множество сензори като Sensorama на Мортън Хейлиг. Малко по-късно се появяват първите дисплеи, които се монтират на главата на човек. Индустрията търпи бурно развитие през 80-те и 90-те години, защото тогава значително се покачва интересът на бизнесмени и учени за намиране на решения, използващи технологии за виртуална реалност. Появяват се първите аркадни машини с периферия за виртуална реалност. През 1995 година за първи път излиза „конзола“ за виртуална реалност Virtual Boy на японската компания за игри Nintendo, достъпна за битовия потребител. За съжаление тази игрова конзола е пълен провал, продават се малко бройки. Често потребителите се оплакват от лоша ергономичност, сухо и болезнено зрение както и неудовлетвореност от практически едноцветната червена палитра от цветове. Около 30 години по-късно се появяват първите традиционни хедсети за виртуална реалност, като дизайна от това време остава непроменен. Едни от популярните за това модели са Oculus Rift, HTC Vive, PlayStation VR и други по-съвременни разработки като Meta Quest и Valve Index. Други приложения на системите за виртуална реалност са във военната и авиационната индустрия за обучение на войници и пилоти.

1.3.2. Видове VR системи

Системите за виртуална реалност се разделят основно според това до каква степен отвличат вниманието на потребителя от реалния свят и да го потопят във виртуалния свят. Това са напълно потапящи, полупотапящи и непотапящи. Други характеристики, по които се характеризират са броя от необходими сензори, цена, специфична употреба и достъпност [3].

1.3.2.1. Напълно потапяща VR системи

- Използва дисплеи, монтирани на главата (Head Mounted Display, HMD);
- Осигурява пълно потапяне във виртуална среда;
- Идеално е за детайлна визуализация на мебели и архитектура.

1.3.2.2. Полупотапящи VR системи

- Към тези спадат системите за автоматична виртуална среда CAVE (Cave Automatic Virtual Environment);
- Използват се големи прожекционни екрани в определени за тази нужда помещения;
- Полезни за съвместни прегледи на архитектурни проекти.

1.3.2.3. Непотапящи VR системи

- 3D среди, базирани на настолни компютри;
- Обикновено са по-достъпни, тъй като не се изисква допълнителна техника;
- Подходящи са за първоначални концепции за проектиране и презентации на клиенти.

1.3.2.4. Разширена реалност (Augmented Reality, AR) и смесена реалност (Mixed Reality, MR)

- проектират се виртуални обекти върху реалния свят;

- Имат нужда от камера;
- Подходящо за визуализация на разположението на мебелите на място.

1.3.3. Основни компоненти на VR системите

Основните компоненти, които системите за виртуална реалност използват, се разделят основно на хардуерни и софтуерни.

1.3.3.1. Хардуерни компоненти

- **Дисплеи, монтирани на главата (HMD)** – Освен дисплея или дисплеите, които показват картината, се включват и оптични лещи;
- **Сензори (жироскопи и др.)** – Използват се за проследяване на позицията на главата, ръцете и тялото в реалния свят, за да може точно да се пренесе перспективата във виртуалния свят;
- **Системи за проследяване на ускорението при движение (акселерометри)** – Външни или вътрешни сензори за проследяване движенията на главата и ръцете на потребителя;
- **Входни устройства** – Включват се ръчни контролери като Oculus Touch, Valve Knuckles и други или хаптични ръкавици за по-естествено взаимодействие. Обикновено се включват бутони, тригери или джойстици;
- **Компютър** – Системите за виртуална реалност сами по себе си не са компютри. Те служат само за показване на картина и взаимодействие със специализирана периферия и затова се нуждаят от компютър. Необходим е достатъчно мощен процесор и видео карта;
- **Аудио устройства** с вградени аудио процесори.

1.3.3.2. Софтуерни компоненти

- **3D двигатели** – Осигуряват цялостен набор от инструменти за създаване на VR преживявания (включват се симулатори и 3D видео изживявания);

- **Framework за разработка на VR (OpenVR/SteamVR, Oculus SDK, Google VR SDK и др.)** – Част от основните инструменти за разработка. Предоставят се готови модули и системи. Също така се осигурява комуникация между хардуера и софтуера;
- **Софтуер за 3D моделиране и анимация** – Използва се за създаване на модели.

Тези компоненти работят заедно, за да създадат цялостното VR изживяване. HMD осигурява на потребителя визуализирането на виртуалната среда, системите за проследяване и входните устройства позволяват взаимодействието на потребителя с виртуалния свят както и неговото точно позициониране, по-мощен хардуер позволява визуализирането на сложни 3D модели в реално време без да предизвиква нежелани странични ефекти у потребителя, софтуерът свързва всичко заедно, като се занимава и с физиката, осветлението и въвеждането от потребителя [3].

В мебелната и архитектурната визуализация тези компоненти позволяват реалистичното изобразяване на материали и осветление и интерактивно манипулиране на мебелни елементи, ако има такава функционалност.

1.4. Човешко-машинни интерфейси, използвани в системите за виртуална реалност

1.4.1. Дисплеи, монтирани на главата (HMD)

Дисплеите, монтирани на главата, са основният визуален интерфейс в съвременните VR системи. Състоят се от:

- **Дисплеи** – Обикновено по един за всяко око, който осигурява стереоскопично 3D виждане;
- **Лещи** – За фокусиране и оформяне на зрителната зона;

- **Сензори за проследяване на главата** – За следене на движението на главата и съответно регулиране на изгледа.

Основните характеристики на съвременните хедсети, които са от съществено значение са разделителната способност за по-ясни изображения, полезрението (Field Of View, FOV), честота на опресняване на дисплея, осигуряваща гладко предаване на визуалната информация, и ергономичността, която е от значение за продължителната употреба на продукта.

1.4.2. Системи за следене на движението

Проследяването на движението на потребителя спомага за правилното пренасяне на позицията и перспективата на потребителя във виртуалната реалност, подобрявайки усещането за присъствие вътре [3]. Делят се на два основни вида:

1.4.2.1. Проследяване отвън-навътре

- Използват се външни сензори или камери за проследяване на движението на потребителя;
- Имат висока точност и по-голяма област на проследяване;
- Изискват настройка и калибриране преди използване.

1.4.2.2. Проследяване отвътре-навън

- Използват се камери около главата на потребителя, намиращи се на хедсета, за проследяване на движението спрямо околната среда;
- По-лесни са за настройване и са по-преносими;
- Губи се малко точност спрямо външните системи в определени случаи;
- Хедсета става малко по-тежък, което довежда до усещане за неудобство.

1.4.3. Хаптични устройства и обратна връзка

Хаптичната технология добавя усещането за допир към различен вид изживявания, като се подобрява потапянето в средата и се осигурява обратна връзка при взаимодействие със средата.

Към видовете хаптика и хаптични устройства се включва вибратори в контролерите, хаптични ръкавици и костюми за цялостно проследяване на тялото. Последните са рядко използвани, като реално приложение се намира в киноиндустрията и анимацията [3].

1.4.4. Входни устройства

Устройствата за въвеждане на команди позволяват на потребителите да взаимодействат с виртуалната среда и да я манипулират [3].

1.4.4.1. Ръчни контролери

- Това е най-разпространеният метод за въвеждане на данни за потребителската VR;
- Обикновено се предлагат по двойки за взаимодействие с двете ръце;
- Разполагат с бутони, спусъци и често с джойстици или тъчпади.

1.4.4.2. Ръкавици

- Позволяват по-естествени движения на ръцете и пръстите;
- Дават обратна връзка, което подобрява взаимодействието;
- По-рядко се използват и са по-скъпи от контролерите.

1.4.4.3. Проследяване на очите

- Следи движението на очите за взаимодействие със системата;
- Може да се използва от хора, които имат проблеми с подвижността.

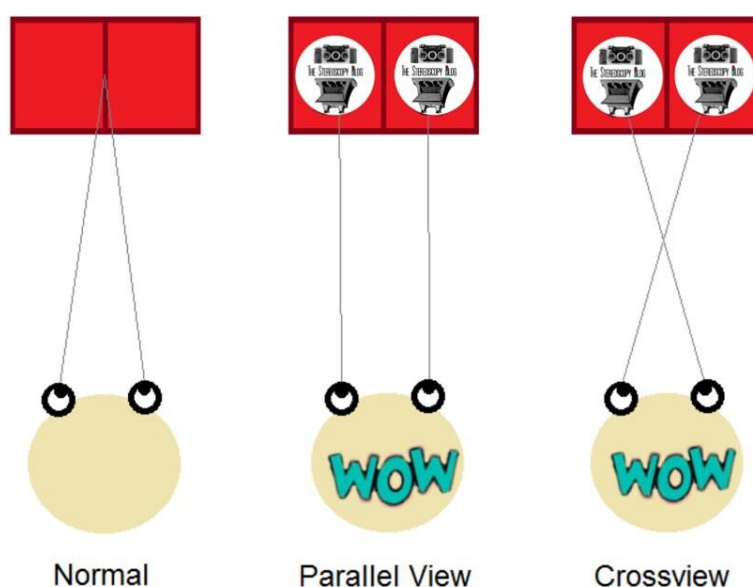
1.5. Ергономичност на човешкото тяло, сетива и проблеми

1.5.1. Визуално възприятие във VR

Визуалните възприятия на човека във VR служат за създаване на правдоподобни и удобни изживявания особено при визуализация на мебели и архитектура. По-надолу са описани някои от ключовите характеристики [3].

1.5.1.1. Стереоскопия

- Осигурява се възприемане на дълбочината чрез представяне на малко по-различни изображения за всяко око. Така Обикновени 2D растерни изображения с обекти, които трябва да са триизмерни, им се придава този дълбок вид;
- Обяснява пространството при използване на нетрадиционни дисплеи. На фигура 1.1 е показано разликата между нормалното възприятие на изображения на 2D екрани и стереоскопичните дисплеи.



Фигура 1.1: Методи за представяне на стереоскопични изображения. Отляво е показано типичното възприятие на човек, гледащ в плосък дисплей.

1.5.1.2. Полезрение (FOV)

- По-широкият FOV увеличава колко повече информация от околната среда, посредством зададения обхват на кадъра на камерата, се показва на потребителя;
- Човешките очи имат около 210 градуса хоризонтален ъгъл на видимост. Повечето VR хедсети показват между 90 и 110 градуса;

1.5.1.3. Разделителна способност и плътност на пикселите

- По-високата разделителна способност намалява влиянието на аномалията, предизвикваща видимостта на „решетка“ около пикселите;
- Подобрява се възприемането на детайлите посредством по-голямата детайлност.

1.5.1.4. Честота на опресняване и плавност на движенията на обектите

- По-високата честота на опресняване от 90 Hz намалява усещането за загуба на равновесие и дезориентацията;
- Подобрява се плавността на обектите, като се предоставя повече на брой моменти на придвижване на обектите, ако са в движение за единица време, най-често 1 секунда.

Една забележка, която е добре да се направи, е, че както с всеки дисплей очите се фокусират върху него, докато се фокусират с виртуални обекти на различни разстояния, което предизвиква напрежение в очите и дискомфорт при продължителна употреба.

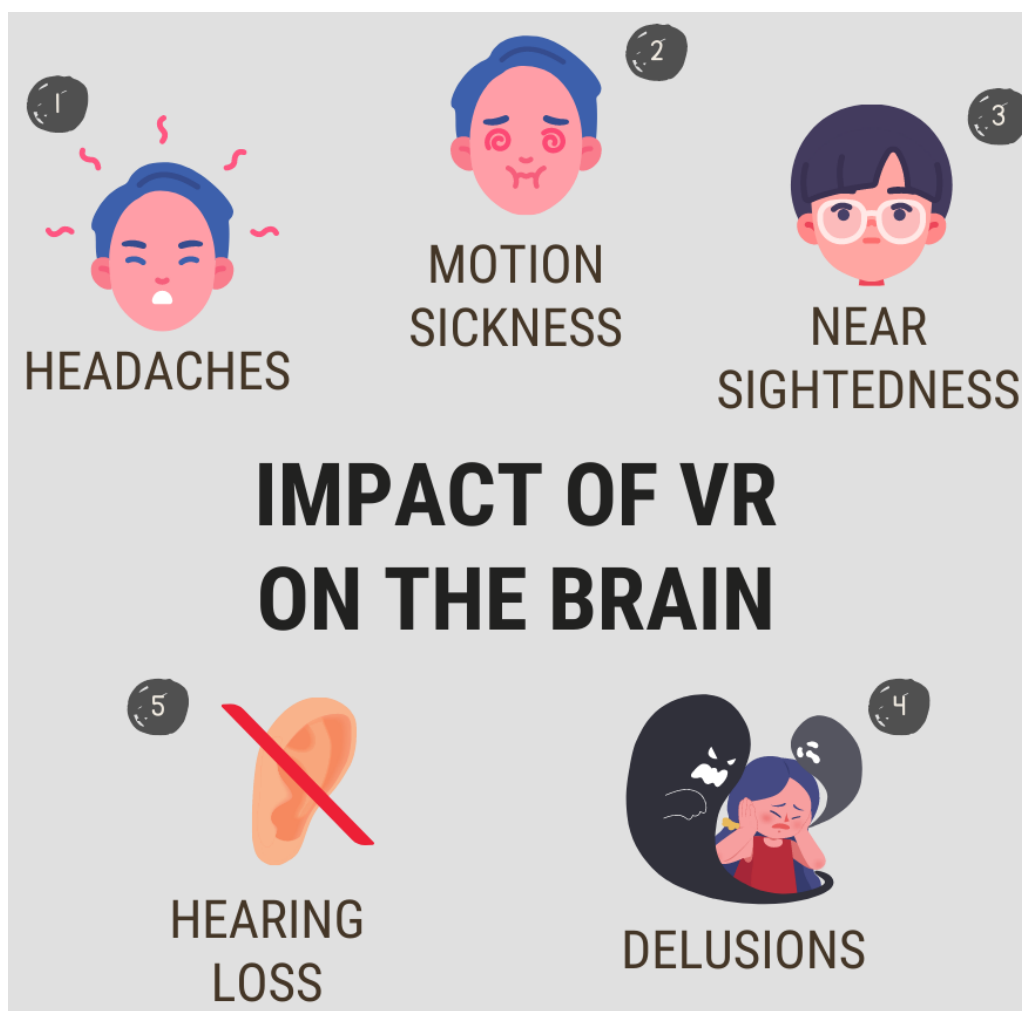
1.5.2. Пространствен звук

Пространственото аудио значително подобрява усещането за присъствие във VR средите. Това е приложимо за обекти, които са източници на звукова информация, и потребителят се опитва да разграничи на какво разстояние е

този обект. В това се включва звук за сигнализация и акустични характеристики на материалите, което придава по-голяма достоверност [3].

1.5.3. Проблеми със загубата на равновесие, дезориентацията и комфорта

Понякога VR предизвиква дискомфорт или дезориентира човек при продължителна употреба. Разбирането на и справянето с тези проблеми е необходимо за създаването на удобни за потребителите VR изживявания [3]. На фигура 1.2 е показано какви странични ефекти могат да бъдат причинени.



Фигура 1.2: Странични ефекти при продължителна употреба на система за виртуална реалност. Най-често се получава главоболие, загуба на ориентация и късогледство. Във редки случаи може да се получи загуба на слух или халюцинации.

1.5.3.1. Причини за нежелани странични ефекти от работата с VR

- Сензорно несъответствие между зрителната и вестибуларната (равновесната) система;
- Закъснение между движението на обекта и визуалната опреснение на екрана;
- Изкуствени методи за придвижване като телепортиране и летене.

1.5.3.2. Решения за справяне с такива проблеми

- Поддържане на висока и стабилна честота на опреснение на екрана;
- Прилагане на удобни методи за придвижване и със съобразена скорост на придвижване;
- Постепенно въвеждане във VR преживяванията.

1.5.3.3. Ергономични съображения при работа със системи за виртуална реалност

- Разпределението на теглото на хедсетовите ще допринася за равномерно натоварване на мускулите на врата и главата;
- Възможността за адаптиране към различни размери на главата на човека спомага с комфорта;
- Вентилация за предотвратяване на замъгляването на лещите и натрупването на топлина в очно-дислейното пространство.

1.5.3.4. Продължителност на работа със система за виртуална реалност

- Препоръчително е да се правят редовни почивки за предотвратяване на напрежението и умората на очите при продължителна работа;
- При работа със системи за виртуална реалност се изисква техническо време не само да се свикне с новите устройства, но и с цялото изживяване.

1.5.3.5. Съображения за достъпност от хора с увреждания

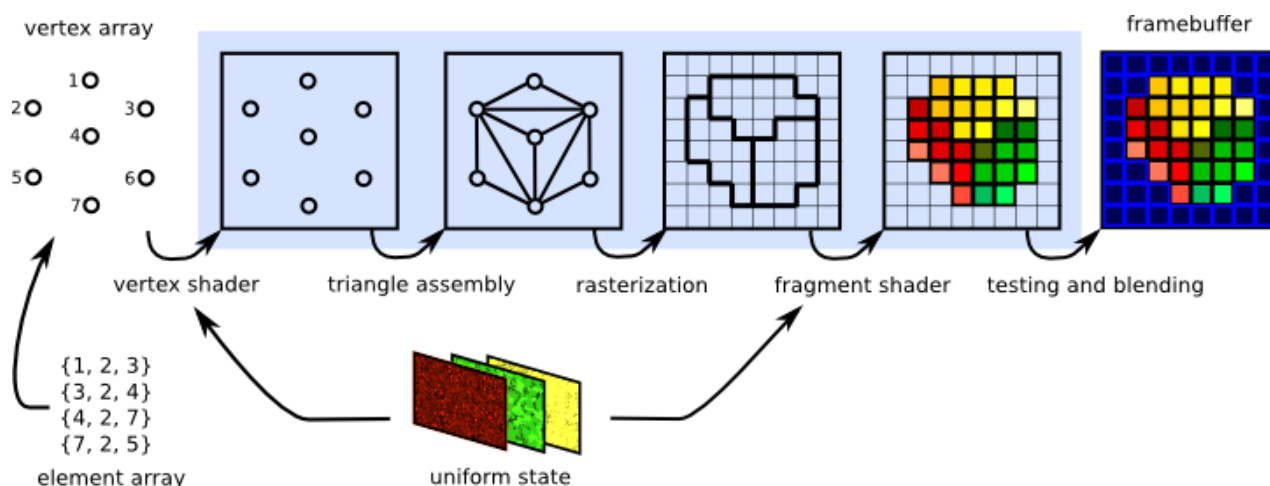
- Някои потребители имат затруднения с подвижността си и е необходимо да се ползва хедсета в легнало или седнало положение;
- При неналичието на възможност за виртуална „разходка“, каквито много рядко се ползват, е необходимо да има добри функции за движение;
- В някои случаи е необходимо да има допълнителни настройки за потребители със зрителна (но не пълна слепота), звукова или говорна недостатъчност.

1.6. Рендиране на графика в реално време за виртуална реалност

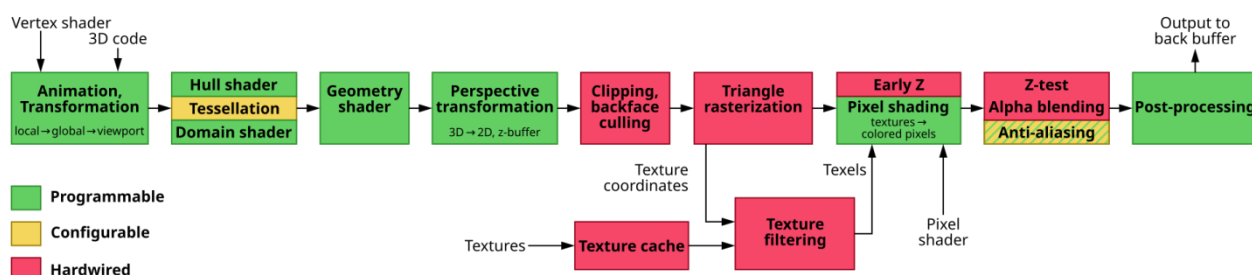
1.6.1. Стъпки за обработка на графика и графични сцени

Така наречените графични конвейери (Graphics pipelines) са поредица от стъпки, с които се трансформират данните от 3D сцената в 2D изображение, за да се покаже на дисплей. В системите за виртуална реалност този процес трябва да е изключително ефективен, за да се поддържа висок брой на кадрите и ниско времезабавяне при извършване на действие и видимостта на резултата [2], [3].

Тук ще се обясни основните етапи в дефинирани стъпки без допълнителен детайл в типичен графичен поток за VR приложения, без да се навлиза в подробности, понеже не са фокуса на проучването. Важно е редът на тяхното действие. На фигура 1.3 е показано визуално и кратко как изглежда процеса на обработка на видео данни. На фигура 1.4 е представен традиционният поток на засенчванията (Shading). Това е вид компютърна програма, алгоритъм, с който се определя цвета и осветеността на пикселите.



Фигура 1.3: Визуално представяне на графичния поток. Включват се основни етапи на обработка като приложение (Application), геометрия (Geometry), Растеризация (Rasterization), пост-процериране, докато не се получи готово изображение във видео паметта, готово да се покаже на екрана.



Фигура 1.4: Класически поток със стъпки за обработка на графични данни — засенчване. Забелязва се, че стъпки на засенчване може да има на всеки етап и повечето стъпки са програмируеми с цел оптимизиране на производителността.

1.6.1.1. Етап на начално процесиране на сцената (Application stage)

- 1) Създаване и управление на сцената;
- 2) Физически симулации;
- 3) Актуализации на анимацията.

1.6.1.2. Обработка на геометрията (Geometry)

- 1) Засенчване на върховете;
- 2) Теселация – Процес на намиране на повърхности с повтаряеми текстури;
- 3) Засенчване на геометрията;
- 4) Прожектиране на 2D екран.

1.6.1.3. Растеризация

- 1) Превръщане на векторни данни в пиксели.

1.6.1.4. Обработка на пиксели

- 1) Прилагане на текстури (Texture Mapping);
- 2) Засенчване на фрагменти;
- 3) Изчисляване на осветлението за всеки пиксел.

1.6.1.5. Пост-процесиране

- 1) Създаване на чувство за дълбочина;
- 2) „Съшиване“ на цветовете съседни пиксели (Blending);
- 3) Заглаждане на ръбове (Anti-Aliasing).

1.6.1.6. Специфични за VR съображения

- 1) Стерео рендиране – Генериране на отделни изображения за всяко око;
- 2) Корекция на изкривяването на лещите.

1.6.2. Техники за оптимизация

Чрез Оптимизиране на графичните ефекти и алгоритми се печели увеличение и поддържане на висока и стабилна производителност, в резултат на което се увеличава и комфорта. Понякога производителността се увеличава за сметка на качеството на картината [2]. Едни от основните техники за оптимизация са:

- **Рендериране на фокус и периферия (Foveated Rendering)** – Изобразяват се областите, в които потребителят гледа, с висока детайлност, като същевременно се намалява детайлността в периферното зрение. Изисква се да има устройство за проследяване на очите;
- **Стерео рендиране с еднократно минаване** – Рендерират се изгледите на двете очи едновременно и се намалява натоварването на процесора, тоест рендира се едно цяло изображение и се разделя на две, като едната част се показва на едното око, а другата част на другото;
- **Засенчване с много разделителни способности** – Прилага се различна гъстота на пикселите в изображението въз основа на изкривяването на изображението от оптичната леща;
- **Ниво на детайлност (Level Of Detail, LOD)** – Динамично се регулира детайлността на модела, въз основа на разстоянието на потребителя или приоритета на обекта;
- **Изчистване на околни обекти (Culling)** – Избягва се рендирането на обекти, които не се виждат от камерата;
- **Инстанциране при многобройни еднакви копия на обект** – Ефективно визуализиране на множество копия на един и същ обект;
- **Комбиниране на текстури (Atlasing)** – Комбинира се множество от текстури в една по-голяма текстура, за да се намали броят на извикванията за рисуване, което не само подобрява производителността, но и може да спести видео памет.

1.6.3. Съображения, свързани с честотата на опреснение и времезабавянето

1.6.3.1. Изисквания за броя на кадрите

- Необходимо е да се поддържа минимум 90 кадъра в секунда за повечето VR приложения, за да се избегне нежелани странични ефекти. Някои системи от висок клас се стремят да постигнат 120 FPS или повече;
- Постоянната честота на кадрите – произвеждането на един и същи брой кадри за единица време без флуктуации в броя или времето от кадър до кадър често е по-важна от по-високата.

1.6.3.2. Съображения, свързани с времезабавянето

- Закъснението между показването на кадрите трябва да бъде по-малко от 20 милисекунди за комфортно VR изживяване. Източниците на закъснение включват проследяване, рендиране и технология на дисплея;
- Техники като предсказващо проследяване и асинхронно репроектиране намаляват възприетото времезабавяне.

1.6.3.3. Балансиране на качеството и производителността

- Използване на динамично мащабиране на резолюцията като NVidia DLSS, AMD FSR, Intel XeSS и други е препоръчително при системи с по-ниска производителност;
- Настройки на качеството трябва да бъдат добре преценени спрямо възможностите на компютърната система;
- Внимателно оптимизиране на сцените – броя и геометрията на обектите.

Тези техники за рендиране и графики в реално време са необходими за създаването на адаптивни и висококачествени VR изживявания в областта на мебелната и архитектурната визуализация.

1.7. Създаване на 3D обекти и взаимодействие във VR

1.7.1. Оптимизиране на 3D моделите за VR

Създаването и оптимизирането на 3D моделите за VR се постига по няколко начина [2]:

1.7.1.1. Оптимизиране на броя на полигоните

- Намалява се броят на полигоните, без да се прави компромис с визуалното качество. Използват се инструменти за опростяване на сложни модели, като се опростява тяхната мрежа и брой полигони.

1.7.1.2. Оптимизиране на текстурите

- Използват се комбинирани текстури за намаляване на системните извикванията;
- Текстурите се детайлизират до определена разделителна способност за постигане на баланс между качеството и производителност;
- Може да се използват таблици с нормални вектори за добавяне на детайл, без да се увеличава броят на полигоните.

1.7.1.3. Прилагане на вариращо ниво на детайлност (Level Of Detail, LOD)

- Създават се множество версии на модели с различни нива на детайлност, като се превключва автоматично между различните модели въз основа на разстоянието от потребителя.

1.7.1.4. Изрязване на невидими обекти (Culling)

- При правилна настройка, използвайки полезрението, се осигурява ефективно рендиране само на видимите елементи, спестявайки системни ресурси.

1.7.1.5. Предварително обработване („Изпичане“, Baking) на осветление в светлинни карти

- Предварително се изчислява информацията за осветлението, за да се намали натоварването при рендиране в реално време, защото работата вече е свършена;
- Има огромна разлика в производителността между „изпечено“ осветление и осветление в реално време.

1.7.2. Ниво на детайлност (LOD) в среди за виртуална реалност

Ниво на детайлност като способ може да се приложи на различни етапи на в графичния поток на един обект. Този ефект зависи от разстоянието между обекта и потребителя. От важно значение е също плавният преход между нивата на детайлност, защото влияе на изживяването. Такива подходи са:

- **Геометричен LOD** – Постепенно се намалява сложността на геометрията на модела;
- **Текстурен LOD** – Прилага се техниката на Mip-текстурирането за създаването на прогресивно намаляващи в детайл текстури в комбинирана текстура. Подходящо е за големи равнинни пространства;
- **Shader LOD** – Опростява се сложността на шейдъра;
- **LOD при растителни обекти и тълпи от хора** – Възможно е намаляването на детайла на масови обекти или обекти с висока детайлност.

1.7.3. Естествени потребителски интерфейси и разпознаване на жестове

Потребителският интерфейс е най-първото нещо, с което потребителят си взаимодейства, и е от голямо значение за ориентацията в приложението. Във виртуалното пространство това, на което човек разчита, е на своите ръце, очи и говор [3].

- **Проследяване на ръцете и разпознаване на жестове** – При изпълнение на естествени движения на ръцете се изпълняват команди, базирани на жестове, за обичайни действия в графиката като мащабиране, завъртане и други, за да се манипулират обекти.
- **Взаимодействие, базирано на погледа** – При използване на устройство за проследяване на очите, което следи за мигане и фокусиране, се предприемат действия. Тези функции имат по-ограничено приложение.

1.8. Присъствие и привличане на вниманието на потребителите във VR среда

1.8.1. Фактори, влияещи върху присъствието

Присъствието на потребителя във виртуалната реалност се отнася до субективното преживяване, че човек се намира на едно място или в една среда дори когато физически се намира на друго място. Няколко характеристики допринасят за създаването на силно усещане за присъствие във виртуалния свят [3].

1.8.1.1. Визуална достоверност

- Двата дисплея, показващи изображения на потребителя, трябва да са с висока разделителна способност, но и да бъдат производителни;
- Реалистичното осветление и сенки допринасят за усъвършенстване на околната среда, но и имат голяма нужда от системни ресурси;
- Точни текстури и материали придават точна картина на продукта, който се разпознава от потребителя, за да се направи сравнение с този от реалния свят.

1.8.1.2. Качество на звука

- 3D пространствен звук спомага за пространствено ориентиране, но е със специфично приложение;
- Реалистични звуци от околната среда;
- Със звукови ефекти за обратна връзка относно действията на потребителя се потвърждава, че определен резултат вследствие от действие е постигнат.

1.8.1.3. Интерактивност

- Естествено и интуитивно управление;
- Реакция в реално време на потребителските входове;
- Взаимодействия, базирани на физиката.

1.8.1.4. Полезрение (Field Of View, FOV)

- По-широко полезрение допринася за визуализирането на по-голяма част от околната среда;
- Съществуват ограничения в софтуера относно колко голямо FOV може да се зададе, преди образът да се огъне (Ефектът на рибеното око). Това зависи от способностите на дисплея.

1.8.1.5. Честота на кадрите и закъснение

- Висока и стабилна честота на кадрите над 90 FPS (Frame/s Per Second) помага да се избегнат проблемите с прилошаването и дезориентацията.

1.8.1.6. Последователност на виртуалния свят

- Последователен и правдоподобен дизайн на средата спомага за придобиване на логическа последователност – такава, каквата е и в реалния свят;

- Реалистичната физика, определяща поведението на обектите, симулира определени реални свойства на обекти като трудноподвижност заради теглото и съпротивления при плъзгане.

1.8.1.7. Аватар и синхронизация с действията на потребителя

- Точното представяне на височината на потребителя е нужно при огледа на околните обекти за създаване на правилна перспектива;
- Синхронизиране на реалните и виртуалните движения задава правилно позициониране и се намаляват отклоненията и нуждата за калибрация.

1.8.2. Техники за подобряване задържането на вниманието на потребителя при работа във виртуалния свят

За да се създаде по-задържащо вниманието VR изживяване, може да се използват няколко подхода.

1.8.2.1. Фотореалистично визуализиране

- Използват се усъвършенствани техники за рендиране като глобално осветяване, с което се повишава реализмът, за сметка на производителността;
- Светлината може да се променя в зависимост от това какво е времето в околната среда, тоест възможно е да има симулации относно времето;
- С меки сенки с изтъняващи отвътре-навън ръбове се имитира сенките от реалния свят.

1.8.2.2. Дизайн на пространствения звук

- Симулация на акустиката на помещението спрямо неговата форма и материалите на обектите.

1.8.2.3. Интерактивни обекти

- Съответните мебели и архитектурни елементи, стига да имат функциите за това, може да се реализират във виртуалния свят, така че потребителят да взаимодейства с тях. За по-добър реализъм е необходимо по-реалистична физика относно поведението на обектите.

1.8.2.4. Референтни мащаби

- За да можем да придадем добра и реална размерност на обектите, може да се включат познати, всекидневни обекти за сравнение на относителен размер. По-подходящо е да може всичко да се представи в точни мащаби спрямо човешкия.

1.8.2.5. Разказвателни елементи

- При предоставянето на информация като диалогови прозорци се дава контекст за обзавеждането.

1.9. Нови тенденции във VR за визуализация на мебели и архитектура

1.9.1. Развитие на хаптичните устройства

Неотдавнашните разработки в областта на хаптичните технологии подобряват обратната връзка при взаимодействие във виртуалната реалност. По-усъвършенстваните системи за обратна връзка позволяват на потребителите да „усещат“ различни материали и текстури, което добавя ново измерение към визуализацията на мебели и архитектура. Тази технология дава възможност на дизайнерите и клиентите да изпитат повърхностните качества на повърхностите и материалите, с което се подобрява вземането на решения в процеса на проектиране.

1.9.2. Облачно базирано процесиране

Възходът на облачните изчисления трансформира VR преживяванията. Висококачественото VR съдържание може да се предава поточно без мощен локален хардуер, а само точка за пренос на данни. Тази тенденция позволява на клиентите и заинтересованите страни да разглеждат и взаимодействат с проекти, използвайки по-широк кръг от устройства. Също така се улеснява сътрудничеството и споделянето на големи, подробни VR среди.

1.9.3. Фотограметрия

Фотограметрията е науката за извършване на измервания по снимки. Все по-често се интегрира във VR. Тази технология позволява създаването на много подробни, фотореалистични 3D модели на съществуващи пространства или обекти. Дизайнерите могат да включат сканираното от реалния свят в своята VR среда, като съчетават проектираните елементи безпроблемно със съществуващи структури или пейзажи.

1.9.4. Сътрудничество с много потребители

Усъвършенстваните инструменти за съвместна работа в реално време в споделени VR пространства стават все по-усъвършенствани. Тези платформи позволяват на множество потребители да взаимодействат с проекти и да ги променят едновременно, независимо от физическото им местоположение. Тази тенденция е особено ценна в областта на архитектурата, тъй като дава възможност на проектантски екипи, клиенти и заинтересовани страни да се срещат във виртуални пространства, да се разглеждат проекти заедно и да се правят промени в реално време.

ВТОРА ГЛАВА

Преглед, анализ и сравнение на технологиите за създаване на виртуална реалност и тримерно моделиране

В този раздел ще се направи цялостен преглед на съвременните технологии за виртуална реалност (VR) и 3D моделиране, като се акцентира върху приложенията им във визуализацията на архитектурен и интериорен дизайн.

Ще се разгледа хардуерните платформи за VR – от системи от висок клас до решения, базирани на мобилни телефони, като ще се анализира тяхната пригодност за задачи по проектиране и ще се даде обосновка за избора на VR система за създаването на прототип.

След това ще се разгледа софтуерната екосистема включително игрови двигатели за разработване на VR и софтуер за 3D моделиране. Анализът ще има акцент върху избраните платформи и ще се направят сравнения с алтернативи, за да се обясни финалният избор.

Ще се споменат допълнителни технологии за разработка на VR.

Като допълнение ще се направи паралел между видеоигрите и симулаторите, изследвайки как интерактивните техники за забавление могат да подобрят професионалните инструменти за визуализация на проекти.

Това проучване ще даде информация за разработването на прототипа, като гарантира, че изборът е основан на задълбочено разбиране на наличните технологии и техните приложения във визуализацията на архитектурен и интериорен дизайн.

2.1. Хардуерни платформи за виртуална реалност

Хардуерът за виртуална реалност се развива значително през последните години, предлагайки редица възможности, които отговарят на различни нужди и бюджети. Разглеждат се различни хардуерни платформи за VR, като се обръща внимание на системите, базирани на мобилни телефони.

2.1.1. Преглед на хардуерните опции за VR

На пазара за хардуер за виртуална реалност се предлага разнообразна гама от решения – от системи от висок клас, изискващи мощни компютри, до самостоятелни устройства и варианти, базирани на мобилни телефони. Всяка категория има своите силни страни и ограничения, като отговаря на различни случаи на употреба и изисквания.

2.1.1.1. Системи за виртуална реалност от висок клас

Системите за виртуална реалност от висок клас са върха на технология за VR, като се предлагат потапящи изживявания с дисплеи с висока разделителна способност, прецизно проследяване на движението и усъвършенствани контролери. За всяко споменато устройство ще бъде предоставено изображение, което може да включи и допълнителни части.

2.1.1.1.1. Oculus Rift S

Oculus Rift S – наследникът на VR революцията, предизвикана от гаражния прототип на Палмър Люки, продължава да разширява границите на виртуалното присъствие, приближавайки се все повече към мечтата за създаване на метавселена [4]. Най-новият модел от компанията Oculus има следните параметри:

- **Дисплей** – Единичен LCD дисплей с бързо превключване, резолюция 2560x1440;
- **Честота на опресняване** – 80 Hz;
- **Поле на видимост** – Около 110 градуса;
- **Проследяване** – Проследяване тип отвътре-навън с пет камери;
- **Контролери** – Контролери Oculus Touch.

Oculus Rift S (Фигура 2.1) осигурява висококачествени визуализации и прецизно проследяване. Той обаче изисква връзка с мощен компютър.



Фигура 2.1: Хедсет Oculus Rift S с 2 контролера.

2.1.1.1.2. HTC Vive Pro

Vive Pro е доказателство за силата на прецизността във виртуалната реалност, предлагайки прозорец към светове, ограничени само от въображението [6]. Компонентите, които съставят HTC Vive Pro, са:

- **Дисплей** – Двоен AMOLED с обща резолюция 2880x1600;
- **Честота на опресняване** – 90 Hz;
- **Поле на видимост** – Около 110 градуса;
- **Проследяване** – SteamVR проследяване, съвместимо с базови станции;
- **Контролери** – HTC Vive.

HTC Vive Pro (Фигура 2.2) се отличава с висококачествени визуални ефекти и прецизно проследяване в мащаба на помещението, но подобно на Rift изисква високопроизводителен компютър и по-сложна настройка.



Фигура 2.2: Хедсет HTC Vive pro с включена аудио система, 2 контролера и 2 станции за калибрация.

2.1.1.1.3. HP Reverb G2

HP Reverb G2, макар и да изисква връзка с компютър, представлява хедсет с дисплеи с висока резолюция и има участие в екосистемата на Windows Mixed Reality [8]. Reverb G2 се превърна в шампион по яснота, доказвайки, че понякога всичко е в детайлите, когато става въпрос за правдоподобни виртуални светове.

- **Дисплей** – Двоен LCD дисплей с резолюция 2160x2160 за всяко око;
- **Честота на опресняване** – 90 Hz;
- **Поле на видимост** – Около 114 градуса;
- **Проследяване** – Проследяване отвътре-навън с четири камери;
- **Контролери** – Контролери за движение HP.

Reverb G2 (Фигура 2.3) е известен с високата си визуална детайлност, което го прави подходящ за детайлни архитектурни визуализации.



Фигура 2.3: Хедсет HP Reverb G2 с включена аудио система и 2 контролера.

2.1.1.2. Самостоятелни VR хедсети

Самостоятелните VR хедсети предлагат междинно звено между системите от висок клас и мобилните VR системи, като се осигурява пълноценното VR изживяване, без да е необходим отделен компютър или мобилен телефон.

2.1.1.2.1. Meta Quest 2

Освобождавайки виртуалната реалност от оковите на външния хардуер, Meta Quest 2 е реализацията на една дългогодишна мечта, която поставя цели виртуални вселени в пакет, който лесно се пренася [10]. Quest 2 предлага:

- **Дисплей** – Единичен LCD дисплей с резолюция 3664x1920;
- **Честота на опресняване** – До 120 Hz;
- **Поле на видимост** – Около 90 градуса;
- **Проследяване** – Проследяване тип отвътре-навън с четири камери;
- **Контролери** – Контролери Oculus Touch.

Meta Quest 2 (Фигура 2.4) балансира производителност и удобство. Може да работи със сложни VR приложения и е лесен за настройка.



Фигура 2.4: Хедсет Meta Quest 2 с два контролера.

2.1.1.2.2. Apple vision pro

Apple Vision Pro е високотехнологичен хедсет за смесена реалност със система на чип (system on a chip, SoC) apple M1. Vision Pro слива цифров и физически свят в елегантен дизайн [12].

- **Дисплей** – Двойни micro-OLED дисплеи, общо 23 милиона пиксела;
- **Честота на опресняване** – До 100 Hz;
- **Поле на видимост** – По-широко от повечето съществуващи VR слушалки;
- **Проследяване** – Множество сензори за проследяване на околната среда, очите, вътрешни сензори и околна светлина;
- **Контролери** – Няма физически контролери. Разчита се на проследяване на очите и ръцете за въвеждане на данни;
- **Уникални характеристики** – Предаване на видео в смесена реалност; външен дисплей EyeSight; вградена камера, аудио говорители.

Apple Vision Pro (Фигура 2.5) има за цел да се съчетае виртуална и добавена реалност. Цената е много висока дори недостъпна и изисква връзка с външно захранване за продължителна употреба повече от 2 часа.



Фигура 2.5: Хедсет Apple Vision Pro с батерия.

2.1.1.3. VR системи, базирани на мобилни телефони

Системите за виртуална реалност, базирани на мобилни телефони, използват изчислителната мощ и дисплея на смартфоните за създаване на VR изживявания. Тези системи обикновено се състоят от кул хедсет с лещи, в който се поставя съвместим смартфон.

2.1.1.3.1. Google Cardboard

От проект за уикенда до глобален феномен, Cardboard доказва, че понякога с най-простите идеи могат да се отворят най-големите врати, правейки VR достъпно за милиони за една нощ [14]. Функциите на Google Cardboard (Фигура 2.6) включват:

- Съвместимост с широка гама смартфони;
- Проста картонена или пластмасова конструкция. Зависи от модела;
- Базова система от лещи;
- Не са включени вградени контролери. Разчита се на сензорите на телефона или на въвеждане с Bluetooth контролер;
- Ниска цена, което го прави идеален за прости VR преживявания.



Фигура 2.6: VR очила Google Cardboard с поставен телефон.

2.1.1.3.2. Samsung Gear VR

Чрез съчетание на мобилните технологии и иновациите в областта на виртуалната реалност, Gear VR поставя началото на ера, в която високотехнологичните изживявания могат да се получават от устройството в джоба на всеки човек [16]. Samsung Gear VR (Фигура 2.7) предлага:

- Съвместимост с избрани смартфони Samsung;
- Пластмасова конструкция с регулируеми лещи;
- Вграден тъчпад и бутони за управление;
- Софтуерна платформа, базирана на Oculus SDK;
- По-високо качество на изживяването от Cardboard, но е ограничено само до определени модели телефони на Samsung.



Фигура 2.7: VR очила Google Cardboard с поставен телефон.

2.1.2. Сравнение на стандартен Хедсет за виртуална реалност с VR системи за мобилен телефон

За да се илюстрират разликите, ще се направи сравнение между Oculus Quest 2 със Samsung Gear VR. В таблица 2.1 са описани параметрите на всички разглеждани в този документ системи за виртуална реалност [5], [7], [9], [11], [13], [15], [17].

1. Производителност

- **Oculus Quest 2** има специализиран VR процесор, оптимизиран за VR приложения.
- **Samsung Gear VR** разчита на процесора на смартфон, който е податлив на прегряване.

2. Дисплей

- **Oculus Quest 2** има резолюция 3664x1920 и честота на опресняване до 120 Hz.
- **Samsung Gear VR** зависи от телефона, който се ползва. По-старите съвместими телефони имат по-малко системни ресурси, са с по-ниска разделителна способност и честота на опресняване на дисплея.

3. Проследяване

- **Oculus Quest 2** има 6 DOF (Degrees Of Freedom, степени на свобода) – 3 DOF съответно за главата и ръцете, където са контролерите.
- **Samsung Gear VR** има само 3 DOF проследяване за главата.

4. Контролери

- **Oculus Quest 2** използва специални контролери с прецизно проследяване.
- **Samsung Gear VR** може да използва вграден тъчпад или отделен 3DOF контролер. За съжаление контролерът няма сензори.

5. Екосистема от приложения и устройства

- **Oculus Quest 2** съдържа голяма библиотека от специализирани VR приложения и игри.
- **Samsung Gear VR** има по-ограничен избор от мобилни VR приложения. Включени са функционалности, предоставени от Windows Mixed Reality.

6. Цена

- **Oculus Quest 2** е изначално по-скъп, но не изисква закупуването на допълнителен хардуер.
- **Samsung Gear VR** има по-ниска цена при закупуване, но изисква съвместим смартфон Samsung от поредицата Galaxy, Galaxy Note/Edge, като самите модели да не са по-стари от серия S6.

	Oculus Rift S	HTC Vive Pro	HP Reverb G2	Meta Quest 2	Apple Vision Pro	Google Cardboard	Samsung Gear VR
Дисплей/и и Резолюция	Единичен, 2560 x 1440	Двоен, 4896 x 2448	Двоен, 4320 x 2160	Единичен, 3664 x 1920	Двоен, 7320 x 3200	Зависи от параметрите на смартфона	Зависи от параметрите на смартфона
Честота	80 Hz	120 Hz	90 Hz	До 90 Hz	До 100 Hz		
Полезрение	Ок. 115°	Ок. 120°	Ок. 114°	Ок. 97°	Ок. 120 °		
Сензори	Сравнете с описанието на устройството					Опционален	Опционален
Контролери	Да, собствени	Да, собствени	Да, собствени	Да, собствени	Не, ръчни жестове		
Цена	1000 Лв.	2000 Лв.	1500 Лв.	От 600 Лв.	10000 Лв.	От 25 до 75 Лв.	140 или 220 Лв.

Таблица 2.1: Сравнение на дискутираните VR хедсети и техните характеристики.

2.1.3. Обосновка за избора на мобилен телефон и VR очила

За разработката на прототип, насочен към визуализация на мебели и архитектура, ще се направи избор да се използва система за виртуална реалност, базирана на мобилен телефон с VR очила. Това решение се основава на долупоменатите фактори.

- 1. Достъпност** – По-лесно е да се достъпи по отношение на цената както и необходимите технически познания.
- 2. Преносимост** – Устройството тежи около 325 грама без смартфона да е прикачен, с което се позволява лесно транспортиране и демонстрации на място.
- 3. Бързо прототипиране** – Понеже тези приложения се разработват в често срещани десктоп среди и се компилират и пакетират, за да работят на телефони, това позволява бърза разработка и множество итерации на VR приложения.
- 4. Широко разпространена съвместимост** – Има потенциал за достигане до по-голяма потребителска база заради по-голяма съвместимост с повече смартфони.
- 5. Задоволителна производителност** – Съвременните смартфони могат да визуализират детайлни 3D среди със сложно осветление и сенки както и допълнителни графични обработки.
- 6. Бъдеща скалируемост** – Основните функционалности могат да бъдат адаптирани към по-усъвършенствани VR платформи за по-детайлна и задълбочена работа, обхващайки останалата част от пазара.

Избирайки виртуална реалност, чиято среда се визуализира на мобилни телефони, се постига баланс между достъпност, производителност и ефективност на разработката, което съответства на нуждите на дипломната работа за създаване на прототип за визуализация на артикули, основно мебели и интериор във виртуална реалност.

2.2. Софтуер за разработване на VR приложения

За разработването на приложения за виртуална реалност се изисква специализирани софтуерни инструменти най-вече игрови двигатели за създаване на интерактивни среди и софтуер за 3D моделиране за изработване на обекти, използвани в тези среди. В този раздел ще се разгледат основните възможности в двете категории.

2.2.1. Преглед на популярни 3D енджини за разработка на игри

Двигателите за игри са цялостни комплекти от софтуерни инструменти (Software Development Kit, SDK), които предоставят на разработчиците необходимите инструменти за създаване на интерактивни 3D среди. Макар че са предназначени за разработката на видеоигри, тези енджини са намерили широко приложение при разработката на VR благодарение на мощните си възможности за рендиране и симулиране на физични симулации.

2.2.1.1. Unity Engine

Unity е един от най-популярните игрови енджини, известен със своята гъвкавост и лекота на използване. Поддържат се множество платформи включително мобилни VR среди. Използва се C# като основен език за програмиране и се предлага голям магазин за активи с предварително създадени модели и скриптове. Предоставят се надеждни набори от инструменти за разработка на VR (SDK) и разполага с удобен за ползване интерфейс, подходящ както за начинаещи, така и за професионалисти [18].

2.2.1.2. Unreal Engine

Unreal Engine, разработен от Epic Games, е известен със своята реалистична графика. Поддържат се различни платформи включително VR, използва се езика за програмиране C++ и визуалният скриптов инструмент Blueprint. Предоставят се комплексни възможности за рендиране дори на безплатните версии и се предоставя мощен редактор на материали и визуални ефекти за създаване на сложни сцени. Разполага с голям пазар за обекти (Asset Store) и плъгини [19].

2.2.1.3. CryEngine

CryEngine, създаден от Crytek, е широко признат за впечатляващите си визуални възможности. Навремето, когато на пазара е излязла играта Crysis, е било трудно тя да работи с добра производителност. Поддържа се разработката на VR приложения. Използва се програмният език C++ и скриптовият език Lua. Предлагат се сложни системи за осветление и частици. Има специално разработена, базирана на възли и ребра, тоест графова система за визуално скриптиране, наречена Flowgraph [20].

2.2.1.4. Анализ на Unity Engine

Средата за разработка Unity се откроява по няколко причини в контекста на разработването на VR приложения от останалите инструменти [42].

- **Съвместимост между различни платформи** – Unity поддържа широк набор от VR платформи включително мобилни VR системи както и други, несвързани с виртуална реалност среди;
- **Лесно оптимизиране на производителността** – В Unity се предлагат различни инструменти за оптимизиране на производителността, което е значимо за разработката на мобилни VR системи, понеже са с по-ниска производителност;
- **Специфични за разработката на VR функции** – В Unity се предоставят специфични за VR компоненти и дадености, с които се опростява разработката на приложения за виртуална реалност;
- **Голяма общност и ресурси** – Разполага се с огромна потребителска база, което води до изобилие от уроци, форуми и ресурси от трети страни, улесняващи усвояването на знания и техники за разработка на всякакви приложения, включително за VR;
- **Редовни актуализации** – Екипът зад разработването на Unity Engine често актуализира поддръжката на инструментите за разработка на VR и е в крак с модата на бързо развиващите се VR технологии.

2.2.1.5. Сравнение На Unity Engine с Unreal Engine

Въпреки че и Unity, и Unreal са способни да създават висококачествени VR изживявания, те имат различни характеристики. В таблица 2.2 са описани и сравнени общи характеристики на няколко игрови двигателя [46].

1. Лесно използване

- **Unity** като цяло се счита за по-лесен за изучаване особено за начинаещи.
- **Unreal** е по-трудно да се научи, но има мощна система за визуално скриптиране, която понижава нивото на трудност за изучаване.

2. Качество на графиката

- В **Unity** има възможности за висококачествени графики, но може да се изисква повече настройки и работа от програмиста.
- **Unreal** е известен с фотореалистичните си графики, предоставени директно на потребителите, но това не винаги е най-подходящото решение поради началната неопитност на нови потребители и очакваната липса на оптимизации.

3. Производителност

- **Unity** като цяло е ефективна платформа за мобилни приложения и е предпочитана среда за такива нужди.
- **Unreal** в исторически аспект е по-взискателна платформа, но технологиите за мобилни устройства продължават да се подобряват.

4. Език за програмиране

- В **Unity** се използва C#, който има много повече обучителни ресурси, правейки го по-широко използван и по-лесен за изучаване.
- В **Unreal** се използва C++, който е по-сложен за изучаване понеже е на по-ниско ниво, но има много голям потенциал за създаването на по-производителен софтуер.

5. Инструменти за разработка на VR

- В **Unity** се предоставя цялостен набор от VR инструменти.
- В **Unreal** се поддържа на добро ниво VR инструментите особено за силни корпоративни среди. Но в този случай, за тези функции се изисква плащането на значителна сума за лиценз за използване.

	Unity Engine	CryEngine	Unreal Engine 3	Unreal Engine 4
Поддържани платформи	MS windows, MacOS, Xbox, PlayStation, IOS, Android, Web, Nintendo Wii U & 3DS, Gear VR, Android TV, Tizen OS	MS windows, Linux, MacOS, PlayStation, Xbox, Nintendo Wii U, IOS, Android	MS windows, Linux, MacOS, PlayStation, Xbox, Nintendo Wii U, IOS, Android	MS windows, Linux, MacOS, PlayStation, Xbox, Nintendo Wii U, IOS, Android
Програмен език	C#, C/C++	C++	C++, Assembler	C++, Assembler
Скриптов език	UnityScript, Boo (Изоставени), HLSL & GLSL (Shaders)	Lua	UnrealScript, Python, HLSL & GLSL (Shaders)	
Безплатна версия	Да	Да	Да	Да

Таблица 2.2: Сравнение на дискутираните 3D енджини и техните характеристики.

2.2.1.6. Обосновка за избора на Unity със C#

За разработката на прототип се избира Unity със C# по няколко причини [50]:

- **Фокус върху мобилното VR приложение** – 3D енджинът Unity се предпочита за разработка на приложения за мобилни платформи;
- **Лекота на разработване** – C# и интуитивният интерфейс на Unity дават по-добри възможности за бързо създаване на прототипи;
- **Широка поддръжка на VR компонентите** – Пакетът от VR инструменти на Unity предоставя необходимите инструменти за разработване приложения за виртуална реалност;

- **Голяма общност** – Богатият набор от учебни ресурси и пазарът за предмети, налични в Unity, помагат за по-бързо учене и прототипиране;
- **Бъдеща скалируемост** – Междуплатформеният характер на Unity позволява потенциално разширяване към други платформи, поддържащи VR функционалности.

2.2.2. Преглед на софтуери за 3D моделиране

Софтуерът за 3D моделиране е програма, служеща за създаването на обектите, използвани във VR средите. Различните видове софтуер задоволяват различни нужди не само в архитектурата и дизайна. Затова подбор на софтуер, който е достъпен и има необходимите функционалности, е ключово.

2.2.2.1. CAD софтуер

Софтуерът за проектиране с помощта на компютър (Computer Aided Design, CAD) се използва предимно за прецизни технически чертежи и 3D модели с по-ниско ниво на артистична детайлност [21], [22].

2.2.2.1.1. AutoCAD

- Има широко приложение в промишлеността и инженерните науки за 2D и 3D технически чертежи;
- Предоставят се прецизни инструменти за измерване и анотиране (поставяне на бележки);
- Поддържа се параметрично чертане и 3D моделиране;
- Има голяма библиотека от готови архитектурни и инженерни компоненти;
- Труден е за научаване, но дава много възможности за технически проекти.

2.2.2.1.2. SketchUp

- Известен е с удобния си интерфейс и лекотата на изучаване;
- Подходящ е за концептуално 3D моделиране в архитектурата;
- Съдържа обширна библиотека, състояща се от създадени от потребители 3D модели;
- Има по-ниска прецизност от AutoCAD, но по-бързо се работи с него;
- Поддържат се различни плъгини, за да се разширява функционалността.

2.2.2.2. Софтуер за BIM

Софтуерът за информационно моделиране на сгради (Building Information Modeling, BIM) се използва за създаване на цялостни богати на данни модели на сгради [24], [25].

2.2.2.2.1. Revit

- Това е водещ в индустрията BIM софтуер, използван предимно в архитектурата и инженерните науки;
- Ползва се за създаване на богати на данни модели, включващи структурна и механична, електрическа и водопроводна и канализационна (Mechanical, Electrical and Plumbing, MEP) информация;
- Съдържа мощни инструменти за съвместна работа върху големи проекти;
- Реализирани са параметрични компоненти за ефективни и бързи промени в проекта;
- Труден е за изучаване, но дава добра основа за умения за проектиране на сгради.

2.2.2.2.2. ArchiCAD

- Това е BIM софтуер, който е специално разработен за архитекти;
- Има интуитивен интерфейс с фокус върху архитектурното проектиране;

- Силно развит е в областта на 3D визуализацията и документацията;
- Поддържа се работа в екип и сътрудничество в облачното пространство;
- Дава добър баланс между лекота на ползване и професионални инструменти.

2.2.2.3. Софтуер за 3D моделиране с общо предназначение

Тези софтуерни приложения предлагат по-голяма гъвкавост при създаването на подробни 3D модели за различни цели включително за VR, като има по-малко фокус върху точни измервания, защото фокусът тук се поставя на креативното моделиране [27], [28].

2.2.2.3.1. Blender

- Безплатен софтуер с отворен код, съдържащ подробен пакет от инструменти за създаване на 3D модели;
- Наборът от инструменти включва възможности за моделиране, текстуриране, анимация и рендиране. Възможно е създаването на летливи частици чрез система за симулация на частици (Particle System);
- Общността около Blender е активна и се предоставят многобройни учебни ресурси и плъгини;
- Поддържат се скриптове на скриптовия език Python за персонализиране на елементи;
- Софтуерът непрекъснато се подобрява и често се актуализира.

2.2.2.3.2. 3ds Max

- Софтуерът е стандартен в индустрията за 3D моделиране, анимация и рендиране;
- Предоставят се мощни инструменти за моделиране с широка поддръжка на плъгини;

- Силите на този софтуер са в архитектурната визуализация и продуктивния дизайн;
- Добре се интегрира с други продукти в екосистемата на Autodesk;
- За ползване от потребители се основава на базата на абонаментен модел.

2.2.2.4. Анализ на Blender

Софтуерът за 3D моделиране Blender се откроява от останалите възможности по няколко причини [47], [51]:

- **Цена** – Софтуерът е безплатен и с отворен код, което го прави достъпен за всички потребители;
- **Универсалност** – Предлага се пълен комплект от инструменти за създаване от най-малките до най-комплексните обекти с голям детайл – от моделиране през текстуриране до осветяване и рендиране;
- **Общност** – Голямата и активна общност предоставя ресурси и поддръжка;
- **Редовни актуализации** – Подобренията и новите функции са чести;
- **Възможност за персонализиране** – Има възможност за разширяване на интерфейса и функционалността чрез Python скриптови програми и плъгини;
- **Съвместимост с VR** – Blender има добра поддръжка за създаване на специфични за виртуалната реалност съдържание.

2.2.2.5. Сравнение на Blender с 3ds Max и AutoCAD

За да се сравнят избраните от този набор софтуери за 3D моделиране, ще се разгледат Blender, 3ds Max и AutoCAD спрямо следните параметри:

1. Цена

- **Blender** е безплатен за ползване и с отворен код.
- За **3ds Max** и **AutoCAD** се изисква заплащането на Абонамент. Лошото нещо е, че това се явява наистина значителен разход – цената е много висока.

2. Лекота на научаване

- **Blender** е практически лесно да се научи.
- **3ds Max** е по-лесно да бъде научено от **AutoCAD**, но все пак изисква да се отдели време.
- **AutoCAD** е трудно да се научи за работа в кратък период от време. Изисква се задълбочена работа за научаване на пълната функционалност.

3. Гъвкавост и приложимост

- **Blender** има висока гъвкавост във функционалността си и е приложим за общи 3D задачи.
- **3ds Max** има универсален дизайн, чийто фокус е върху развлекателната индустрия и дизайна.
- **AutoCAD** се специализира за технически чертежи и CAD.

4. Употреба в индустрията

- **Blender** е разпространен в различни индустрии. Предимно е популярен в независими и малки студия.
- **3ds Max** е широко използван в развлекателната индустрия и за архитектурни визуализации.
- **AutoCAD** е стандартното приложение в инженерните науки и архитектурата.

5. Създаване на VR съдържание

- **Blender** подобрява поддръжката на инструментите си за създаване на виртуална реалност с всяка актуализация. Общността е в крак с модата;
- **3ds Max** има силни възможности за създаване на VR съдържание.
- **AutoCAD** има ограничена функционалност за виртуална реалност. Почесто се използва в комбинация с друг софтуер.

2.2.2.6. Обосновка на избора на Blender

За разработката на прототип се избира Blender поради следните причини:

- **Цена** – Понеже Blender е безплатен и с отворен код софтуер, той съответства на ограниченията в ресурсите на разработката;
- **Гъвкавост и приложимост** – Софтуерът за моделиране предоставя всички необходими инструменти за създаване на готови за вмъкване във виртуална реалност 3D модели;
- **Активна разработка** – Редовните актуализации осигуряват съвместимост с развиващите се стандарти за VR;
- **Подкрепа от общността** – Тя предоставя огромни ресурси за обучение и решаване на проблеми както и пазар за готови модели и текстури;
- **Интеграция с Unity** – Blender има добра съвместимост за импортиране на модели в игровия двигател Unity;
- **Бъдещо развитие** – Нарастващото приемане в различни индустрии предполага дългосрочна жизнеспособност на този софтуер за моделиране.

Този избор позволява да се създаде висококачествени 3D модели за прототипа, като същевременно се запазва гъвкавостта на процеса на разработка.

2.3. Допълнителни технологии и езици за разработване на VR приложения

Макар че игровите енджини и софтуерът за 3D моделиране са в основата на разработването на приложения за виртуална реалност, няколко допълнителни технологии и инструменти също могат да имат роля в създаването на цялостни и поглъщащи VR изживявания. В този раздел са разгледани тези допълнителни технологии, техните приложения и значението им при разработването на виртуална среда.

2.3.1. SDK и API за VR

Комплектите за разработка на софтуер (SDK) и интерфейсите за програмиране на приложения (API) са основни инструменти, с които се позволява на разработчиците да взаимодействат с хардуера за виртуална реалност и да се създават оптимизирани VR приложения [28], [29], [30], [31].

2.3.1.1. OpenVR/SteamVR

- Разработено е от Valve Corporation;
- Осигурява се стандартен API за достъп до VR хардуер;
- Поддържа се широк набор от VR хедсети и контролери;
- Позволява разработване на VR приложения за множество платформи.

2.3.1.2. Oculus SDK

- Разработено е първоначално от Oculus, после усвоени от Facebook, който вече са преименувани на Meta;
- Оптимизирано е за хардуера на Oculus;
- Осигурява се достъп на ниско ниво до специфични за Oculus устройството функции;
- Включват се инструменти за оптимизиране на производителността и използване на социални медии чрез API.

2.3.1.3. Google VR SDK

- Поддържа се разработка за платформите Google Cardboard и Daydream;
- Осигурява се API за проследяване на главата, стереоскопично рендиране и въвеждане от потребителя;
- Има междуплатформена поддръжка за Android и IOS.

Тези SDK и API често са интегрирани в игровите енджини като Unity и Unreal и по този начин е опростен процесът на разработка и се осигурява съвместимост с различните VR платформи.

2.3.2. 3D файлови формати и стандарти

Обменът на 3D данни между различни софтуери и платформи е от значение при разработването на VR приложения. Няколко широко използвани файлови формати и стандарти улесняват този процес [32], [33], [34], [35].

2.3.2.1. FBX (Filmbox)

- Това е широко поддържан формат за обмен на 3D данни;
- Съхраняват се йерархии, анимации и материали;
- Обикновено се използва за прехвърляне на модели от 3D софтуер към игрови двигатели.

2.3.2.2. glTF (GL Transmission Format)

- Това е стандарт с отворен код за ефективно предаване на 3D сцени и модели;
- Поддържат се PBR материали, анимации и структура на сцената;
- Става все по-популярен за уеб базирани и мобилни VR приложения.

2.3.2.3. OBJ

- Това е прост и широко поддържан формат за 3D геометрия;
- Ограничена е поддръжката за материали и липсва поддръжка за анимации;
- Често се използва за статични модели и архитектурни елементи.

2.3.2.4. COLLADA

- XML-базиран формат за обмен на цифрови обекти;
- Поддържат се сложни сцени с анимации и физика.

2.3.3. Инструменти за 360-градусови снимки и видео

360-градусовите медийни елементи позволяват повърхностното разглеждане на сцените с ограничени функции за взаимодействия. Това е подходящо при изработката на демонстративни VR изживяванията. Следните инструменти улесняват създаването и интегрирането на това съдържание.

2.3.3.1. 360-градусови камери

- С устройства като Insta360, GoPro Max или Ricoh Theta се заснема поглъщащи зрението и вниманието панорамни изображения и видеоклипове;
- Предоставя се бърз начин за създаване на реалистична среда за VR.

2.3.3.2. Софтуер за фотограметрия

- С инструменти като Agisoft, Metashape или RealityCapture се създават 3D модели от множество снимки;
- Полезно е за заснемане на реални среди или обекти за направа на модели и вграждането им във виртуална среда.

2.3.3.3. Софтуер за съшиване на изображения

- С програми като PTGui или Autopano се сглобява множество от изображения в 360-градусови панорами със или без видими граници;
- Тези програми служат за създаване на висококачествени 360-градусови изображения от изображения, заснети на DSLR фотоапарат.

Тези инструменти позволяват на разработчиците да включат реалистични, поглъщащи среди в своите VR приложения, засилвайки усещането за присъствие на потребителите, както и skybox-ове (обгръщащо 3D фоново изображение или анимация).

2.3.4. Инструменти за проектиране, специфични за VR

С развитието на VR технологиите се появяват нови инструменти, които позволяват на дизайнерите да работят директно във виртуална среда, предлагайки уникални възможности за 3D моделиране и пространствено проектиране.

2.3.4.1. Gravity Sketch

Gravity Sketch е платформа за 3D проектиране, базирана на VR, която позволява на потребителите да се създават и манипулират 3D модели директно във виртуалното пространство [36].

- Предоставя се интуитивни инструменти за 3D скициране и моделиране;
- Поддържа се съвместно проектиране в споделени виртуални пространства;
- Има възможност за експортиране в стандартни 3D файлови формати за по-нататъшно обработване или използване в други VR приложения;
- Особено полезен е за концептуално проектиране и бързо създаване на прототипни модели във VR.

2.3.4.2. Medium

Medium е приложение, което вече е част от Creative Suite на Adobe. Представява инструмент за виртуална реалност за скулптуриране, който позволява детайлно 3D моделиране във виртуална среда [37].

- Предлага се набор от инструменти за скулптуриране, имитиращи техники за скулптуриране със съответните инструменти в реалния свят;
- Поддържа се работа с детайли с висока разделителна способност;
- Може да се прави интуитивно мащабиране и манипулиране на 3D обекти;
- Полезен е за създаване на органични елементи и обекти и детайлни модели за VR среди.

2.4. Прилики и разлики между видеоигрите и симулаторите

Световите на видеоигрите и професионалните симулатори особено тези, използвани в архитектурния и интериорния дизайн, са сближени в няколко общи точки. Това сближаване се дължи най-вече на напредъка в технологиите за визуализация в реално време, виртуалната реалност и дизайна на потребителския интерфейс. Въпреки че видеоигрите се създават предимно за забавление, а [архитектурните] симулатори – за професионална употреба и понякога – за учебна и развлекателна цел, разграничението между тези категории става все по-малко ясно.

Както при видеоигрите, така и при симулаторите се използват сходни технологии за създаване на завладяващи, интерактивни 3D среди. Въпреки това техните крайни цели, целеви аудитории и специфични функции се различават значително. Тези прилики и разлики са по-очевидни отколкото не за проектирането и прилагането на тези инструменти в развлекателен или в професионален контекст [44], [45].

2.4.1. Общи елементи в игрите и симулаторите

- 1. Интерактивни 3D среди** – И в двата случая се използва поглъщащи вниманието 3D пространства с възможност за навигация на потребителя;
- 2. Визуализация в реално време** – И двата продукта изискват ефективна графична обработка в реално време и то с достатъчно добра производителност и видимост за безпроблемна работа на потребителите;
- 3. Вход от потребителя и обратна връзка** – И в двата софтуера се разчита на взаимодействието с потребителя и да се осигурява незабавна визуална, аудио или хаптична обратна връзка;
- 4. Физическа симулация** – В много игри и симулатори се включва реалистична физика за постигане на автентичност на околната среда;
- 5. Персонализиране** – И в двата вида софтуер често се позволява на потребителите да променят средата или обектите във виртуалното пространство посредством определени взаимодействия.

2.4.2. Уникални аспекти на симулаторите

- 1. Прецизност и точност** – В симулаторите се дава приоритет на точните измервания и представянето на мащаба;
- 2. Ограничения, свързани с реалния свят** – Симулаторите трябва да се придържат към физическите ограничения и размери за по-голяма достоверност;

- 3. Интеграция на професионални инструменти** – Обикновено има добра съвместимост със стандартните за индустрията CAD и BIM софтуери;
- 4. Фотореалистично визуализиране** – Акцентът в симулаторите е поставен върху високодетайлното и реалистично визуално представяне;
- 5. Функции за съвместна работа** – В някои софтуери може да се правят многопотребителски сесии и презентации за клиенти. Това е по-различно от игра по мрежата дали в кооперативен, или „един срещу друг“ вариант;
- 6. Възможности за експортиране на данни** – Възможност за генериране на технически чертежи, спецификации или 3D обекти най-вече се случва в симулаторите.

2.4.3. Преодоляване на разликата между развлекателните и професионалните инструменти

- 1. Превръщането на обучителния процес в игра (Геймификация, Gamification)** – В образователния процес може да се използват игрови елементи за преподаване на базови принципи и умения за работа;
- 2. Удобни за ползване интерфейси** – Добра идея е да се заемат интерфейси за интуитивното управление от игрите поради тяхното сходство и естествена работа, за да се направят професионалните инструменти по-достъпни и леки за ползване;

- 3. Виртуални разходки** – Използват се технологии от игровите двигатели, за да се правят поглъщащи презентации за клиенти чрез виртуални разходки точно като демонстрациите (Demos);
- 4. Проектиране с помощта на обществото** – Използват се платформи за споделена работа, подобни на други ентерпрайз продукти, и работни среди за съвместни проекти, подобно на Steam работилницата (Workshop).

ТРЕТА ГЛАВА

Проектиране и разработка на прототип на виртуална среда и обекти

3.1. Описание на проблема и цели

3.1.1. Описание на проблема, който се решава

Традиционните методи за визуализация на интериорни проекти като 2D рендери или статични 3D модели, гледани на екран, често не се предоставя истинско усещане за пространство, мащаб и атмосфера. Тези ограничения могат да доведат до недоразумения, което довежда до неудовлетвореност от крайния резултат. Освен това със статичните визуализации не се успява да се пресъздаде динамичен характер на обитаването на дадено пространство, с което се ограничава способността на клиента да се ангажира напълно с предложения проект и да го разбере и възприеме.

Основният проблем, който се стреми да се разреши, е разминаването между концептуалния интериорен дизайн и способността на клиента да изпита и да взаимодейства с този дизайн преди изпълнението му. Това разминаване може да доведе до:

- Неправилно тълкуване на пространствените взаимоотношения и мащаба;
- Трудности при оценката на потока и функционалността на пространството;
- Ограничено разбиране на взаимодействието на различните елементи на дизайна;
- Невъзможност за лесно визуализиране на алтернативи или модификации на дизайна.

3.1.2. Основни цели и задачи на прототипа

За да се решат тези проблеми, се разработва VR приложение за визуализация на интериорен дизайн. Основните цели на този прототип са:

1. Да се създаде поглъщаща, интерактивна 3D среда, която точно представя проектирано интериорно пространство;
2. Да се предостави на потребителите по-интуитивно разбиране на пространствените взаимоотношения и мащаба;
3. Да се даде възможност на потребителите да изследват и взаимодействат с проектираното пространство по естествен и ангажиращ начин;
4. *(По избор)* Да се предложи платформа за лесно визуализиране на алтернативни и модифицирани версии на обекти.

Конкретните задачи, които прототипа се цели да изпълни, включват:

- Да се визуализира реалистична среда на стая в подходящ мащаб. Стаята да включва под, стени, таван и мебели. Да има поставени текстури, наподобяващи боя, тапети, плочки и други материали върху повърхности;
- Да се разработят и моделират 3D обекти на битови мебели;
- *(По избор)* Да се осъществи и вгради потребителски интерфейс във виртуалното пространство;
- Да се включат интерактивни елементи, които да позволят на потребителите да променят свойствата на дизайна;
- Да се осъществи плавно и интуитивно движение на потребителите във виртуалната среда;
- Да се осигури реалистично осветление, за да се засили усещането за потапяне и да се представи точно обстановката;
- Да се включат аудио елементи, за да се подобри реализмът на виртуалната среда.

Чрез постигането на тези цели и задачи с VR приложението ще се осигури по-ангажиращ и информативен инструмент за визуализиране на интериорни проекти, като се дадат нови възможности за визуализиране на концептуален дизайн и разбирането на клиента. Прилагайки този подход има потенциал да се намали необходимостта от преработки на проектите и в крайна сметка да се доведе до по-ефективни и ефикасни процеси на интериорно проектиране.

3.2. Фаза на проектиране

3.2.1. Концептуален проект

Концептуалният дизайн на VR приложението се фокусира върху създаване на поглъщащо и интуитивно преживяване за потребителите, за да се изследват взаимодействията с интериорни проекти. В основните елементи на дизайна се включват:

- **Фокус върху една стая** – Приложението първоначално ще показва само една стая, за да се демонстрира ефективно концепцията, като същевременно се запази обхвата на разработката;
- **Реалистично представяне** – Виртуалната среда ще има за цел да наподобява максимално реалните пространствени размери и свойства на материали;
- **Ориентиран към потребителя подход** – При проектирането ще се дава приоритет на лесната употреба и естественото взаимодействие, за да се осигури достъпност за клиенти, които е много вероятно да не са запознати с технологията за виртуална реалност.

3.2.2. Дизайн на потребителското изживяване (UX)

Дизайнът на UX за това приложение има за цел да създаде безпроблемно и ангажиращо преживяване за потребителите. В основните съображения са включени:

- **Интуитивна навигация** – Потребителите трябва да могат да се движат във виртуалното пространство по естествен начин, като се имитира максимално точно движението в реалния свят;
- **Ясна визуална обратна връзка** – Приложението ще предостави визуални сигнали, за да помогне на потребителите да разберат позицията си в пространството и как могат да взаимодействат с него;
- **Минимизиране на страничните ефекти** – Ще се внимава да се намалят факторите, които могат да допринесат за дискомфорт, предизвикан от употребата на устройства за виртуална реалност като внезапни движения или непостоянна честота на кадрите;
- **Скалируемост** – Въпреки че първоначално се фокусират усилия върху една стая, дизайнът на UX ще включи бъдещо разширяване до визуализация на няколко стаи или на цяла къща.

3.2.3. Дизайн на интерфейса

Дизайнът на интерфейса ще се съсредоточава върху създаване на ненаатрапчив, но функционален потребителски интерфейс, който допълва функционалността на виртуалната средата. От някои ключови елементи се обмисля да се разработи:

- **Минимална показана информация на дисплея (Heads-Up Display, HUD)** – На дисплея ще се показва минимална информация, ако изобщо;

- **Контекстно чувствителни събития** – Елементи на интерфейса ще се появят, когато е необходимо, примерно се подсилва визуална подсказка за взаимодействие, когато потребителят се приближи до интерактивен обект;
- **(По избор) Система от менюта** – За достъп до различни функционалности или опции ще се предостави проста и лесна за навигация система от менюта;
- **Визуална Консистентност** – Дизайнът на потребителския интерфейс ще съответства визуално с цялостната естетика на виртуалната среда, за да се подобри чувството на свързаност на елементите.

В тази фаза на проектиране се полагат основите за разработването на VR приложението, като гарантира, че се създава лесен за ползване, ефективен инструмент за визуализация на интериорен дизайн. С тези насоки се направлява разработката, като крайният продукт ще бъде близък до поставените цели.

3.3. Ключови решения и задачи

3.3.1. Избрани технологии и инструменти

Като се има предвид предишният анализ в избрани части от втора глава, тук само се обобщава накратко избора на технологиите, избрани за разработка на приложението:

- **Хардуер** – Система за виртуална реалност, базирана на мобилен телефон, с очила за виртуална реалност, избрани заради достъпността;

- **Платформа за разработка (Game Engine)** – Unity, избрана заради добрите си функционалности за разработка на VR и широката подкрепа от общността;
- **Софтуер за 3D моделиране** – Blender, избран заради своята гъвкавост и съвместимост с Unity.

3.3.2. Принципи и подходи за проектиране

Въз основа на избраните технологии и целите на проекта ще се използват следните принципи и подходи за проектиране:

1. Оптимизация за мобилни устройства

Предвид избора ни на мобилен хардуер за виртуална реалност, оптимизирането на производителността е ключова. Тук е необходимо:

- 3D моделите да имат ефективен брой полигони;
- текстурите да са оптимизирани;
- Внимателно да се управляват системните извикванията за рисуване.

2. Unity-ориентирано разработване

Ще се използват вградените функции на Unity и неговият графичен поток за рисуване на графика, за да се оптимизира разработката.

- Може да се използва система за просто осветление на Unity за реалистично и същевременно оптимално по производителност осветление;
- Използва се вградената системата за потребителски интерфейс на Unity за всички необходими елементи на интерфейса;
- Използва се вградената поддръжка за VR на Unity – Unity XR за внедряване на мобилна VR технология.

3. Модулен дизайн

За да се улесни бъдещо разширение и модификация в приложението, Може да се възприеме модулен подход, където:

- Се създават модели за многократна употреба на мебели и елементи на помещения;
- Се реализира гъвкава система за оформление на стаи, която може лесно да променя или разширява, тоест готови параметрични стаи.

4. Итеративно разработване

Ще се използва итеративен процес на разработка, където първо усилия се съсредоточават върху основните функционалности, а по-късно върху по-сложните.

- Приоритизира се основната визуализация и навигация в стаята;
- Поетапно се добавят функции за разполагане на мебели и персонализиране;
- Оставят се по-сложни функции като усъвършенствано осветление или звук за по-късни версии на приложението, ако времето и ресурсите позволяват това.

Като се придържаме към тези принципи и подходи, създаваме прототип, който ефективно демонстрира потенциала на VR визуализации на интериорен дизайн, като същевременно работи в рамките на ограниченията на избраната от нас мобилна VR платформа и се решават основните проблеми, идентифицирани в изложението на проблема.

3.4. Средата за разработка Unity

Unity [42] е интегрирана среда за разработка (IDE), специализирана в създаването на интерактивни тримерни приложения. Често се използва в игралната индустрия и предоставя интуитивен интерфейс за създаване на

визуализации, анимации и игри. Unity поддържа различни платформи като Web, Android, iOS, PC и други, позволявайки на разработчиците да обхванат множество устройства с едно и също приложение. Средата има богата екосистема от инструменти и ресурси включително графичен редактор, система за анимации, интегрирани физически двигатели и множество плъгини за разширение на функционалността. Unity Engine е известен със своята гъвкавост и мощност и е предпочитан избор за множество проекти.

3.4.1. Преглед на Unity Hub

По-нататък в разработката ще бъде направен подробен преглед на Unity и неговите функции. Тук се обръща се внимание на Unity Hub (Фигура 3.1).

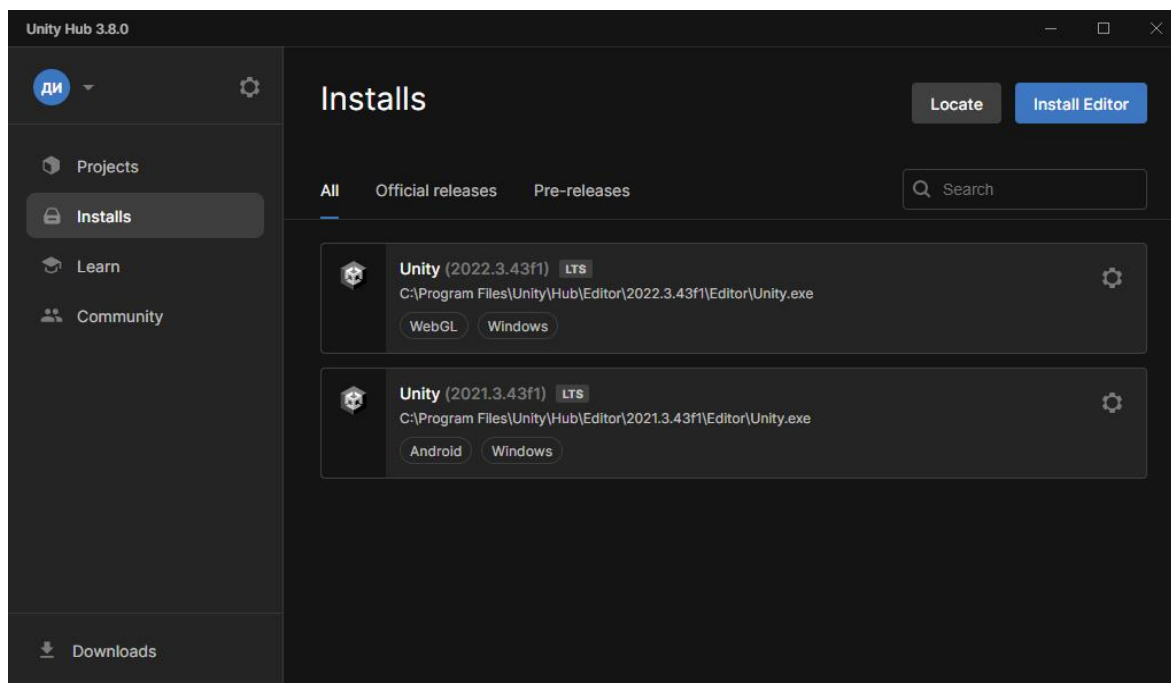
Unity Hub е инструмент за управление и навигация, предоставен от Unity Technologies, с цел подобряване на процеса на разработка в Unity [18]. Той служи като централен интерфейс за управление на множество версии на Unity Editor, организиране на проекти и достъп до различни услуги на Unity и функционалности. Посочени са част от тях.

- **Управление и създаване на проекти. Избор на версия на Unity Editor** – Възможно е да се създават нови проекти на Unity с лекота, като се зададе желана от потребителя версия на Unity Editor и да се изберат различни шаблони, подходящи за различни типове проекти като 2D, 3D, VR и AR;
- **Отваряне на съществуващи проекти** – В центъра се предоставя централизирано местоположение за преглед и отваряне на съществуващи проекти. Това е полезно за работа върху множество проекти едновременно или за превключване между различни проекти;
- **Организация на проекти** – Проектите могат да се организират и категоризират, с което се улеснява управлението и достъпа до тях;

- **Ресурси за обучение** – В центъра има директни връзки към уроци, документация и други ресурси за обучение. Това е изключително полезно за начинаещи разработчици, които искат да научат Unity.

Версията на Unity Hub, която е използвана за изграждането на прототипа, е „Unity Hub 3.8.0”, а версията на Unity, която ще се използва, е „Unity 2019.1.12f1”. От версията зависят актуализации и нови имплементации, които имат възможност да сринат проекта, ако се ползват критични функции.

Съответната версия на Unity Editor Трябва да поддържа Open JDK и Android SDK & NDK Tools.



Фигура 3.1: Главен прозорец на Unity Hub.

3.4.2. Инсталиране на Unity

Инсталацията на Unity включва изтегляне на инсталационен пакет от уебсайта на Unity и преминаване през на инструкциите на инсталатора. При конфигурирането на Unity се избират различни опции за инсталиране на различни компоненти и добавки включително поддръжка за различни

платформи като iOS, Android и виртуална реалност. След успешната инсталация потребителят може да започне да използва Unity.

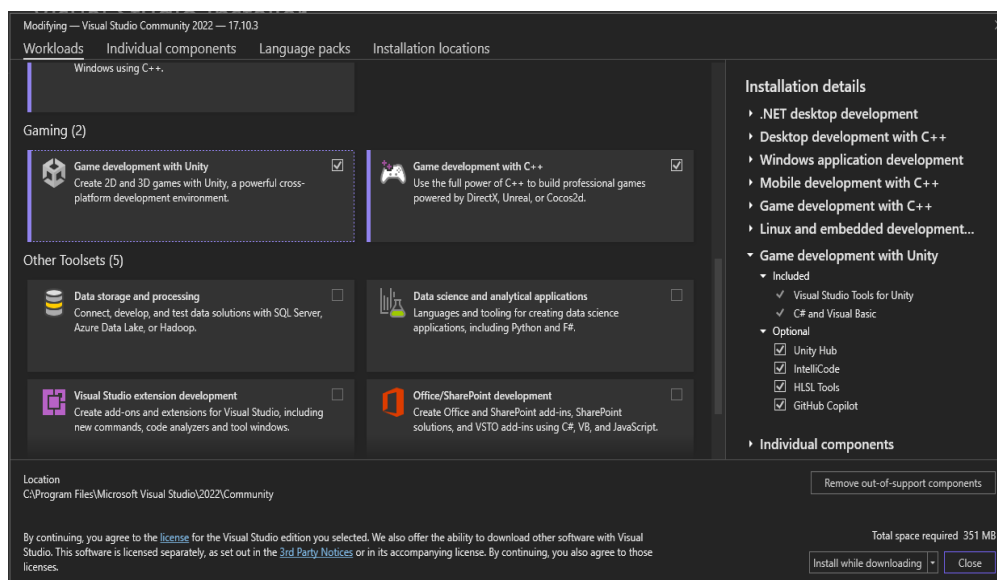
3.4.3. Преглед на Microsoft Visual Studio и пакета Game development with Unity

Visual Studio (Фигура 3.2) е интегрирана среда за разработка (IDE), широко използвана за писане, редактиране и отстраняване на грешки в програмен код в различни езици за програмиране [50]. Ще се разчита на Visual Studio и на интеграцията, която предлага за Unity. Visual Studio служи за улесняване на разработката на игри в Unity чрез предоставянето на следните възможности и функционалности:

- **Интегрирана среда за разработка (IDE)** – Visual Studio служи като комплексно IDE, където може да се пишат, редактират и управляват Unity скриптовете и кодовата база [41]. Има мощни функции като подсветка на синтаксиса, IntelliSense (предложения за довършване на код), инструменти за рефакторизиране, дебъгер и други.
- **Отстраняване на грешки в кода** – Отстраняването на грешки в кода е безпроблемно заради програмата дебъгер, вградена във VS. Може да се задават точки за прекъсване, да се проверят променливи и да се минава постъпково през изпълнението на кода. Това помага за идентифициране и отстраняване на неочаквано поведение в програмата.
- **Управление на проекти** – Visual Studio е оптимизиран за управление на големи проекти. Проектни файлове като *.sln и *.csproj могат да се менажитат точно както в Unity Hub.
- **Интеграция с Unity** – Visual Studio има интеграция със средата за разработка Unity. Автоматично се генерират необходимите файлове за

проекти, след което се създава Unity проект и се свързват директно с IDE-то.

- **Контрол на версиите на приложения (Version Control System, VCS)** – Visual Studio поддържа интеграция на VCS като Git, с което се улеснява управление на промени, сътрудничеството с членове на екипи и проследяването на историята на проекта директно от програмната среда.
- **Разширения и персонализация** – Visual Studio поддържа плъгини не само за работа с Unity. Това подобрява производителността, като се предоставят допълнителни инструменти, шаблони и функции, които са насочени към разработката на 3D приложения с Unity.
- **Обучителни материали и поддръжка** – Microsoft и Unity предлагат обширна документация, учебни материали и обществена поддръжка за използване на Visual Studio с Unity. Това включва ресурси за начинаещи и напреднали потребители.



Фигура 3.2: Инсталиране на пакет Game development with Unity
през Visual Studio Installer.

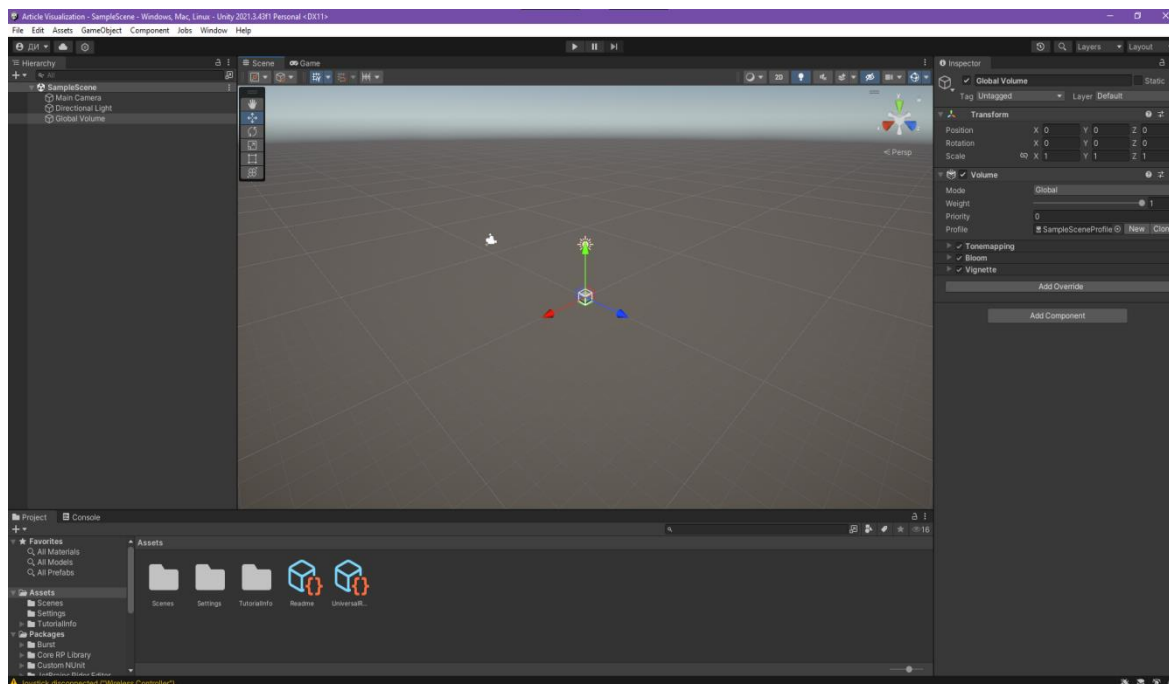
3.4.4. Преглед на C#

Използваният програмен език за разработката на проекти в Unity е C#. Този език е избран поради своята широка поддръжка и мощни възможности за разработка на игри и други приложения. Със C# разработчиците могат да създават сложни функционалности, да управляват графични елементи и да създават взаимодействия между различни компоненти на играта.

C# (Си-Шарп) е обектно-ориентиран програмен език, който е основен инструмент за разработване на софтуер с платформата Unity. Той предлага богат набор от възможности за програмиране включително управление на обекти, изграждане на логика за игри и създаване на потребителски интерфейси. Синтаксисът на C# е подобен на други популярни програмни езици като Java и C++ и предоставя по-високо ниво на абстракция и лесен за усвояване синтаксис.

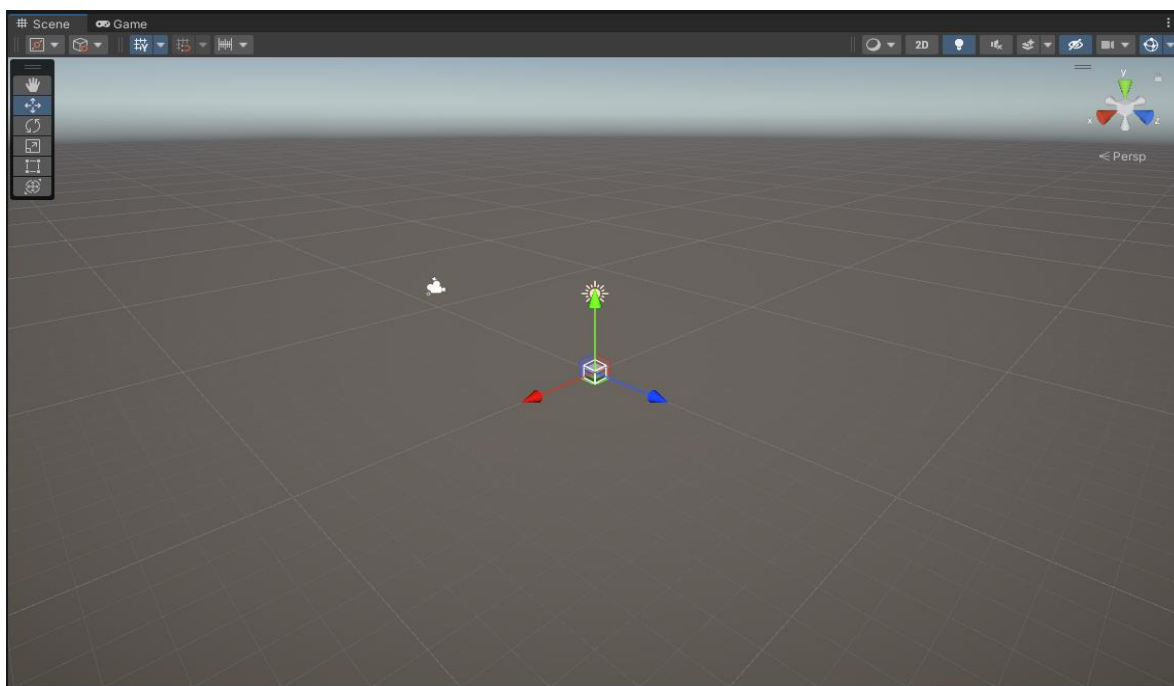
3.4.5. Преглед на елементите на потребителския интерфейс на Unity

На пръв поглед, след избиране на типа на проекта и създаването му, Unity отваря главния прозорец за разработка, които е показан на фигура 3.3 .



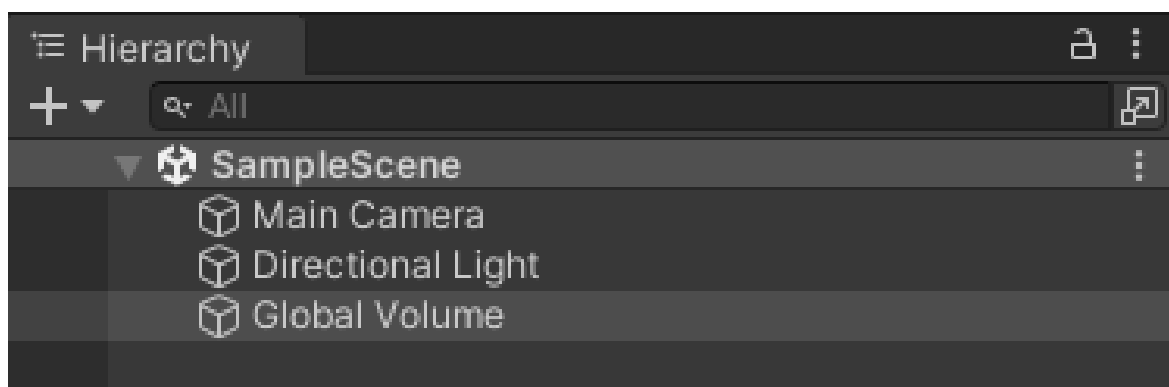
Фигура 3.3: Главен прозорец на Unity.

Прозорецът, показващ сцената (Scene) (Фигура 3.4), предоставя визуално представяне в реално време на игровия свят [42]. Работата с него предоставя лесно манипулиране на обекти, достъп до инструменти за редактиране и функции за отстраняване на грешки. Към прозореца Scene също се включва прозорецът Game, който предлага преглед на играта от погледа на играча. Разработчиците може да тестват симулацията директно в редактора на Unity, позволявайки бърза итерация и отстраняване на грешки. Той осигурява обратна връзка относно функциите на симулацията, визуалните ефекти и цялостната производителност.



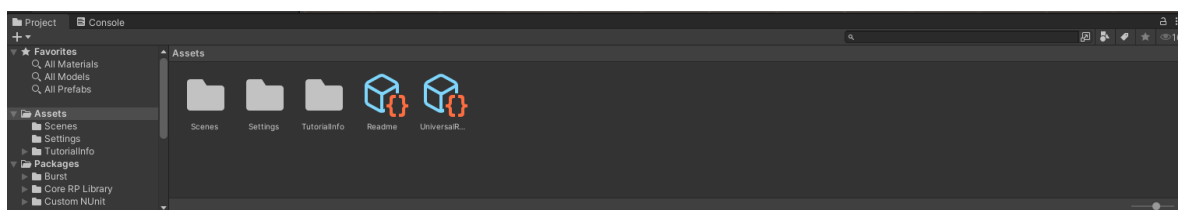
Фигура 3.4: Прозорец Scene на Unity за визуализиране на сцената.

Прозорецът за йерархии (Hierarchy) (Фигура 3.5) в Unity показва йерархичен изглед на всички обекти, присъстващи в текущата сцена. Тези обекти са организирани по структуриран начин, което позволява на разработчиците лесно да се управляват и манипулират. Обектите може да производни на други обекти, тоест да има връзка родител-дете, улеснявайки сложни трансформации и взаимодействия.



Фигура 3.5: Прозорец за преглед на йерархиите Hierarchy в Unity.

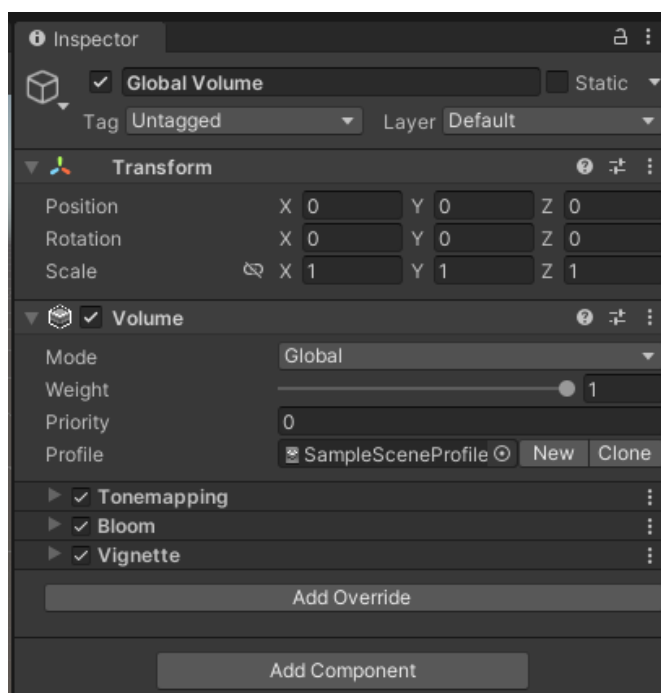
Прозорецът за файлове на проекта (Project) (Фигура 3.6) е място за управление на файлове на проекта като скриптове, текстури, модели и аудио файлове. Това улеснява вмъкването, организацията и манипулирането на файлове, като дава възможност на разработчиците да управляват ефективно ресурса на проекти. Отдясно стои файлово дърво, което може да се преглежда.



Фигура 3.6: Прозорец за преглед на файловете на проекта Project.

Прозорецът с контролите и опциите (Inspector) (Фигура 3.7) предоставя подробна информация и контроли за избрани игрови обекти или модели. С него може да се променят свойства като трансформация, материали, скриптове и други компоненти. Прозорецът на инспектора служи като мощен

инструмент за фина настройка на свойствата на елементите на играта, позволявайки прецизен контрол върху тяхното поведение и външен вид.



Фигура 3.7: Прозорец за фина настройка Inspector в Unity.

3.5. Програмата за моделиране Blender

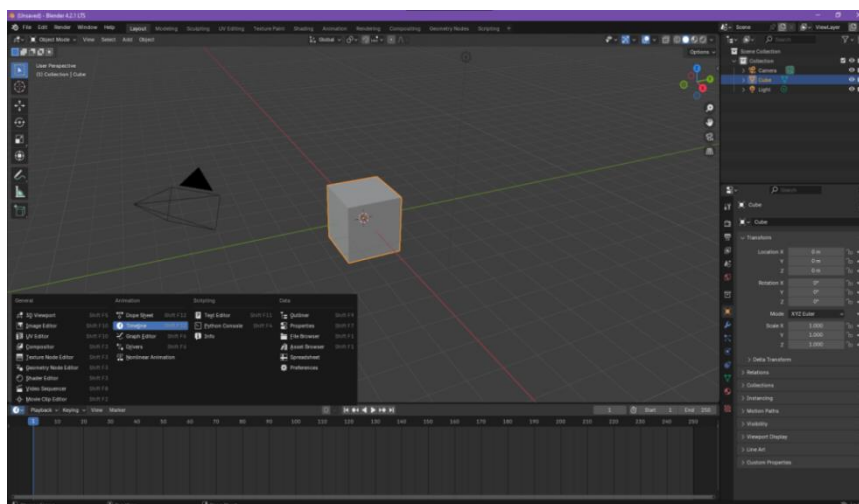
Blender [47] е безплатен софтуер за създаване на тримерни модели, анимации и визуализации. Той предлага широка гама от инструменти за моделиране, текстуриране, анимиране и рендериране и е подходящ за различни проекти от игри и филми до архитектурни визуализации и виртуални среди. Blender е популярен сред артисти и професионалисти поради своята гъвкавост.

3.5.1. Инсталиране на Blender

За инсталация и конфигурация на Blender потребителите изтеглят безплатно софтуера от уебсайта на Blender и го инсталират, следвайки стъпките на инсталатора. След успешна инсталация се конфигурират настройките според предпочитанията и софтуерът е готов за ползване.

3.5.2. Създаване на нов проект

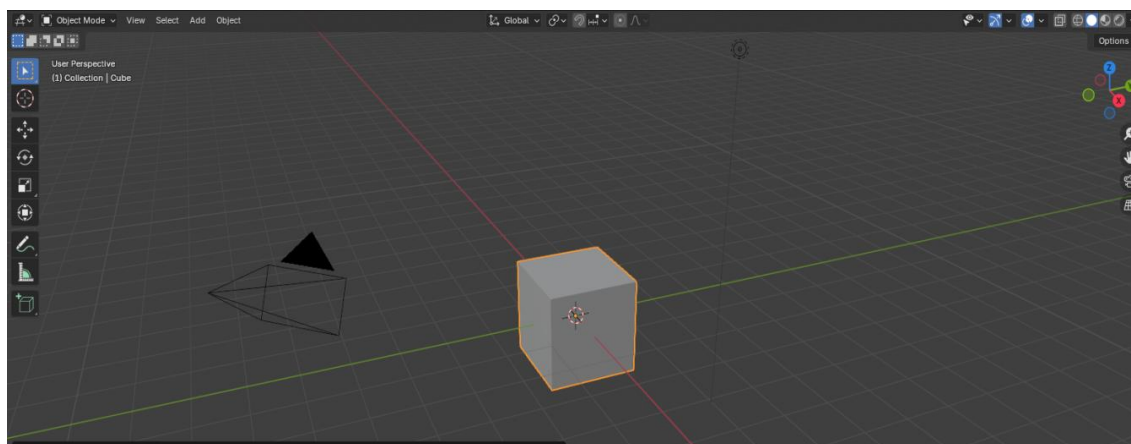
При създаването на нов проект в Blender се започва с избора на подходящ шаблон или празен проект. След като се избере начална отправна точка, може да започне добавянето на обекти, да се настройват материали, да се прилагат текстури и да се използват различни инструменти за моделиране и анимация, за да се гради проектът (Фигура 3.8).



Фигура 3.8: Главен прозорец на Unity.

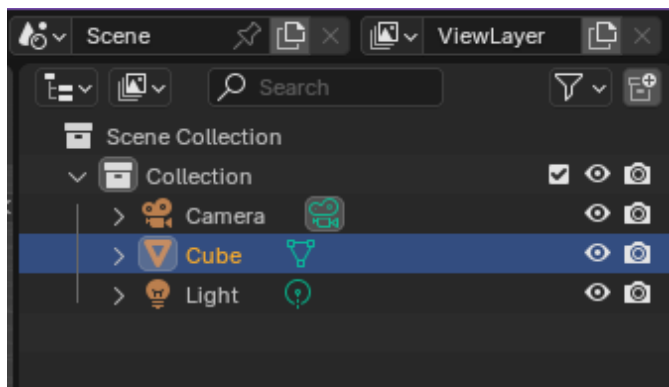
3.5.3. Основни елементи на Blender

В Blender сцените (Фигура 3.9) представляват виртуалната среда, в която се разработва визуалното съдържание. Всяка сцена се състои от обекти, които могат да бъдат моделирани и анимирани. Има множество елементи като камери, светлини, текстури и ефекти.



Фигура 3.9: Прозорец Scene на Blender за визуализиране на сцената.

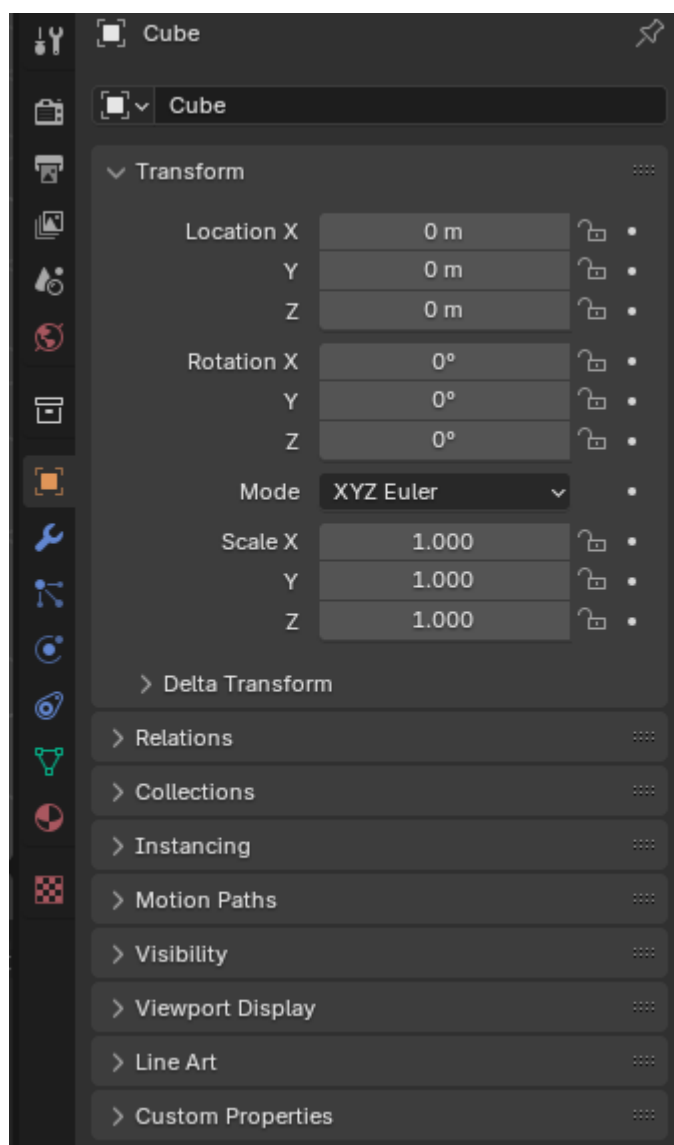
Добавянето и манипулирането на обекти в Blender се извършва чрез използване на различни инструменти за моделиране и трансформация. Потребителите могат да създават разнообразни форми и геометрични обекти като кубове, сфери, цилиндри и други. След създаването им, обектите може да се манипулират чрез промяна на техния размер, форма и позиция в сцената. Те могат също така да бъдат групирани, изрязвани, модифицирани и текстурирани (Фигура 3.10).



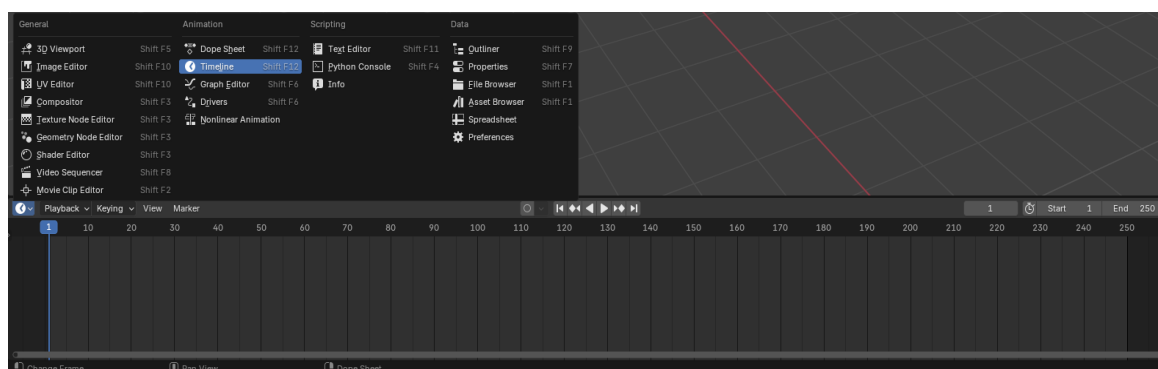
Фигура 3.10: Прозорец за преглед на йерархиите на обекти в сцената в Blender.

В Blender основните панели и инструменти са организирани в различни изгледи и редактори, които предоставят разнообразни възможности за извършване на операции. Някои от основните прегледи включват сцената, UV редактор за текстуриране, Node Editor за работа с материали и много други.

Инструментите включват примитиви, мрежи, модификатори (Фигура 3.11), а също и инструменти за рисуване, текстуриране и анимация (Фигура 3.12). Всеки преглед и инструмент предлагат широк набор от функционалности и настройки, които позволяват на потребителя да създаде сложни и креативни визуализации.



Фигура 3.11: Прозорец за фина настройка Inspector в Blender.



Фигура 3.12: Прозорец за допълнителни функции.

3.6. Структура на проекта

3.6.1. Създаване и настройка на проект

Създаването на нов проект в Unity е началната точка за разработка на интерактивно приложение. За да се създаде нов проект, е необходимо да се отвори Unity Hub, което е централно място за управление на проекти. След това се избира опцията "New" и се конфигурират основните параметри на проекта като име, път, място за съхранение на файловете и версия на Unity. След като се потвърдят настройките, Unity ще създаде нов проект и ще предостави празна сцена.

3.6.2. Проектиране на сцени

Основите на сцени и обекти в Unity са съществен елемент от процеса на разработка на игри и приложения. В Unity, сцената представлява визуална среда, където можете да се поставят и организират обекти като герои, околния свят, ефекти и други елементи на приложението. Обектите в Unity представляват елементи, които са включени в сцената, и които могат да бъдат визуализирани, манипулирани и програмирани.

При работа със сцени в Unity може да се добавят и преместват обекти, да се настройват техните свойства като размер, цвят и материал както и да се програмира поведение чрез скриптове. Също така може да се създават и управляват множество сцени в един проект, което позволява да се създават сложни и интерактивни среди.

Чрез интуитивния графичен потребителски интерфейс на Unity може лесно да се добавят различни видове обекти към сцената. Библиотеката на Unity включва голямо разнообразие от предварително създадени обекти като герои, околна среда, препятствия, ефекти и други.

След като се добавят обекти към сцената, може да се манипулират по различни начини – тяхната позиция, завъртане, големина, свойства и

компоненти. Може да се добавят скриптове, които да управляват техните поведения и да ги правят интерактивни. Този процес позволява да се създадат разнообразни и комплексни сцени, които да отговарят на нуждите и визията за проекта.

3.7. Подробности на прототипа

За да може да се визуализира прототипа на мобилен телефон, очевидно е, че трябва да има изпълним файл на устройството. Трябва в настройките за сглобяване на изпълнимо приложение (build settings) да се настрои компилиране за android устройства. В този етап се добавят и сцените за показване. В настройките на играча (player settings):

- Се включва поддръжката за виртуална реалност;
- Се указва, че се използва Cardboard SDK;
- Премахва се поддръжката за графичното API Vulkan;
- Променя се името на пакета;
- Променя се минималното целево ниво на Android API.

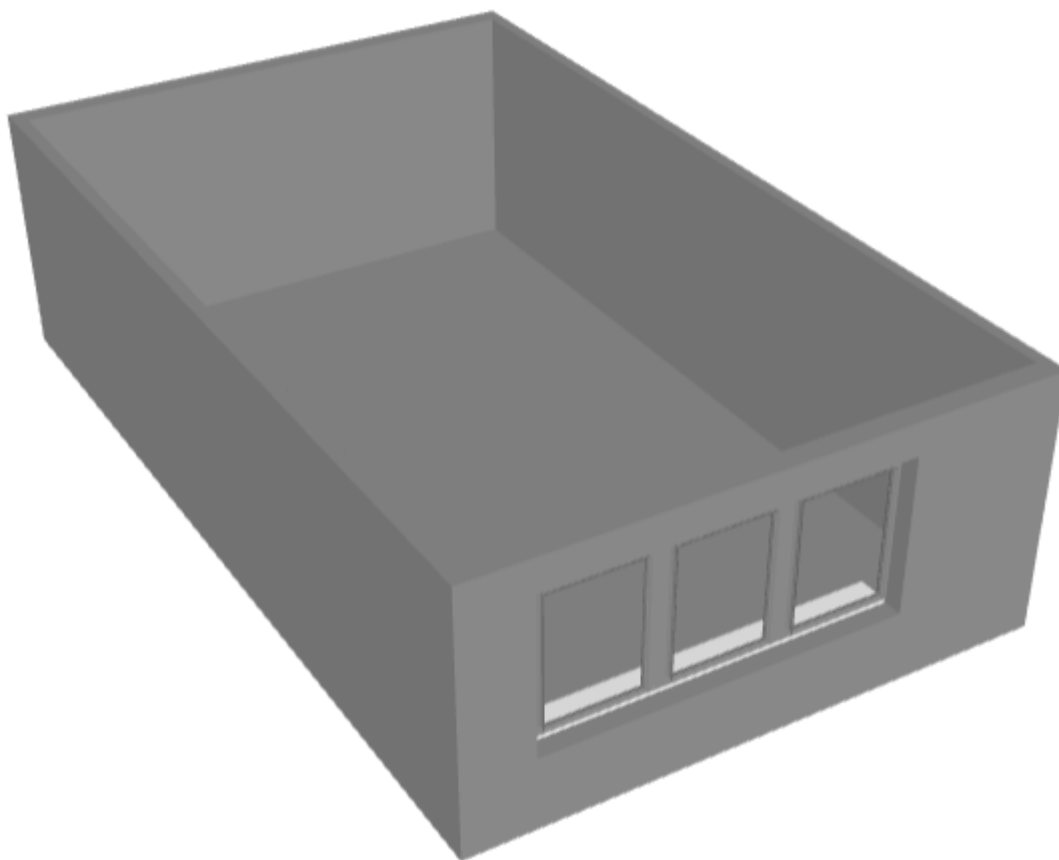
Мобилното устройство, за да може да се тества, трябва да бъде свързано с компютъра чрез кабел и на него да има инсталирано приложението Unity Remote 5.

В Unity Editor е необходимо да се инсталира пакета GVR SDK for Unity v1.200.1. От този пакет в йерархията се добавя обект от тип GvrEditorEmulator и GvrEventSystem.

3.7.1. Дизайн на стаята – под, стени, таван

Дизайна на стаята е сравнително прост (Фигура 3.13). Включва се под, таван, четири стени и един 3-секционен прозорец. Като самостоятелен обект представлява една цяла част. Подът има текстура на ламиниран паркет. Стената е бяла, със ситна украса за текстура, наподобяващо тапет. Таванът е

кремав на цвят, подобно на боя. Той се добавя последен. Текстуриите и материалите са взети наготово от Unity Asset Store.



Фигура 3.13: Геометрия на стаята без текстури.

3.7.2. Разполагане на предмети и мебели

Мебелите (Фигура 3.15) са разположени по такъв начин, за да се накара стаята да изглежда като всекидневна. Стаята съдържа четири отделни секции (Фигура 3.14), сглобени в една голяма част. На нея има принтер, монитор и две големи тонколони. На отсрещната страна са разположени син, разгъваем диван, скрин с поставено червено кубче отгоре. То служи за демонстрация на функционалността на бутона. До тях са разположени червен кожен диван, стъклена масичка и срещу тях – телевизор. Стаята, секциите, червеният диван, скринът, масичката и телевизорът са създадени с помощта на Blender и са интегрирани в Unity като готови обекти. Синият диван, тонколоните,

принтерът и мониторът са взети от Unity Asset Store. Тексурите също са взети оттам. Основната причина е трудоемката им направа.



Фигура 3.14: Четвъртият по вид скрин, текстуриран, осветен отгоре.



Фигура 3.15: Мебелите, както са показани в Unity Editor.

3.7.3. Създаване на контролер за потребителя

За да може потребителят да взаимодейства с околния свят, базовото нещо, което е необходимо, е вътре в симулацията да има камера. Създава се празен обект – родител на основната камера. Добавя се обект `GvrReticlePointer`, който ще изпълнява функцията на мерник. Пита се въпроса защо на потребителя му е необходим мерник? Google Cardboard предоставя функционалността от цял един бутон. Това дава скромни възможности на потребителя да взаимодейства с околната среда. Мерникът е оцветен в зелено за по-голямо удобство в полумрачни помещения. На обекта камера се добавя компонента `GvrPointerPhysicsRaycaster`. Това представлява основният метод, с който ще се намират функционални обекти.

Демонстративно, ще се покаже функция за смяна на цelta на едно кубче, намиращо се на сцената. За цelta на обекта кубче се създават 3 event trigger инстанции и се създава нов скриптов файл, който помещава функционалността (Фигура 3.16). Функциите се навързват с компонентите event trigger.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

Unity Script | 0 references
public class ChangeColor : MonoBehaviour
{
    0 references
    public void Red()
    {
        GetComponent<Renderer>().material.color = Color.red;
    }

    0 references
    public void Magenta()
    {
        GetComponent<Renderer>().material.color = Color.magenta;
    }

    0 references
    public void Green()
    {
        GetComponent<Renderer>().material.color = Color.green;
    }
}
```

Фигура 3.16: Функции за смяна на цвят.

3.7.4. Осветление

За осветление се ползва глобален източник на осветяване, подобен на слънцето. Той е насочен направо към прозореца в стаята от небето. По интензитет е малко по-силен от обикновено. Светлината е очаквано „топла“.

В кубчето е вграден точков източник на светлина, който при натискане на екрана върху обекта може да го включи или изключи (Фигура 3.17). По подразбиране е изключено. Кубчето има компонента скрипт LightSwitch и event trigger – Pointer Down.

```
public class LightSwitch : MonoBehaviour
{
    public Light my_light;
    0 references
    public void ToggleLight()
    {
        my_light.enabled = !my_light.enabled;
    }
}
```

Фигура 3.17: Функции за включване и изключване на светлинен източник.

3.7.5. Динамични текстури

Понеже Google VR SDK не предоставя допълнително контроли, с които да се придвижваме през множество опции в панел, за да се демонстрира, че е възможна смяната на текстури, ще накараме телевизора в сцената да покаже картинка на екрана си. За целта, точно както при лампата (кубчето с осветителния източник), създаваме UI компонент платно (canvas) с негов обект-дете RawImage и платното се позиционира пред дисплея. Скриптовият файл на телевизора SwitchTV действа по аналогичен начин на LightSwitch. В действителност това, което трябва да се случи, ако искаме да сменим текстурата на даден обект, е, че трябва да променим атрибута на свойството, което указва каква текстура се показва на екрана. Технически смяната на цвета на кубчето и показването на изображение се зачитат като смяна на текстура. Но няма как да има фин контрол за избор поради констатирания проблем в началото.

3.7.6. Аудио

Добавен е аудио източник към монитора, който може да бъде включен и изключен. За целта е създаден обект Audio Source – дете на монитора (Screen). Създадени са скрипт и event trigger и са свързани по същия начин както осветлението на кубчето. Аудиото е 3D и се губи в зависимост от ориентацията на главата и разстоянието на потребителя до източника.

3.7.7. Движение на потребителя

Това, че Google Cardboard поддържа 3 DOF не означава, че някакъв смислен механизъм за движение не може да се направи. Такава функция е дадена при системи за виртуална реалност, поддържащи външни контролери към тяхната екосистема, специално направени за тях, и са включени към пакетите за разработка. На фигура 3.18 е показано по какъв начин тази функционалност е реализирана.

```
public class Movement : MonoBehaviour
{
    public float speed = 3.5f;

    public float gravity = 10f;

    private CharacterController controller;

    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {
        controller = GetComponent<CharacterController>();
    }

    // Update is called once per frame
    Unity Message | 0 references
    void Update()
    {
        PlayerMovement();
    }

    1 reference
    void PlayerMovement()
    {
        float horizontal = Input.GetAxis("Horizontal");
        float vertical = Input.GetAxis("Vertical");

        Vector3 direction = new Vector3(horizontal, 0, vertical);
        Vector3 velocity = direction * speed;
        velocity = Camera.main.transform.TransformDirection(velocity);
        velocity.y -= gravity;
        controller.Move(velocity * Time.deltaTime);
    }
}
```

Фигура 3.18: Функционалност за движение на потребителя.

Първо е необходим реквизит. Необходимо е потребителят да разполага с джойстик. В йерархията с обекти трябва да се сложи празен обект, който ще служи като потребител. В този обект се вграждат камерата с мерника като деца. И последно се добавя компонента `CharacterController` и скриптов файл `Movement` да помещава кода.

Необходимо е да се зададе с каква скорост се придвижва потребителят в пространството на XZ равнината и Y линията. Гравитацията най-лесно се изчислява, като от координатата Y на потребителя извадим стойността на параметъра за гравитация. Тук се забелязва, че промяната е линейна, а не квадратна (ускоряваща вследствие на продължителното по време влияние на силата на гравитация). Такъв модел за точност не е необходим в момента, понеже не е необходимо да се симулират специфични физически взаимодействия. За придвижване в XY равнината се вземат първо две стойности – силата по линия на хоризонта и силата по линия на вертикала. Тук се ползват полярни координати. Създава се данна за насочен вектор в пространството, след което се прилага силата за движение. Накрая обновяваме стойностите на потребителя. Това се случва за всеки интервал от време, в който се засича постъпване на входни данни.

При разработката на контролера за движение е важно да се отбележи, че на мебелите, поставени в сцената, и частите на околната среда като стени, под и таван трябва да се зададе физичен обект за колизия, обикновено 3D куб, обхващащ техния модел, за да не може потребителят да изпадне от зададения игрови свят.

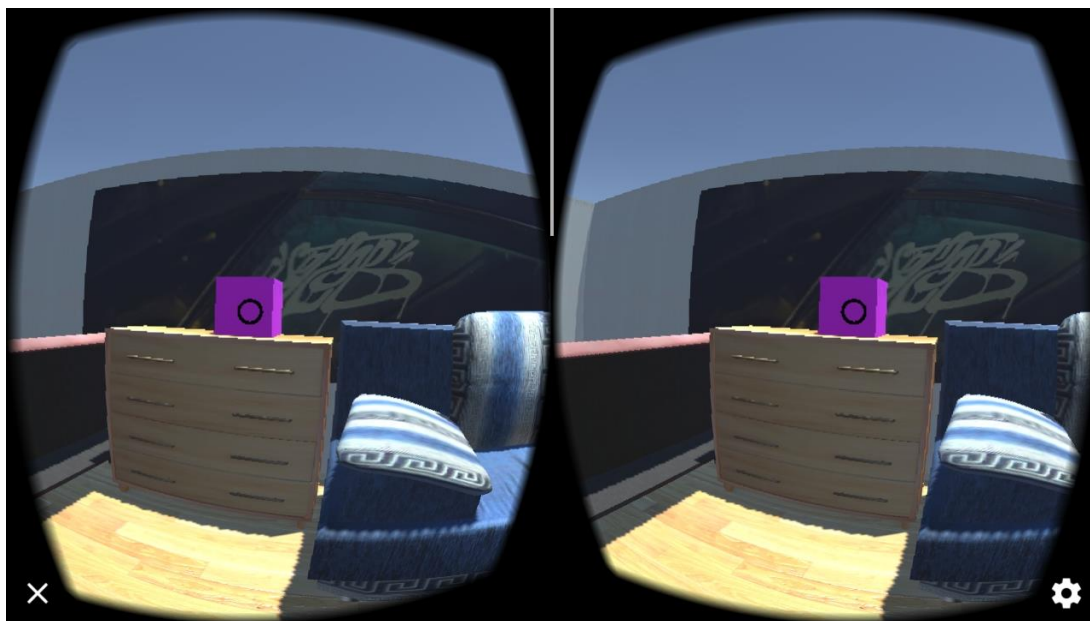
3.8. Тестване на приложението и получени резултати

Приложението използва стандартния графичен поток за 3D сцени, който Unity предоставя, и е компилирано за Android телефони не по-стари от Android 4.4 KitKat. Телефонът трябва да има сензор жирокоп.

Приложението е тествано с телефон Samsung Galaxy J5 (2017) модел SM-J530F със следните важни параметри [54]:

- **Размер** – 146.2 x 71.3 x 8 мм.;
- **Тип екран** – Super AMOLED;
- **Размер на екран** – 5.2 инч, 75 кв. см.;
- **Резолюция** – 720 : 1280;
- **Отношение** – 9 : 16;
- **Операционна система** – Android 9.0 (Pie) с One UI;
- **Чипсет** – Exynos 7870;
- **Процесор** – 8 ядра; 1.6 GHz;
- **Архитектура** – Cortex-A53, AArch32 & AArch64 ISA;
- **Видео чип** – Mali-T830, 1 клъстер, 700 MHz;
- **РАМ памет** – 2 GB;
- **Сензори** – Акселерометър, Жироскоп.

Приложението върви много гладко дори и опреснението на екрана да е 60 Hz. Производителността е задоволителна както и осветлението и функционалността на кубчето (Фигура 3.19).



Фигура 3.19: Изглед вътре в играта.

3.9. Затруднения по време на разработване на прототипа

Първоначално тази разработка е използвала Oculus SDK за създаване на цялата функционалност. Това, което не беше предвидено, е, че тази платформа не работи изобщо със мобилни телефони, защото тези устройства не са включени в съвместимите устройства от разработчиците. Изпитани са следните проблеми.

3.9.1. Несъвместимост с Oculus SDK

- **Несъвместими архитектури на системите на чип (SoC)** – Телефонът Samsung Galaxy J5 (2017) е 64-битов телефон, защото има разширение в своята архитектура с инструкции. Модулът Unity XR за Android, който се използва за разработката, поддържа само 64-битови устройства. Но Unity отказва да компилира/пакетира приложението за телефона с грешката, че устройството не се разпознава като съвместима 64-битова машина. Това навежда на мисълта, че в двата хардуера – на телефона и Oculus Quest 2 има базова и съществена разлика относно архитектурите и това как обработват машинния код.
- **Липса на обработка на данни от сензорите на телефона** – Камерата на потребителя по време на тестване не обновява своята позиция на завъртане или разположение в пространството. Това навежда на мисълта, че богатството на сензори в Oculus Quest 2 и неговите контролери е много необходимо за проследяване и без всичките налични сензори устройството ще откаже да работи, защото не може да изчисли точен резултат.

3.9.2. Несъвместимост с игрови контролер за PlayStation 4

- **Несъвместимост с контролер за PlayStation 4 с Oculus SDK** – Указва се, че Oculus SDK не може да работи с контролера за игровата конзола PlayStation 4, макар че контролерът има жироскоп и акселерометър за сензори. Не го показва в приложението. Това навежда на мисълта, че богатството на сензори в Oculus Quest 2 и неговите контролери е дори много необходимо за проследяване и без всичките хардуерни компоненти да знаят кое къде се намира в пространството няма да работят добре.
- **Несъвместимост с контролер за PlayStation 4 с Google VR SDK** – Указва се, ще функционалността за натискане на екрана чрез външен контролер и негов бутон не е зададена по подразбиране и при нейна липса не може да се използват интерактивните обекти. Също така Google VR SDK не предоставя възможности тази тъй функция да се задава системно.

3.9.3. Режим на работа на потребителя при реализирана с Google VR SDK функционалност

След създаването на контролера на потребителя се оказва, че тази функционалност и функционалността за натискане на екрана с бутон и насочване на мерника има конфликт. Двете функционалности не могат да работят заедно. Функционалността на движение присъства в игровия контролер за PlayStation 4, но не и зададен бутон за натискане на екрана. Безжичният Bluetooth контролер, който е комплект с VR очилата, може да реши проблема, но не и без да създава допълнителни проблеми.

Контролерът има 3 режима на работа, от които 2 от тях са интересните. Това са режим „Мишка“ и режим „Игра“. В режим „Мишка“, потребителят не може да се движи, но може да взаимодейства с обектите във виртуалната

среда. В режим „Игра“ потребителят може да се движи в околната среда, но функцията за натискане на екрана не работи и не може да се взаимодейства с интерактивни обекти. Смяната на режима изисква задържането на 2 бутона за продължителност от 5 секунди. Това създава неудобство в работата на потребителя, но е адекватно решение на проблема с конфликтната допълнителна функционалност.

Съществува допълнителен проблем. При смяната от режим „Игра“ към режим „Мишка“ и обратно джойстикът губи своята калибрация, което означава, че придвижването във виртуалната среда се затруднява, защото интуитивните посоки за движение не отговарят на реалността в приложението. Накратко ляво спира да е ляво, може да не е дясно и може самата зона, отговаряща за посоката, да е или увеличена, или намалена.

Това може да дезориентира потребителя при работа със системата. Препоръчва се първо да се направи виртуална разходка в средата в режим „Игра“, след което да се превключи в режим „Мишка“ и да се взаимодейства с интерактивните обекти в неподвижна позиция. Превключването на режима отново не е препоръчително.

3.9.4. Липса на функционалност за регулиране на височината

Google VR SDK не предоставя възможности за пряко регулиране на височината при промяна на положението на потребителя в реалния свят например при сядане в стол или клякане.

Също така приложението е на практика неизползваемо в легнало положение, защото няма регулация на ъгъла на ориентиране на камерата. Не става на въпрос за местене на главата, макар че това допълнително намалява движението, а изкуствено да се регулира позицията на камерата.

3.9.5. Липса на функционалност за управление на менюта

Google VR SDK не предоставя възможности за инстанциране на външни менюта. Това ограничава възможностите за създаване на меню, извиквано с бутон, защото такава функционалност не само липсва, но и се усложнява многократно логиката на взаимодействие между менютата и виртуалната реалност. Това за съжаление означава, че функционалност за динамично създаване и позициониране на обекти и динамично поставяне на текстури не може да се реализира по 3 главни причини:

- Липса на функционалност за създаване на менюта;
- Затруднения в анализа и логиката при установяване на контекста на предприето действие;
- Липсата на допълнителни функционални бутони.

Премислено е решение за поставяне на външен обект Canvas с поставен плъзгач на него за регулиране на височината някъде във виртуалното пространство на приложението. Това решение ще се окаже неестествено и проблемно, защото с регулиране на височината на потребителя той ще се придвижи надолу към пода или нагоре към тавана и този плъзгач ще излезе от неговото полезрение.

3.10. Бъдещи подобрения

С развитието на прототипа възникват няколко области за бъдещи подобрения, които ще подобрят функционалността и потребителското изживяване. Тези подобрения ще допринесат за създаване на по-стабилно приложение, което по-добре отговаря на нуждите на потребителите в областта на визуализацията на мебели и архитектура.

- **Разширена библиотека със съдържание** – Добре е да се разполага с по-обширна библиотека от 3D модели с разнообразна гама от стилове мебели и елементи на дизайна, което позволява на потребителите да създават по-персонализирани интериорни среди;

- **Подобрено взаимодействие с потребителите** – Добавянето на повече интерактивни функции например възможността за персонализиране на цветовете или материалите на мебелите в рамките на VR средата, ще ангажира допълнително потребителите. Това помага с индивидуалните предпочитания;
- **Навигация в няколко помещения** – Включването на среда с няколко стаи и преходи между пространствата би осигурило по-съгласувано пространствено изживяване чрез симулиране на цели домове или апартаменти;
- **Усъвършенстване на осветлението и добавяне на ефекти на околната среда** – В бъдещи актуализации може да се включи прилагане на динамично осветление и ефекти на околната среда в зависимост от времето на деня. Подобряването на реализма ще подобри разбирането как елементите изглеждат в различен контекст;
- **Включване на процедурно генерирани елементи;**
- **Усвояване на обратна връзка** – Чрез анализ на данните на цялостно потребителско изживяване се предоставя ценна информация за потребителските взаимодействия и областите, които се нуждаят от корекции;
- **Съвместимост с различни платформи** – С напредването на технологиите и разширяващата се съвместимост с други платформи с приложения за виртуална реалност се увеличава максимално достъпността и ще се разшири потребителската база.

ЗАКЛЮЧЕНИЕ

В този документ се описва подробно разработването на прототип на приложение за виртуална реалност (VR), насочено към подобряване на визуализацията на мебели и архитектура. Чрез цялостен анализ на съществуващите технологии за VR и 3D моделиране са установени пропуските в традиционните методи за визуализация и е предложено решение, използващо виртуална реалност, с което се преодолява това разминаване. Чрез интегриране на интерактивни елементи, привличащи вниманието, прототипът има за цел да предостави на потребителите правдоподобно усещане за пространство, мащаб и реална представа за дизайна, което често липсва при традиционните техники.

Основните постижения включват успешна концепция и проектиране на 3D среда, която представя интериорни пространства. Прототипът включва реалистични настройки на стаите с под, стени, таван и мебели, като се използват оптимизирани 3D модели, за да се осигури ефективна работа на мобилните платформи за виртуална реалност. Процесът на разработване се ръководи от поставени принципи за разработка, като се акцентира върху интуитивната навигация и безпроблемни взаимодействия, които са от решаващи за ангажирането на потребители, които може да не са запознати с технологиите за взаимодействие и възприятие на VR.

Изборът на технологии включително Unity като платформа за разработка и Blender за 3D моделиране улеснява създаването на многофункционално и динамично приложение. Използването на виртуална реалност, базирана на мобилен телефон, е насочено към достъпност по цена и разпространеност на устройствата. Освен това решенията, свързани с модулния дизайн и итеративната разработка, гарантират, че прототипът може да се развива и адаптира въз основа на смяна на хардуерната система.

Проектът не само осигурява функционален инструмент за визуализация на архитектурен и интериорен дизайн, но и показва потенциала за по-нататъшни иновации в сферата на VR приложенията. Идентифицирани са потенциални области за бъдещи подобрения като например навигиране в няколко помещения и сложни ефекти на околната среда, с което се илюстрира път за по-нататъшно развитие.

В заключение тази дипломна работа представлява стъпка напред в интегрирането на виртуалната реалност в процеса на проектиране. Чрез обединяването на най-съвременните технологии с творчески техники за визуализация прототипът има потенциал да промени начина, по който проектанти и клиенти са ангажирани с архитектурни проблеми.

Докато вървим напред с времето, е наложително хората да останат адаптивни и отворени към развиващия се пейзаж на VR технологиите, като нашите инструменти трябва да останат актуални и ефективни в общността на проектантите. Малките успехи на този прототип подчертават значителни ползи от интегрирането на VR технологии в места, където малко се използват.

ИЗПОЛЗВАНА ЛИТЕРАТУРА

1. Емилиян Петков. Основи на компютърната графика. Фабер. Велико Търново. 2013;
2. John F. Hughes. Computer graphics principles and practice. Addison-Wesley Professional. Providence, Rhode Island. 2013;
<https://www.amazon.com/Computer-Graphics-Principles-Practice-3rd/dp/0321399528>;
3. William R. Sherman. Understanding Virtual Reality. Morgan Kaufmann. California. 2018. <https://www.amazon.com/Understanding-Virtual-Reality-Interface-Application/dp/1558603530>;
4. <https://www.oculus.com/rift-s/>
5. <https://www.vrstore.bg/oculus-rift-s>
6. <https://www.vive.com/us/product/vive-pro2/overview/>
7. <https://www.vrstore.bg/htc-vive-pro>
8. <https://www.hp.com/au-en/shop/hp-reverb-vr-headset>
9. <https://www.vrstore.bg/hp-reverbe-g2-v2>
10. <https://www.meta.com/quest/products/quest-2/>
11. <https://www.vrstore.bg/oculus-quest2-64gb>
12. <https://www.apple.com/apple-vision-pro/>
13. <https://ardes.bg/product/apple-vision-pro-256gb-siv-mql83ll-a-393511>
14. <https://arvr.google.com/cardboard/>
15. <https://www.ubuy.bg/bg/brand/google-cardboard-vr>
16. <https://www.samsung.com/us/support/mobile/virtual-reality/gear-vr/gear-vr-with-controller/>
17. <https://www.vrstore.bg/samsung-vr>
18. <https://unity.com/>
19. <https://www.cryengine.com/>
20. <https://www.unrealengine.com/en-US/download>

21. <https://www.autodesk.com/products/autocad/overview>
22. <https://www.sketchup.com/en>
23. <https://www.solidworks.com/>
24. <https://www.autodesk.com/products/revit/overview>
25. <https://graphisoft.com/solutions/archicad>
26. <https://www.blender.org/>
27. <https://www.autodesk.com/products/3ds-max/overview>
28. <https://github.com/ValveSoftware/openvr>
29. <https://store.steampowered.com/steamvr>
30. <https://developer.oculus.com/>
31. <https://developers.google.com/vr/develop/unity/get-started-android>
32. <https://www.autodesk.com/products/fbx/overview>
33. <https://www.khronos.org/glTF>
34. <https://www.adobe.com/products/substance3d/discover/what-are-obj-files.html>
35. <https://www.khronos.org/collada/>
36. <https://www.gravitysketch.com/>
37. <https://www.adobe.com/products/medium.html>
38. <https://assetstore.unity.com/>
39. <https://pixabay.com/>
40. <https://unity.com/unity-hub>
41. <https://visualstudio.microsoft.com/>
42. Nicolas Alejandro Borromeo. Hands-On Unity 2022 Game Development. Packt Publishing. Birmingham. 2022. <https://www.amazon.com/Hands-Unity-2022-Game-Development/dp/1803236914>;
43. Jason Gregory. Game Engine Architecture, Third Edition. A K Peters/CRC Press. Florida. 2018. <https://www.amazon.com/Engine-Architecture-Third-Jason-Gregory-dp-1138035459/dp/1138035459>;

44. Andrew Rollings, Dave Morris. Game Architecture and Design: A New Edition. New Riders Pub. Indianapolis. 2003. <https://www.amazon.com/Game-Architecture-Design-Andrew-Rollings/dp/0735713634>;
45. Венета Кочовска, И. Гарванов. Проектиране на видео игра. УНИБИТ. София. 2016. https://fin.unibit.bg/images/pdf/bakalavri/Garvanov_56_knr.pdf;
46. Софтуерни платформи за разработка на видео игри. ФМИ Пловдив. 2019. <https://npict.fmi-plovdiv.org/wp-content/uploads/2019/10/Software-platforms-for-video-game-development.pdf>;
47. David Millet, Arthur Tombs. Blender 3D: Noob to Pro. https://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro;
48. Carrie Cousins, Ryan T. Riddle. Mobile UI Design Trends Present & Future: Mobile Card Interfaces. <https://www.uxpin.com/studio/ebooks/mobile-ui-design-trends-card-interfaces/>;
49. Jerry Cao, Marcin Treder. The Indispensable Designer: A Guide to Influential Design. <https://www.uxpin.com/studio/ebooks/the-indispensable-designer-a-guide-to-influential-design/>;
50. Terry Norton. Learning C# by Developing Games With Unity 3D... . Packt Publishing. Birmingham. 2013. <https://www.amazon.com/Learning-Developing-Games-Unity-Beginners/dp/1849696586>;
51. Jonathan Linowes. Unity 2020 Virtual Reality Projects. Packt Publishing. Birmingham. 2020. <https://www.amazon.com/Unity-2020-Virtual-Reality-Projects/dp/1839217332>;
52. Quickly preview your game on Android device | Unity tutorial. <https://www.youtube.com/watch?v=iCXwaehzRFQ> ю
53. How do I turn on the Developer Options menu on my Samsung Galaxy device?. <https://www.samsung.com/uk/support/mobile-devices/how-do-i-turn-on-the-developer-options-menu-on-my-samsung-galaxy-device/>
54. [https://www.gsmarena.com/samsung_galaxy_j5_\(2017\)-8705.php](https://www.gsmarena.com/samsung_galaxy_j5_(2017)-8705.php)

Източниците са достъпни в интернет считано от дата 02.07.2024 г.