

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc1132)

[АНАЛИТИЧЕСКАЯ ЧАСТЬ 7](#_Toc3864)

[1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 7](#_Toc3148)

[1.1 Анализ подразделения кафедра информационных систем организации МУИВ 7](#_Toc26810)

[1.1.1 Дерево бизнес-направлений организации 7](#_Toc5184)

[1.1.2 Сопоставление бизнес-процессов и критических факторов успеха организации 9](#_Toc22635)

[1.1.3 Анализ структуры и нормативной документации, регламентов подразделения «Базовая кафедра исследований информационных технологий на базе сервиса «ГитФлик»» под раздел кафедры информационных систем университета, регулирующих выполнение генерации игрового контента 11](#_Toc23584)

[1.2 Моделирование бизнес-процесса «генерация игрового контента» 12](#_Toc13076)

[1.2.1 Моделирование генерации игрового контента “КАК ЕСТЬ” 13](#_Toc20468)

[1.2.2 Моделирование процесса «как должно быть» 23](#_Toc31821)

[1.3 Анализ рынка программного обеспечения для обеспечения автоматизации бизнес-процесса генерации динамических виртуальных миров. 27](#_Toc30146)

[1.4 Анализ стейкхолдеров и их требований к разрабатываемой системе 28](#_Toc14229)

[1.5 Выбор средств разработки 29](#_Toc10864)

[1.6 Техническое задание на разработку «СДГВМ» 30](#_Toc8212)

[1.7 Выводы по разделу 30](#_Toc12030)

[2 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОЕКТА 31](#_Toc25457)

[2.1 Структурирование требований к разрабатываемой системе 31](#_Toc20594)

[2.2 План разработки ПО 33](#_Toc8343)

[2.3 Подбор лучших моделей для интеграции в СДГВМ для подсистем ИИ генерации игровых миров 35](#_Toc2077)

[2.3.1 Языковая модель Mistral-7B Instruct 35](#_Toc12632)

[2.3.2 Модели для генерации икон Awesome RPG Icon 2000. 36](#_Toc7983)

[2.4 Интеграция ComfyUI для реализации создания икон. 36](#_Toc15939)

[2.4.2 Обоснование выбора модели Awesome RPG Icon 2000 38](#_Toc30636)

[2.5 Процесс реализации системы динамической генерации виртуальных миров на платформе Unity 6000.0.30f1. 38](#_Toc22069)

[2.6 Разработка модели доступа к данным в системе динамической генерации виртуальных миров. 51](#_Toc28221)

[2.7 Выводы по второй главе 53](#_Toc20331)

[3 ТЕСТИРОВАНИЕ И ИНТЕГРАЦИЯ 54](#_Toc9087)

[3.1 Тестирование с оптимизацией производительности и отладкой СДГВМ 54](#_Toc9755)

[3.1.1 Тестирование методом “Черного ящика” 58](#_Toc12464)

[3.1.2 Тестирование методом “Оптимизация производительности” 60](#_Toc27872)

[3.1.3 Тестирование методом “Usability” 62](#_Toc10997)

[3.1.4 Тестирование методом “GUI/UI” 64](#_Toc17788)

[3.2 Составление плана инсталляции и развёртывания СДГВМ 65](#_Toc12676)

[3.3 План интеграции с существующими системами МУИВ. 67](#_Toc19650)

[3.4 Формирование модели обновлений и технической поддержки 69](#_Toc26236)

[3.5 Руководства администратора и пользователя 71](#_Toc31389)

[3.6 Выводы по третьей главе 72](#_Toc28475)

[4 ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ПРОЕКТА “СДГВМ” КАК СТАРТАПА 74](#_Toc16277)

[4.1 Обзор рынка генерации контента для создания квестов и позиционирование “СДГВМ” 74](#_Toc16472)

[4.2 Разработка бизнес модели проекта 83](#_Toc13126)

[4.3 Модель коммерциализации “СДГВМ” и ценообразование в Unity Asset Store 85](#_Toc23238)

[4.4 Прогноз объёма рынка подбор целевого сегмента под “СДГВМ” 89](#_Toc919)

[4.5 Расчёт затрат на привлечение целевой аудитории 94](#_Toc2721)

[4.6 Организационный план и структура затрат для “СДГВМ” 99](#_Toc4798)

[4.7 Расчёты инвестиций 104](#_Toc32169)

[4.8 Анализ рисков и чувствительности 108](#_Toc25169)

[4.9 Оценка эффективности внедрения в МУИВ 112](#_Toc30347)

[4.10 Выводы по экономической части 114](#_Toc7159)

[ЗАКЛЮЧЕНИЕ 115](#_Toc19291)

[СПИСОК ЛИТЕРАТУРЫ 116](#_Toc19565)

# **ВВЕДЕНИЕ**

На данный момент в современной организации, независимо от направления её деятельности, цифровая трансформация и внедрение систем автоматизации значительно упрощает технологические процессы и оптимизирует решение внутренних и внешних процессов, не всегда сотрудники могут решать сложные технические задачи создания игровых прототипов, следовательно разработка информационной системы для автоматизации создания динамических игровых миров стала актуальной темой исследования.

Для удобства использования было принято решение разработать информационную систему на основе платформы Unity 2D, с функциями: генерации квестов, поведения NPC через ONNX модели при помощи LLM-Unity, спрайтов героев и иконок предметов связанных с квестами для визуализации игрового прототипа. Используя современные технологии ИИ и автоматизации можно повысить конкурентоспособность проектов клиента, и повысить прибыль с продаж созданной продукции при помощи разработанного в работе инструмента для генерации контента-наполнения игрового мира.

Актуальность работы: отсутствие доступных инструментов для автоматизации процессов создания игровых прототипов, актуальная проблема инди-студий и соло разработчиков составляющих большую часть российского рынка геймдева. Растет спрос на автоматизированные решения в геймдев-индустрии, особенно в сегменте генерации контента на российском рынке, где ручные процессы снижают рентабельность и замедляют разработку.

Объектом исследования выпускной квалификационной работы является процесс разработки и монетизации инструмента для генерации динамических игровых миров, основная задача которого заключается в создании масштабируемого продукта с интеграцией критических факторов успеха.

Предметом исследования выпускной квалификационной работы является проектирование и оптимизация бизнес-процессов для автоматизации разработки инструмента генерации динамических игровых миров, патентирование и лицензирование, постоянная поддержка проекта в целях коммерциализации готового решения. Оптимизация «To Be» основывается на внедрении автоматизации в ведение существующих бизнес-процессов разработки инструмента генерации динамических миров,

Целью выпускной квалификационной работы является создание доступного инструмента, который упростит процесс создания игр для продажи сообществу инди-разработчиков. Решение не стремиться обгонять существующие решения и крупные платформы, цель – предоставить доступный, практичный инструмент для упрощения монетизации и масштабирования проектов небольших команд и соло-разработчиков, занимающих большую часть на рынке.

Основные задачи необходимые для достижения цели:

1. Разработать техническое задание с глубоким анализом межнейросетевого взаимодействия, структуры моделей и описания процессов преобразования биграмм и n грамм являющихся параметрами в файле формата JSON в эмбеддинги, описать оптимизацию переводом модели в GGUF формат, показать связывание функций системы генерации с автоматическим инструментом создания графического интерфейса для пользователя.

2. Провести исследование с анализом и поиском существующих инструментов генерации виртуальных игровых миров, найти лучший вариант моделей нейросетей, для выполнения генерации квестов, визуалов и диалогов, разобраны производительность и возможность интеграции с платформой

3. Найти подходящую метрику качества «BLEU» сгенерированного контента, применить для анализа проблем качества ассетов сгенерированных «СДГВМ», для соответствия ожиданиям пользователей и ГОСТ.

4. Соединение выбранных моделей при помощи ONNX и LLM-Unity используя файлы формата JSON для взаимодействия и синхронизации логики, в единую архитектуру автоматизированного комплекса по генерации игровых прототипов «СДГВМ».

5. Провести тестирование инструмента генерации «СДГВМ» и при помощи выбранных метрик качества «BLEU» в соответствии с ГОСТ 19.302-79 добиться улучшения качества результатов генерации.

6. Оценка экономической важности внедрения «СДГВМ» в МУИВ.

7. Для оценки интуитивности и понятности интерфейса, нужно провести тестирование при помощи тестеров, на основе полученного фидбека от взаимодействия пользователей с созданным интерфейсом, улучшить понятность и интуитивность интерфейса.

Моя бакалаврская работа состоит из:

Введения

Первой главы – теоретическая часть анализа существующих инструментов генерации и описанием преобразований JSON в эмбеддинги

Второй главы – практическая часть с описанием реализации функционала «СДГВМ», проведения тестов для определения соответствия метрикам качества генерации

Третьей главы – части интеграции, проверка работоспособности, тестирование и улучшение проекта “СДГВМ”

Четвертой главы - экономическое обоснование проекта”СДГВМ” как стартапа

Заключения - выводы

Списка литературы

Первая глава, посвящена глубокому анализу основных терминов связанных с преобразованием JSON в числовые вектора выбранными моделями нейросетей, для межнейросетевого общения ONNX в Unity и анализу разрабатываемого проекта «СДГВМ».

Вторая глава, посвящена интеграции инструментов в единую архитектуру на платформе Unity, разобраны возможности создания инструментов для реализации автоматического создания пользовательских интерфейсов в сцене.

В третьей главе, анализируется рентабельность и преимущества при внедрении системы динамической генерации виртуальных миров в МУИВ.

Четвёртая глава, посвящена бизнес планам, описанию гибридной бизнес модели, пути коммерциализации, описание затрат, расчёт необходимых инвестиций, описание целевой аудитории для проекта, анализу возможных рисков и способы предотвращения возможных экономических проблем стартапа.

Ссылка на git-репозиторий с исходным программным кодом: https://github.com/Slavik993/Diplom\_Shebanov\_2026.git

# **АНАЛИТИЧЕСКАЯ ЧАСТЬ**

# **1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

## 1.1 Анализ подразделения кафедра информационных систем организации МУИВ

### 1.1.1 Дерево бизнес-направлений организации

Система бизнес процессов инструмента генерации контента включает планирование, анализ конкурентов, монетизацию, создание дорожной карты, формирование бизнес-модели для конвертации функций инструмента в денежную прибыль при помощи привлечения новых лидов и продвижения проекта маркетингом, эволюция в инструмент для генерации проекта происходит через обновления и разработку, патентование и лицензирование для продажи и защиты авторского права, исправление проблем инструмента проводя поддержку клиентов, масштабирование для расширения количества продаж и долгосрочного роста финансовых активов, повышение устойчивости компании благодаря автоматизации бухгалтерского учета, и оптимизация процессов разработки при помощи администрирования.

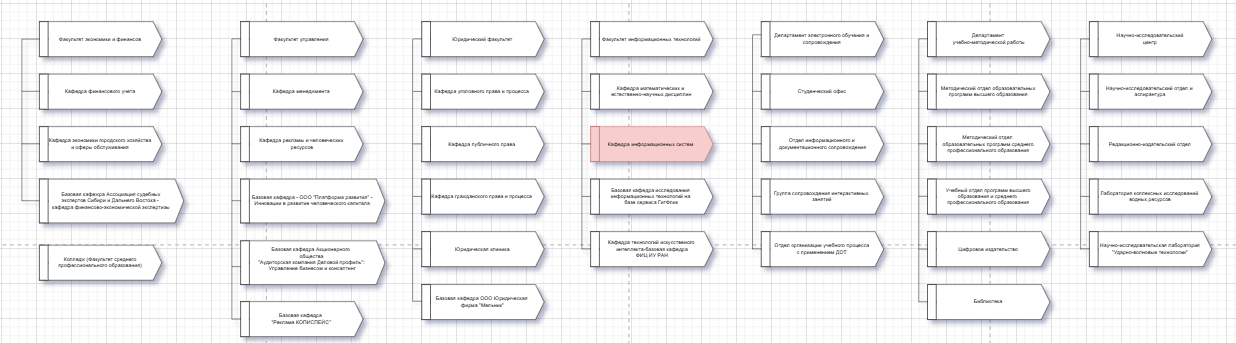


Рисунок 1.1 Обзор бизнес направлений организаций

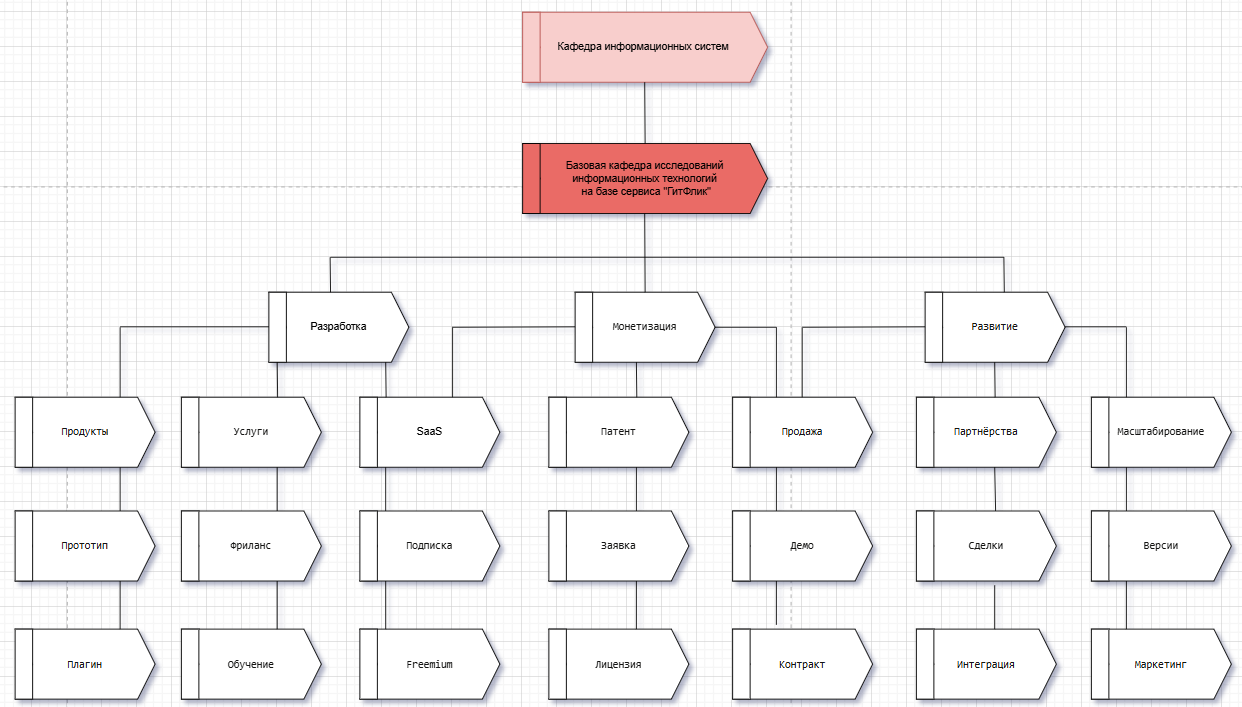


Рисунок 1.1.1 декомпозиция выбранного бизнес направления, для внедрения разрабатываемой системы генерации контента

Ключевые факторы успеха компании должны включать: адаптивность NPC к изменениям в игровом мире или действий игрока, таким образом реализуется защита пользователя от статичных квестов, тем самым повышая реиграбельность проекта. Конверсия лидов необходима для увеличения скорости монетизации и повышения эффективности процессов продаж. Автоматизация вставки контента происходит за счет интеграции инструментов из Unity Packet Manager, ввод параметров для генерации 2D ассетов, квестов и поведения NPC происходит через интерфейс. Массовость продаж гарантируется за счет низкого порога входа и ориентации на обширную целевую аудиторию в растущем сегменте рынка геймдева, проект не ориентирован обогнать существующие решения, задача реализации дать доступный инструмент для инди-студий и соло разработчиков. Необходимо повышать доверие клиентов при помощи увеличения качества генерации получаемого с постоянных обновлений и техподдержки, требуется повысить качество логических связей между переходами и событиями в квестах. Должна проводиться тестирование для корректировки генерируемого контента. Добавление кроссплатформенности увеличивает масштабируемость проекта и помогает интегрировать разработанный инструмент генерации на новые рынки, также расширение локализации увеличит целевую аудиторию, для увеличения продаж.

### 1.1.2 Сопоставление бизнес-процессов и критических факторов успеха организации

Таблица 1.1 Матрица сопоставления бизнес процессов и критических факторов успеха.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Бизнес-процесс | KFS1: Адаптивность NPC | KFS2: Конверсия лидов | KFS3: Интеграция инструментов | KFS4: Доступность для инди | KFS5: Качество генерации | KFS6: Масштабируемость | Степень важности количество КФУ |
| BP1: Стратегическое планирование и анализ | + | + | + | + | + | - | 5 |
| BP2: Формирование бизнес-модели | + | - | + | + | - | + | 4 |
| BP3: Маркетинг и привлечение лидов | + | + | + | + | - | + | 5 |
| BP4: Продажи и конверсия | + | + | + | + | + | - | 5 |
| BP5: Разработка и обновления | + | - | + | + | + | + | 5 |
| BP6: Патентование и лицензирование | + | + | - | + | + | + | 6 |
| BP7: Техподдержка и обучение | - | + | - | + | - | + | 3 |
| BP8: Аналитика и отчеты | + | + | - | + | - | + | 4 |
| BP9: Партнерства и масштабирование | - | + | - | - | + | + | 3 |
| BP10: Администрирование | + | - | - | + | + | + | 4 |

Выделяю патентование и лицензирование «СДГВМ» главным бизнес процессом благодаря влиянию на все критические факторы успеха проекта, гарантирует качество и конверсию идеи в доход.

Стратегическое планирование помогает обеспечить интеграцию процессов, адаптивность к запросам рынка, определяет направление развития и организует рациональное использование ресурсов.

Увеличивать клиентскую базу и адаптировать продукт под запросы потребителей помогает проводить конверсию лидов, увеличивать доступность для целевой аудитории потребителей включающей инди-студии и соло разработчиков.

Доходы компании генерирует бизнес процесс: продажи и конверсия, фокусируется на увеличении лидов, и масштабированию продаж на рынках.

Формирование бизнес-модели не влияет на качественные показатели реализуемого программного продукта фокусируется на стратегии делая процесс планирующим для поиска и анализа рисков и помощи в выборе действий увеличивающих прибыль от введения проекта.

Разработка и обновления реализует постоянную доработку для увеличения качества генерируемых квестов и поведения NPC, бизнес процесс повышает конкурентоспособность на рынке.

Принимать решения и корректировать стратегию развития проекта помогает бизнес процесс: аналитика и отчеты, фокусируется на адаптации к возможным угрозам и снижении рисков, путём укрепления выявленных аспектов в управлении проектом.

Партнерства и масштабирование, фокусируется на увеличении прямого дохода, не затрагивает адаптивность под запросы на рынке, используется для интеграции с новыми модулями Unity из Packet Manager Unity.

Соответствие ГОСТ стандартам обеспечивает бизнес процесс администрирование, даёт необходимую оптимизацию в управлении проектом, добавляет использование бухгалтерии 1С для автоматизации технических процессов денежных потоков и корректировку данных в отчетности для снижения рисков получения ошибок в отчетности.



Рисунок 1.1.2 – Матрица ранжирования бизнес-процессов.

### 1.1.3 Анализ структуры и нормативной документации, регламентов подразделения «Базовая кафедра исследований информационных технологий на базе сервиса «ГитФлик»» под раздел кафедры информационных систем университета, регулирующих выполнение генерации игрового контента

Кафедра информационных систем основана для обеспечения подготовки специалистов в областях автоматизации при помощи программного обеспечения 1С и прикладной информатики, включая искусственный интеллект и анализ данных, перечень направлений подготовки кафедры: бизнес-информатика 38.03.05 и прикладная информатика 09.03.03. Для обеспечения реализаций функционала подготовки студентов в кафедре информационных систем выделены роли: заведующий кафедрой МиЕНД, доцент кафедры, руководитель образовательной программы, старший преподаватель, преподаватель, лаборант кафедры. Регулировка процессов и работ на кафедре информационных систем происходит под действием следующих документов: методические указания МУИВ № 789 от 20.08.2024 «регламентом выполнения и защиты выпускной квалификационной работы», положением о кафедре информационных систем «приказ №456 от 01.02.23», положением о научной деятельности кафедры «приказом №234 от 10.05.23», положением о факультете информационных технологий МУИВ «приказ №123 от 15.09.2020». Основываясь на приказе №123 от 15.09.2020 необходимо обеспечивать «Базовую кафедру исследований информационных технологий на базе сервиса «ГитФлик»» необходимо реализовывать проекты для автоматизации с разбором бизнес процессов и дорожной картой, кафедра исследований предоставляет научное сопровождение для «СДГВМ».

Этими задачами занимается отдел «Базовая кафедра исследований информационных технологий на базе сервиса «ГитФлик»» под раздел кафедры информационных систем.

В задачи которого входят:

1. Распределение обязанностей в разрабатываемом проекте.
2. Техническая консультация разработки проекта, тестирование функционала, применение метрик качества для улучшения качества генерации контента.
3. Помощь в составлении отчетов по ВКР, внесение замечаний, для корректировки в соответствии с эталоном оформления и содержания ВКР.
4. Обеспечение безопасности данных, выполняет анализ возможных рисков, для минимизации ущерба.
5. Исполняет научное сопровождение в ходе исследовательской работы.

Генерация игрового контента «СДГВМ» в виде квестов, диалогов, иконок регулируется регламентами подразделения по научной и практической деятельности.

## 1.2 Моделирование бизнес-процесса «генерация игрового контента»

Бизнес процесс начинается с генерации JSON файла содержащим биграммы и n-граммы с тегами являющимися классами, далее происходит преобразование тегов в цифровые вектора – эмбеддинги, для пошаговой генерации текста в квестах и диалогах используются слои важности определяющие слово, для вставки на следующем шаге генерации, далее диалоги с неигровыми персонажами, квесты и иконки сохраняются в проект.

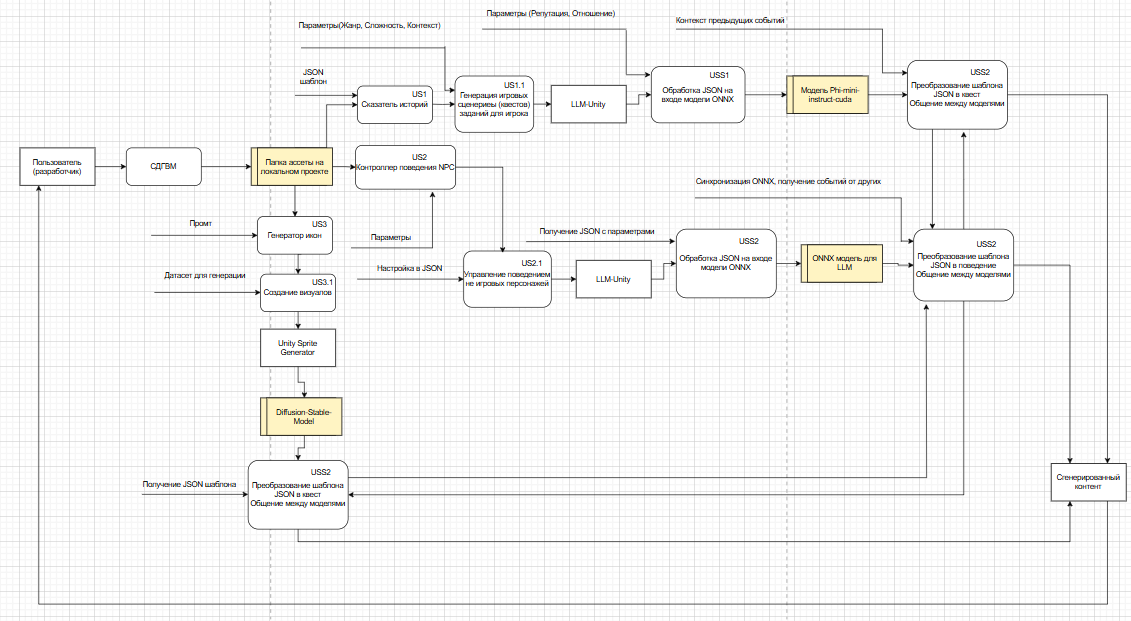


Рисунок 1.2. Моделирование бизнес-процесса на основе ассоциативного представления программных модулей

### 1.2.1 Моделирование генерации игрового контента “КАК ЕСТЬ”

На вход моделям нейронных сетей в системе динамической генерации виртуальных миров подаются JSON, являющиеся n-граммами с параметрами «классами», для начала работы «СДГВМ» нужно запустить сервер генерации визуалов ComfyUI, через скрипт main.py в проекте «можно открыть в Visual Studio Code», для оптимизации генерации, используются специальные настройки для LLMCharacter скрипта на объекте LLMCharacter, инструментами служат: шаблоны типа квеста, для определения уровня сложности, установленные настройки генерации, для достижения логической связанности между диалогами неигровых персонажей и написанием историй связанных между друг-другом, при длительных сессиях с несколькими итерациями генерации за раз. Механизмами «СДГВМ» являются LLM – Unity для реализации диалогов, путём использования ONNX и установленных в проект моделей, ComfyUI нужен для интеграции генерации визуальных элементов при помощи удобного интерфейса настройки моделей sd-turbo, sd3-medium, ComfyUI позволяет создавать цепочки генерации, при недостатке ресурсов локальной машины, а также генерировать синхронно несколькими моделями для увеличения скорости производства. INK помогает вручную добавить диалоговую конструкцию, без генерации. Разработчик – оперирует с конструктором квестов «СДГВМ» и следует инструкциям, для эффективной и качественной генерации. На выходе системы динамической генерации игрового мира получаются квесты, диалоговые деревья с обученными моделями поведения не игровых персонажей, визуалы в виде аватарок, иконок и спрайтов для окружения.

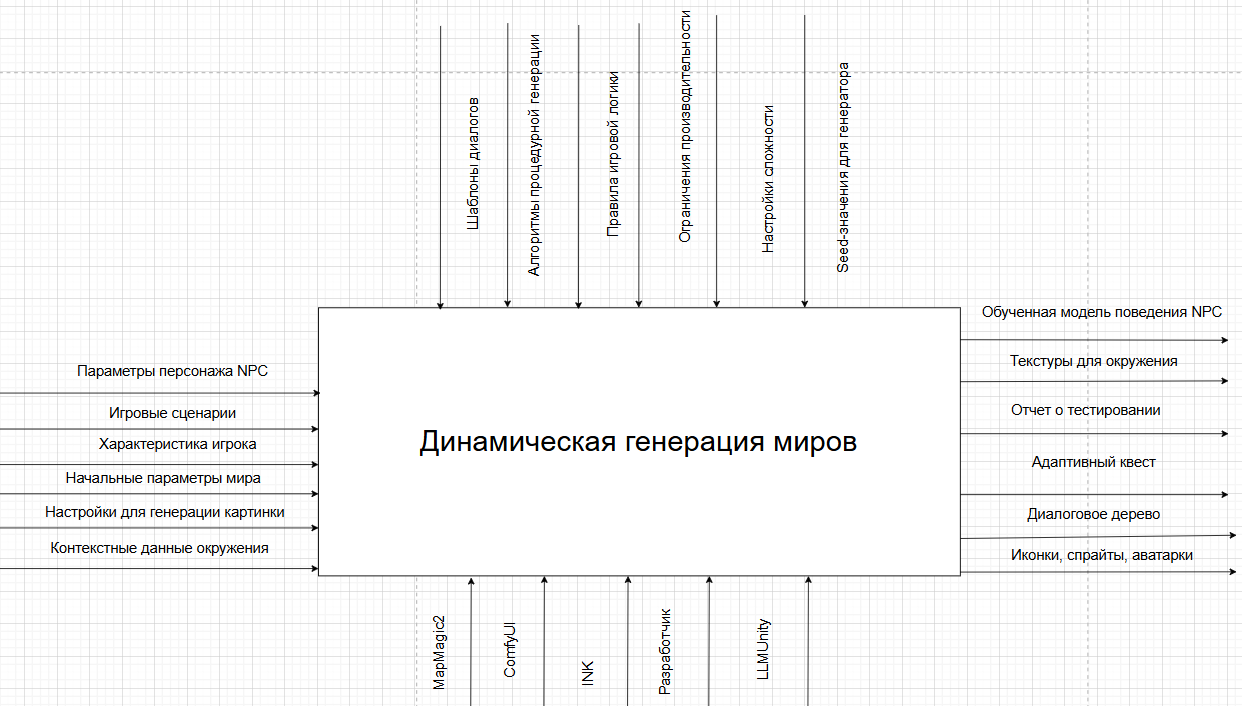


Рисунок 1.2.1 – IDEF0

При нажатии кнопки «сгенерировать» на основе введенных значений генерируется JSON имеющий параметры в виде классов, далее используются скрипты LLM, для запуска сервера генерации языковой модели или main.py с Comfy UI Client.cs для использования диффузионных моделей, далее модели трансформируют JSON в эмбеддинги, Mistral-7V-Instruct генерирует квесты и диалоговые деревья, sd-turbo и sd3-medium генерируют иконки, спрайты, аватарки, далее на основе тестирования получившегося игрового прототипа формулируются отчеты о качестве логических связей в диалогах, оценка соответствия сгенерированных квестов национальной культуре России, после тестирования и апробации элементов соответствия национальной культуре России, можно использовать сгенерированные квесты и диалоговые деревья, для образования студентов в области Российской национальности.

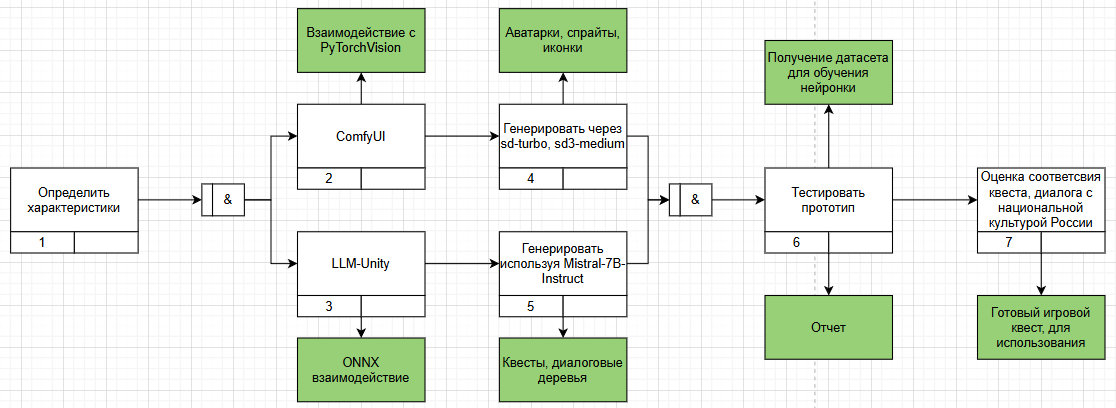


Рисунок 1.2.1.2 – IDEF3

Взаимодействие игрока с неигровыми персонажами в виде логов, используется для генерации поведения не игровых персонажей, разработчик создаёт 2D-визуалы, для сцены с квестом вписывая необходимые параметры в поля ввода, благодаря моделям sd-turbo и sd3-medium происходит генерация спрайтов окружения, аватарок и иконок, а также генерируется текст квеста при помощи Mistral-7B-Instruct, на основе квеста при помощи LLM-Unity создаются диалоговые деревья, после генерации игровой прототип тестируется на соответствие критериям наличия национальной культуры России.

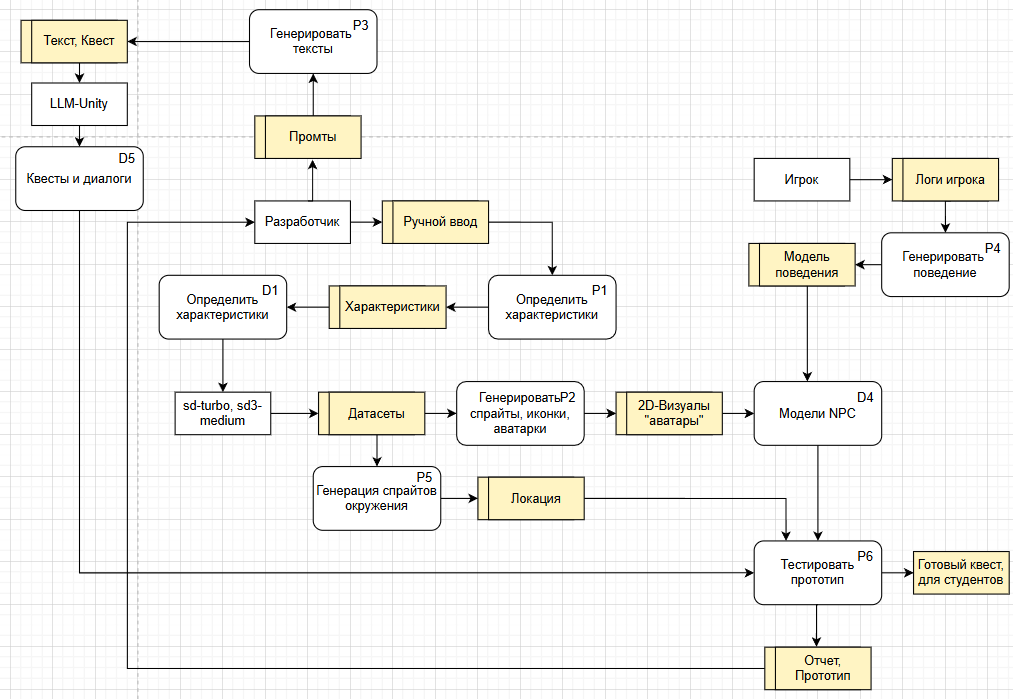


Рисунок 1.2.1.3 – Нотация Гейна-Сарсона

Студент «Игрок» даёт разработчику запрос на желаемый виртуальный мир, происходит генерация виртуального мира, шаблоны сохраняются, для дальнейшего обучения Mistral-7B-Instruct, на датасете собранного из качественных сгенерированных квестов про национальную культуру России, далее происходит персонализация контента под запросы студентов, что увеличивает привлекательность квестов системы динамической генерации виртуальных миров, для студентов, далее сгенерированные квесты дают студентам, для прохождения и повышения уровня знаний в области национальной культуры России, потом проходит обработка фидбека, от студентов прошедших квесты и получаются данные аналитики, для улучшения «СДГВМ».



Рисунок 1.2.1.4 – Нотация Йордона-Де Марко (DFD)

При помощи EPC диаграммы расписаны процессы «СДГВМ»: в блоке F1 определение характеристик разработчик ручным вводом вводит данные в поля ввода и выбирает выпадающие пункты dropdown, для генерации JSON определяющий характеристики поведения не игрового персонажа, далее синхронно происходит F3: Генерация портрета-автара персонажа, иконки, спрайтов при помощи Mistral-7B-Instruct, на выходе получается тексты и диалоговые деревья, в блоке «F2: Генерировать модель и портрет» использует sd-turbo, sd3-medium, «D2: Артефакты NPC», «D3: Квесты и диалоги» используются в совокупности при помощи ONNX для «F4: Генерировать поведение NPC «динамическая реакция»» используя ответы игрока вводящиеся вручную, используются ML-Agents из LLM-Unity, создаёт обученную модель поведения NPC, следующим этапом работы «СДГВМ» – «F5: Создание квеста для студента», при помощи добавления в сцены нужных объектов в иерархию вручную или скриптом в Unity Editor Mode, получаем созданный квест, для студентов, финальным этапом проводим тестирование квеста F6, в режиме Unity Play Mode, в финале получаем «готовый квест, для студентов».

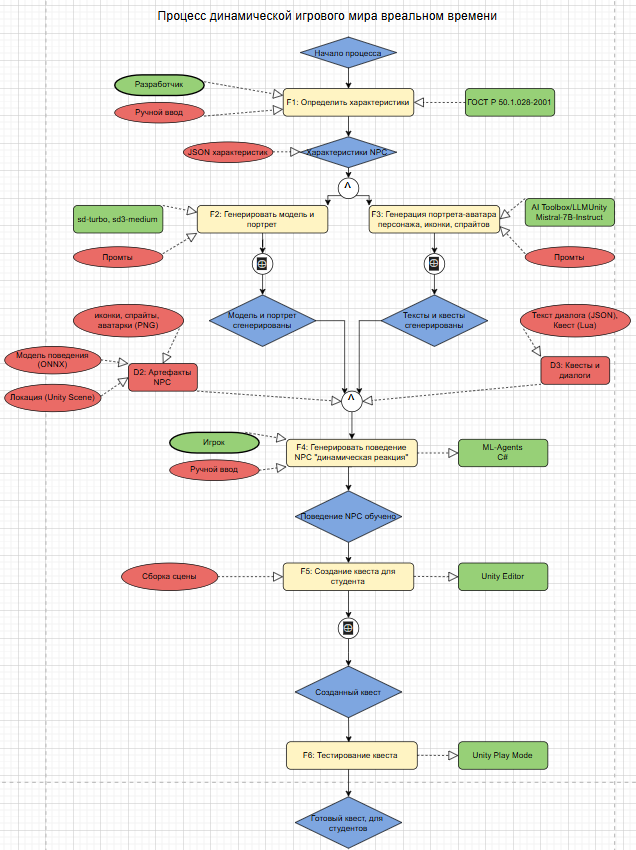


Рисунок 1.2.1.5 – Метод цепочки процессов, управляемой событиями (EPC)

В нотации BPMN, три основных этапа развития бизнес процессов: инициация включающая BP1: стратегическое планирование за счет SWOT-анализа и BP2: разработку бизнес модели благодаря любому графическому редактору, в проекте используется «Draw.io», этап выбора направления: BP3: Маркетинг и лиды, BP4: Продажи, BP5: Разработка «СДГВМ» на платформе Unity 6000.30f1, BP6: Подача свидетельства на программу, этап поддержки включает бизнес процессы: BP7: Техподдержка – решение проблем с инструментом генерации для клиентов, BP8: Аналитика, BP9: Партнерства, BP10: Администрирование – для получения финансовых отчетов. Созданная система бизнес-процессов выдаёт устойчивый доход в 250 000 рублей.



Рисунок 1.2.1.6 – Нотация BPMN

Процесс запуска системы динамической генерации виртуальных миров сопровождается предворительной инициализацией с вводом необходимой для генерации JSON информации, далее модель модели трансформера Mistral-7B-Instruct, получает параметры JSON, далее происходит проверка доступности ресурсов на используемой для запуска локальной машине, при нехватке ресурсов будет проводится обработка ошибок или прерывание процесса при помощи Unity Crash Log, при достаточном количестве ресурсов вычислительной машины, происходит валидация данных и анализ JSON с классами в виде n-грамм механизмом определения важности Mistral-7B-Instruct тридцати двумя слоями трансформеров, далее происходит преобразование n-грамм в эмбеддинги, затем для повышения качества происходит оптимизация весов моделей в формате GGUF, safetensors, meta, в системе динамической генерации виртуальных миров возможна синхронная генрация контента используя три модуля одновременно, но затрачивает большое колчиество ресурсов симстемы, для слабых высчислительных машин реализован асинхронный метод генерации одним модулем, при старте генерации происходит проверка таймаута, необходимо протестировать качество контента на соответсвие метрикам качества BLEU, и METEOR, далее результаты сохраняются нажатием на кнопку, разработчику использующему инструмент генерации квестов “СДГВМ” необходимо, исправить вручную недочеты в историческом и национальном плане несоответсвий, но при этом инструмент “СДГВМ” выполняет большую часть работы по созданию квеста, для студента, нужно дать протестировать созданный квест студентом, для получения обратной связи и внесения корректировок в информационное и смысловое наполнение квеста. При возникновении ошибки в Unity с обработкой в консоли, можно провести восстановление и вернуться к работе. При нехватке ресурсов компьютера может случиться сбой при котором невозможно выполнить восстановление.

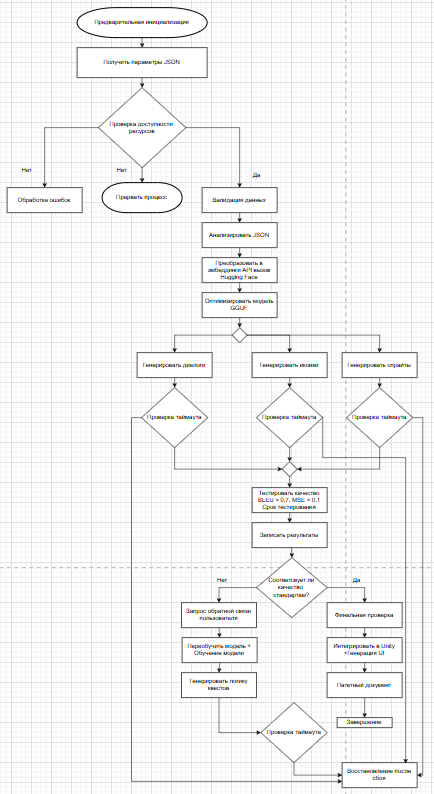


Рисунок 1.2.1.7 – Нотация UML

Матрица распределения ответственностей включает необходимые для реализации системы динамической генерации виртуальных миров роли, включая опытных тестировщиков для получения обратной связи по поводу удобства работы с интерфейсом «СДГВМ», технического консультанта для правильной постановки задач по реализации функционала генерации контента и помощи с решением возникающих технических ошибок в разрабатываемой системе генерации контента, также для полноценной реализации функционала требуется помощь специализированных программистов способных реализовать требуемые функции для генератора контента, нужны специалисты по платформе Unity, для упрощения реализации запланированных подсистем «СДГВМ» основываясь на знании принципов работы LLM-Unity и ONNX межнейросетевого взаимодействия для организации синхронной генерации виртуального игрового пространства.

Таблица 1.2 Матрица распределения ответственности

|  |  |  |
| --- | --- | --- |
| Задача | Роль | ФИО участника |
| Проверка технической документации | Научный руководитель «СДГВМ» | Блощук Андрей Алексеевич |
| Объединение общей архитектуры проекта, разработка функционала | Технический консультант | Максим Владимирович Преображенский |
| Тестирование интерфейса на понятность | Тестировщик интерфейса | Мудров Никита Андреевич |
| Помощь в разработке скриптов функционала | Программист | Исаев Николай Александрович, Шебанов Вячеслав Викторович |
| Тестирование функционала и соотнесение с ГОСТ 19.302-79 | Знающий стандарты платформы Unity специалист | Преображенский Максим Владимирович, Блощук Андрей Алексеевич |
| Обзор ошибок в архитектуре текстового содержания диплома и раздача замечаний по улучшений | Научный руководитель «СДГВМ» | Блощук Андрей Алексеевич |
| Оптимизация технической составляющей | Технический консультант | Максим Владимирович Преображенский |
| Подготовка к защите ВКР | Куратор проекта «СДГВМ» | Блощук Андрей Алексеевич |

### 1.2.2 Моделирование процесса «как должно быть»

Эволюция системы динамической генерации виртуальных миров начинается с глубоко анализа рынка геймдева, поиска не занятой актуальной ниши в области создания игрового контента, на основе полученных данных о росте спроса на генерацию игрового контента и создание квестов при помощи искуственного интелекта, стратегическое планирование позволяет создать Бизнес-модель для реализации запланированного функционала, разработка html сайта для привлечения новых лидов, сам процесс разработки был бы не актуален, без предарительного анализа рынка и анализа возможных рисков. Во время разработки получал обратную связь от научного руководителя и технического консультанта, для улучшения качества реализуемой “СДГВМ”, поддержка системы происходит в период реализации системы динамической генерации виртуальных миров вплоть до защиты, патентование происходит подачей заявки третьего ноября на авторское свидетельство на программу в ФИПС Российской Федерации, патентование необходимо для монетизации патента и получении прибыли, продажи возможны благодаря реализованому ресурсу html сайта презентующего кратко функционал системы динамической генерации игровых миров. Важно провести аналитику качества при помощи метрик “BLEU”, “METEOR”, “FID”, “Inception Score”, “CLIP Score” повысить качество генерируемого виртуального игрового пространства и наполняющих мир диалоговых деревьев и квестов, необходимо повышать качество генерируемых спрайтов, квестов, диалоговых деревьев, для повышения вероятности партнерства и монетизации патента частным и юридическим лицам, что приводит к получению выручки от разработки проекта.

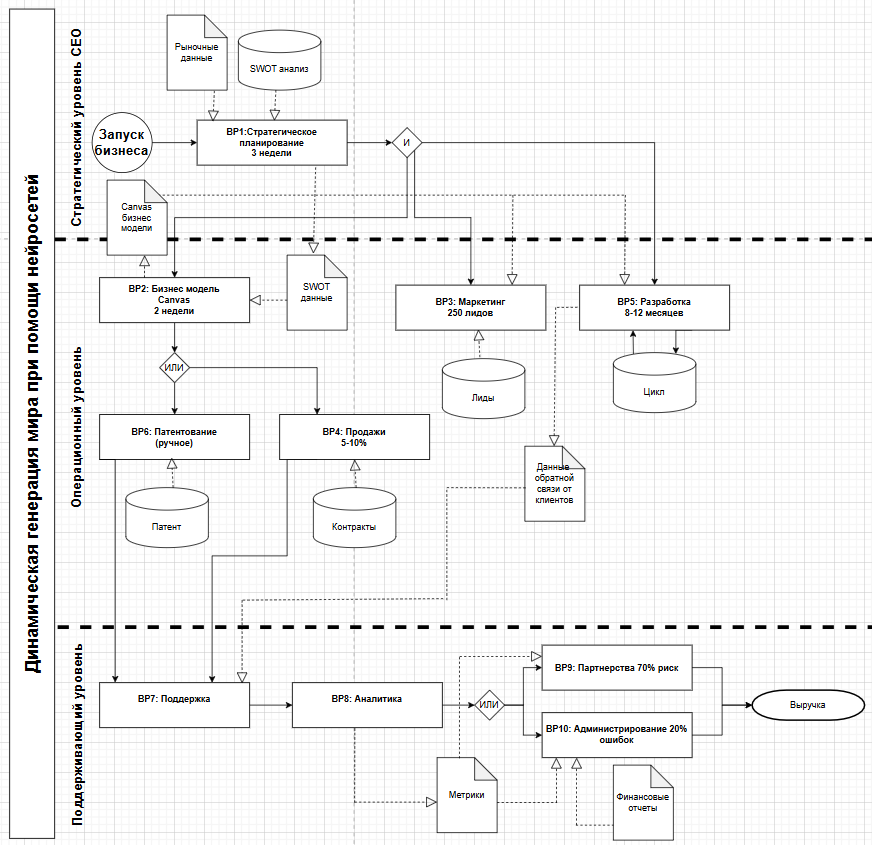


Рисунок 1.2.2 Диаграмма AS IS «Как есть»

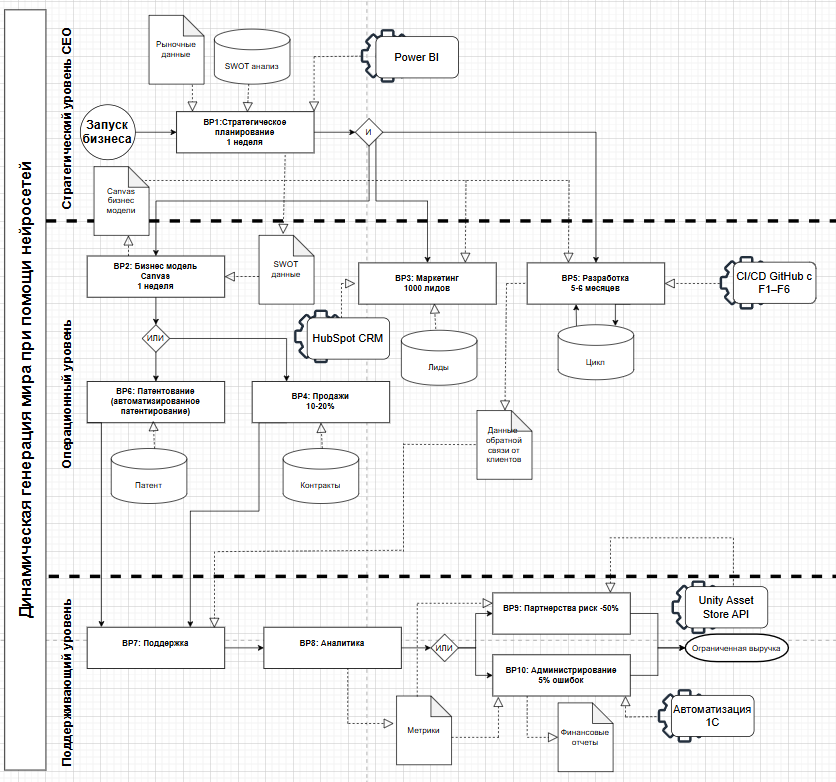
Оптимизация модели AS IS включила в себя использование сервиса GitHub с начала создания проекта «СДГВМ», она позволила автоматизировать процесс разработки и выдавать временные карточки для реализации работ создания функционала «СДГВМ» в сроки. Использование инструментов мощной бизнес аналитики Power BI – 1С:Битрикс 24, помогло автоматизировать поиск возможных рисков в бизнес процессах приводящих к снижению прибыли, использование умной платформы на базе искусственного интеллекта HubSpot совместно с пассивной рекламой в виде html сайта с демонстрацией функционала системы динамической генерации виртуальных миров используя SEO продвижение, страницы приводит к увеличению потенциальных лидов, повышающих прибыль от реализации проекта «СДГВМ» Московскому Университету имени Витте. Для снижения вероятности рисков и отказов в партнерской программе МУИВ, необходимо реализовать интеграцию с API Unity Asset Store, выставить инструмент динамической генерации виртуальных миров на Unity Asset Store, и монетизировать части функционала «СДГВМ». Для успешной реализации бухгалтерского учета нужно, интегрировать 1С для автоматизации создания необходимых отчетов и регистров накопления.   


Рисунок 1.2.2.1 Диаграмма IS AS «Как есть»

Таблица 1.3 Шкала и критерии оценки БП и степени проблемности.

|  |  |  |
| --- | --- | --- |
| Оценка процесса | Критерии оценки | Значение степени проблемности |
| Отлично | Высокая оценка студентами и преподавателями национальных аспектов квестов и диалоговых деревьев, спрайтов, иконок, аватарок, есть запоминание системой предыдущих диалогов | 1 |
| Хорошо | Увеличено качество генерации при помощи метрик: BLEU, METEOR, FID, Exception Score, CLIP Score. |  |
| Удовлетворительно | Достигнута стабильная генерация |  |
| Не очень хорошо | Не используется межнейросетевое общение при помощи ONNX |  |
| Плохо | Конструктор квестов без машинного обучения |  |

**1.3 Анализ рынка программного обеспечения для обеспечения автоматизации бизнес-процесса генерации динамических виртуальных миров.** Проведя глубокий анализ рынка направления «СДГВМ», был обнаружен рост ежегодно около двадцати семи процентов, процесс генерации квестов по национальной культуре России востребован в США и Австралии в вузах и школах, рассмотрел аналоги систем генерации квестов при помощи нейросетей используемых в образовании, существующее аналоги, имеют отрицательную связь с национальной культурой России, для аналогов требуется длительная адаптация или не подходят под необходимый формат, MyQuest LMS и TeachQuest не доступны в России, но подходят под требования к конструктору квестов, при помощи нейронных сетей. Многие квестовые генераторы дополнительно требуют VPN, «не полная блокировка» в России: Google AI Quests, Duolingo for Schools, Quatltrics.

Таблица 1.4 Конкуренты «СДГВМ»

|  |  |  |
| --- | --- | --- |
| Название системы | Производитель | Стоимость годовой лицензии |
| myQuest LMS | myQuest “Австралия” | 500 - 2000 долларов |
| TeachQuest | TeachQuest “США” | 1200 - 3000 долларов |
| Google AI Quests | Google “США” | Бесплатно «Google Classroom» |
| Duolingo for Schools | Duolingo “США” | 1000-5000 долларов для университетов |
| Qualtrics XM | Qualtrics “США” | 5000-15000 долларов |

## 1.4 Анализ стейкхолдеров и их требований к разрабатываемой системе

Финансирование, контроль качества и выполнения требований к «СДГВМ» выполняет администрация МУИВ, администрация отвечает за реализацию вычислительных машин с восьмью гигабайтами оперативной памяти, для запуска «СДГВМ» в МУИВ. Преподаватели будут стейкхолдерами, оценивающими связанность квестов и диалоговых деревьев, с национальной культурой России, выставляющими требования к визуальному виду интерфейсу. Команда ВКР как стартап, выставляют условие, для реализации автоматизации работы конструктора квестов при помощи нейросетей, реализуют мониторинг результатов работы «СДГВМ» и документации. Эксперты по культуре, проверяют точность культурных аспектов, дают данные, для дообучения на чистых данных по национальной культуре России. Студенты являются пользователями квестов, генерируемых системой динамической генерации виртуальных миров, дают обратную связь по исправленным преподавателями квестам, указывают, на недостатки созданных квестов, необходимые элементы для улучшения.

## 1.5 Выбор средств разработки

Конструктор квестов должен быть автоматизирующей бизнесс процесс системой, для увеличения дохода МУИВ, необходимо приминить машинное обучение и геймфикацию путём создания виртуального игрового пространства повышающего интерес студентов к образованию.Существующее на предприятие программное обеспечение, позволяет разработать и интегрировать в информационную среду МУИВ проект “СДГВМ”, для генерации образовательного контента. Использовать Google Classroom не эффективно, требуется ручная генерация контента, нет возможности интегрировать национальную культуру России, анализ показал низкую вовлеченность студентов в Google Classroom по данным других университетов Toronto «Канада», North Carolina State University «США», University of Notre Dame «США», причины и недостатки Google Classroom, были учтены, при разработке «СДГВМ», была добавлена игрофикация, и ситуации для увеличения интереса к особенностям национальной культуры России, при использовании Google Classroom есть необходимость тратится на новое программное обеспечение по типу VPN. “СДГВМ” требует уже существующее программное обеспечение на компьютерах: платформу Unity 6000.0.30f1 и Python. Язык C# используется, для работы c нейронными сетями и реализации функционала в Unity, Python использован для запуска локального сервера генерации визуалов реализованного с помощью ComfyUI. Среда разработки Unity подходит под деятельность базовой кафедры GitFlic и используется для обучения студентов на направлении геймдев, рационально будет использовать эту среду, по причине активного использования среды университетом на компьютерах, среда уже показала оптимизированную работу на установленных компьютерах в МУИВ.   
 Для работы с нейронными сетями и межнейросетевым общением в Unity использованы фреймворки: ML-Agents «MML-Unity» – UnDreamAI, Sentis Unity - ONNX, INK, ComfyUI – рабочее пространства для создания производственных цепочек для генерации иконок, спрайтов, аватарок.

## 1.6 Техническое задание на разработку «СДГВМ»

Техническое задание на разрабатываемую систему динамической генерации виртуальных миров представлено в Приложении 1.

## 1.7 Выводы по разделу

Создание актуальной идеи для монетизации невозможно, без изучения перспективных направлений на рынке, анализ рисков необходим на протяжении разработки проекта «СДГВМ», для монетезации важно получить авторское свидетельство на программу, создать сайт и заниматься SEO продвижение при помощи интеллектуальной платформы HubSpot CRM, для повышения эффективности разработки нужно использовать GitHub, назначая актуальные задачи при помощи временных карточек «issues», проанализированы аналоги конструкторов квестов, большинство аналогов имеют ограничения на использование в Российской Федерации, и возникают сложности с интеграцией настоящей национальной культуры России, также были учтены проблемы из университетов «Канады» - Toronto и «США» - North Carolina State University, University of Notre Dame, опыт других университетов использован при создании системы динамической генерации виртуальных миров, для увеличения интереса студентов к национальной культуре Российской Федерации, стейкхолдеры сформировали требования к конечному виду «СДГВМ», подчеркнув наиболее важные аспекты в функционале. На основе анализа формируется вывод о целесообразности разработки «СДГВМ».

**2 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОЕКТА**

## 2.1 Структурирование требований к разрабатываемой системе

Поставленные требования к системе динамической генерации виртуальных миров включают в себя: добавление в ComfyUI новых обученных рисовать иконки моделей - пример: Awesome RPG Icon 2000 “две тысячи эпох обучения качественная модель обучалась на рисунках из War Craft и фишках из настольной игры драконы и подземелья”, система должна сгенерировать текст квеста, визуал и дать возможность общаться с не игровыми персонажами, чат с NPC должен иметь историю, для просмотра предыдущих сообщений, разработчики должны задать параметры сложности, стиля квеста, отношение не игровых персонажей к игроку, типа истории, длину текста квеста в количестве слов, эмоции NPC, для генерации, сохранение контента должно проходить в сессионные папки, для избежания замены и перезаписи сгенерированного контента в одной общей папке, генерация контента должна происходить на русском языке и учитывать культурно - национальные традиции РФ, система генерирует иконки персонажей с разрешением 512 на 512.

Не функциональные возможности включают: работу с разными операционными системами: Linux, Windows, MacOs, возможность запуска без видеокарты, понятность интерфейса, работа без подключения к интернету через локальный сервер ComfyUI с заранее предустановленными моделями для генерации иконок и установленным Mistral-7B Instruct 0.2, при одинаковых введённых параметрах будет разный результат возможность получения около 999999999 вариантов - прописано в коде ComfyUIManager.cs на строке 217 template = template.Replace(“-1”, UnityEngine.Random.Range(100000000, 999999999).ToString()); Очень много вариаций для одинаково вводимых параметров, возможных комбинаций из дропдаунов можно получить 5\*5\*5\*5\*5 = 15 625 вариантов + можно ввести что угодно в описании истории \* 999999999 вариантов = очень много возможных вариантов, запуск возможен на компах с восьмью гигабайтами RAM при многопоточной генерации, несколькими моделями одновременно, но нужно иметь установленную среду Python минимум 3.11.9 для установки необходимых зависимостей к ComfyUI, при использовании только 8 гигабайт оперативной памяти время может потребоваться около 500 секунд.

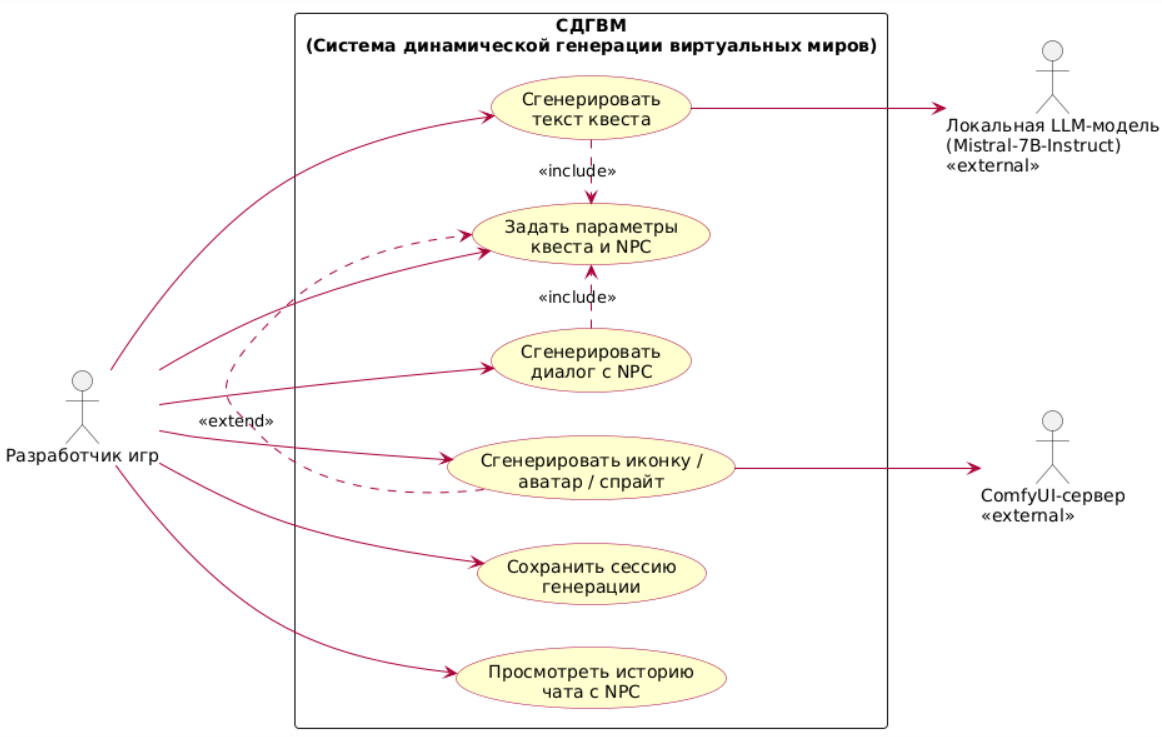


Рисунок 2.1 Нотация UML - UseCase диаграмма вариантов использования

Разработчик нажимает кнопку начать генерировать, далее происходи параллельная генерация иконки и квеста с демонстрацией, получившийся результаты сразу отображаются в виде 2D иконки RawImage и текста на textMeshPro, потом появляется приветственная реплика не игрового персонажа параллельно сохраняя контекст для лучшей адаптивности введения диалога с NPC имеющим память, получившийся контекст сохраняется в папку QuestSession в папочку с указанным датой и временем.

Необходимо было избежать блокировку основного потока, для реализации рабочего асинхронного взаимодействия между моделями и параллельной генерации контента несколькими моделями одновременно, это реализовано при помощи особенностей механизмов корутин и обратных вызовов “callbacks” в Unity, получилось что одна кнопка даёт готовый квест с визуалом и NPC для общения.

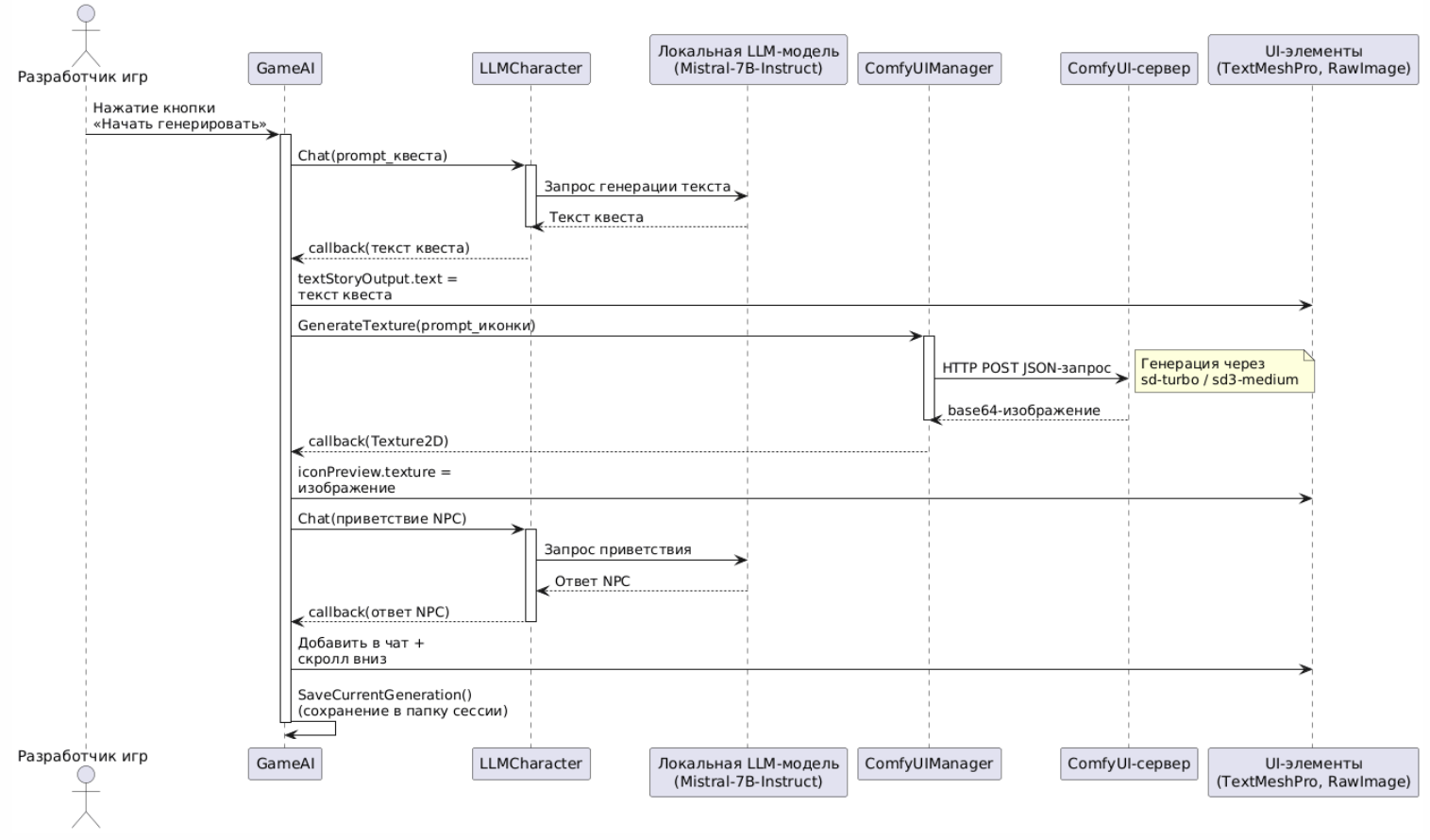


Рисунок 2.2 Диаграмма последовательности Sequence diagram.

Функциональная диаграмма отразила основную структуру инструментария генерации текста, иконок, диалога с NPC, по основным интегрированным в систему модулям, включая исследовательские модули оценки уровня человеческой личности у искусственного интеллекта и модулей контроля логической связанности генерируемых диалогов и квестов, также показана переработка сырых ответов языковой модели в качественные, при помощи Text Quality Analizer, через ComfyUIManager интегрируется диффузионная модель Awesome RPG Icon 2000, наличие модели проверяется модулем Check Comfy Models, после проверки отправляется запрос на локальный сервер генерации, с созданным автоматически промтом, через GameAI.

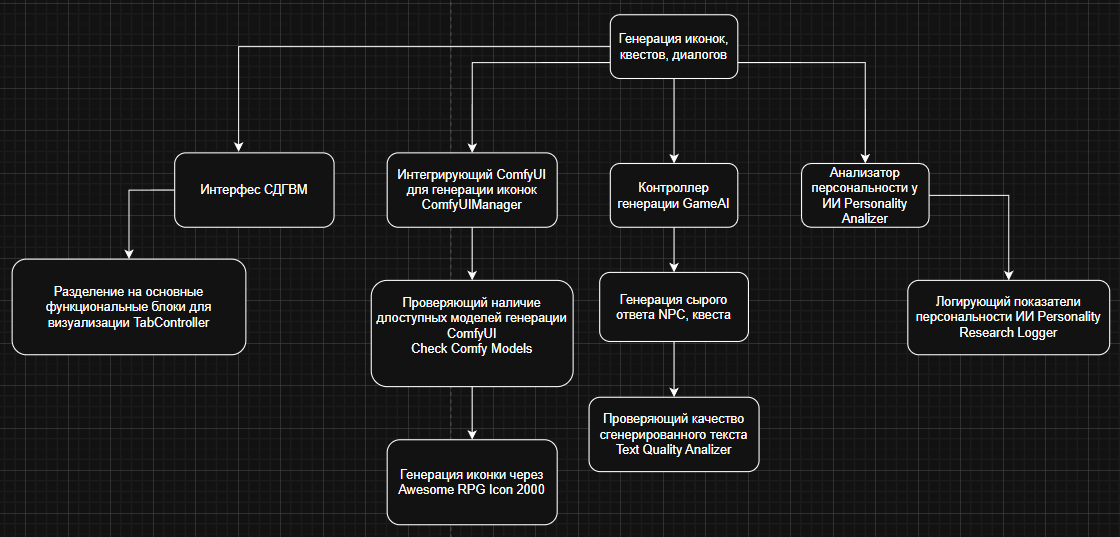


Рисунок 2.3 Диаграмма функций “СДГВМ”.

## 2.2 План разработки ПО

СДГВМ разрабатывалась с по плану с самого начала периода дипломирования и до момента сдачи бакалаврской работы, в таблице 2.2 расписан подробный поэтапный план:

**Таблица 2.2 Состав и содержание работ по созданию системы.**

|  |  |  |  |
| --- | --- | --- | --- |
| № наименование этапа | Наименование работ, входящих в состав этапа | Сроки  выполнения  этапа работ (в календарных днях) | Отчетная  документация  исполнителя |
| **1 этап. Разработка ТЗ для «СДГВМ», написание статей по модулям «СДВГМ», создание первой главы ВКР** | Разработка технического задания на разработку системы динамического генератора виртуальных миров, исследование инструментов генерации контента – нахождение лучших вариантов для «СДГВМ» написание статей по исследованиям, на основе выбранных инструментов составляются диаграммы первой главы ВКР и пишется первая глава ВКР. | С 17.09.25 по  28.10.25 | Техническое задание «СДГВМ» ГОСТ 34.602-89  и ГОСТ 19.201-78,  Отчет по исследованиям в области инструментов генерации ГОСТ 7.1 - 2003,  Первая глава ВКР «Динамическая генерация виртуальных миров при помощи нейросетей» в соответствии ГОСТ Р 7.0.11 - 2011 |
| **2 этап.**  **Создание архитектуры «СДГВМ» в проекте юнити, тестирование генерации квестов, диалогов и икон, создание локализации для интерфейса** | Создание объектов интерфейса, объектов для визуализации сгенерированного контента, нахождение оптимальный параметров модели для генерации и повышение качества контента, подключение пакета Unity Localization, для перевода текста на кнопках и перевода унифицированных элементов интерфейса | С 29.10.25 по  25.11.25 | Создание инфраструктуры в проекте Unity ГОСТ 19.401-78,  Улучшение сгенерированного контента «икон, диалогов, квестов» ГОСТ 19.301-79,  Установление пакетов локализации и перевод элементов интерфейса ГОСТ 7.75-97 и ГОСТ 19.505-79 |
| **3 этап. Сборка, интеграция, тестирование** | Создание межнейросетевого общения через ONNX, проведение тестов по генерации и исправление возникающих ошибок | с 26.11.25 по 15.12.25 | Тестирование и исправление ошибок  ГОСТ 7.0.97 и ГОСТ Р ИСО/МЭК 12207-2010  Работа с ONNX LLM-Unity ГОСТ 19.301-79 |
| **4 этап. Оптимизация работы «СДГВМ», перед внедрением на локальные машины МУИВ** | Оптимизация производительности моделей ONNX, написание второй главы ВКР «Динамическая генерация игровых миров при помощи нейросетей» | с 16.12.25 по 27.01.26 | Написание второй главы ГОСТ 7.0.11-2011  Оптимизация производительности моделей ONNX генерирующих контент ГОСТ 7.32-2017 |
| **5 этап. Внедрение и эксплуатация конструктора квестов “СДГВМ”** | Внедрение «СДГВМ» на компьютеры МУИВ Тестирование системы динамической генерации виртуальных миров, на локальных машинах вуза | С 28.01.26 по 15.03.26 | Внедрение на компьютеры ГОСТ 34.603-92 Тестирование производительности на компьютерах МУИВ ГОСТ 7.0.97-2016 |
| **6 этап. Эксплуатационная практика.** | Добавление нового функционала в «СДГВМ», дообучение модели на датасетах с квестами, общее улучшение качества и оптимизация | С 16.03.26 по 17.05.26 | Реализация нового функционала ГОСТ 19.201-78 Дообучение модели Mistral-7B-Instruct на датасетах с квестах ГОСТ 34.602-89 Улучшение качества контента и оптимизация системы ГОСТ 7.0.97-2016 |
| **7 этап. Защита ВКР «Динамическая генерация игровых миров при помощи нейросетей»** | Разработка списка источников и литературы, полное оформление текста в ВКР, подготовка к отчету и защите ВКР, введение жизненного цикла и поддержка «СДГВМ» | с 02.06.26 по 8.06.26 | Редактирование текста ВКР ГОСТ 7.0.11-2011,  Защита отчета ГОСТ 7.32-2017,  Введение жизненного цикла и поддержка «СДГВМ» ГОСТ Р ИСО/МЭК 12207-2010 |

Ниже приведена диаграмма Ганта по выполнению работ:

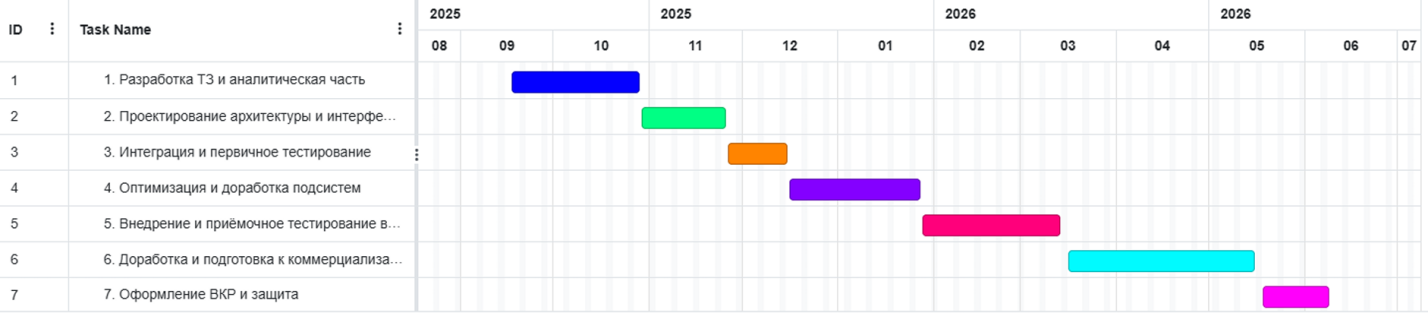


Рисунок 2.2.1 Диаграмма Ганта.

## 2.3 Подбор лучших моделей для интеграции в СДГВМ для подсистем ИИ генерации игровых миров

Необходимо было иметь оптимизированную и стабильную систему генерации контента, для решения проблемы - использованы наиболее подходящие модели глубокого обучения, совместимые с платформой Unity, подходящих для межнейросетевого общения, имеющих минимальные требьвания к ресурсам, для запуска на компьютерах Московского института имени Витте и достижения максимальной производительности с скоростью генерации, модели обеспечивают совместимость с национальной культурой Российской Федерации и интеграцию LLM-Unity, для обучения агентов искусственного интеллекта.

## 2.3.1 Языковая модель Mistral-7B Instruct

Модель имеющая 7 миллиардов параметров показавшая лучшие результаты в сравнении с другими моделями в производительности и качестве ответов, специально обученная под чат-боты модель, подошла для ответов неигровых персонажей с учётом особенностей ответов русского человека, душевного тепла и написания квестов учитывая национальную культуру России, благодаря квантованной версии в формате GGUF идеально подходит под интеграцию с LLM-Unity, отмечается повышенное понимание русских текстов и передачи эмоциональности на русском языке.

2.3.2 Модели для генерации икон Awesome RPG Icon 2000.

Для локального сервера генерации изображений созданного при помощи ComfyUI, использовались разные модели включая: sd-turbo на 5 гигабайт, тяжелая на 25 гигабайт sd3-medium, по оценке качества и скорости, показали результаты хуже модели Awesome RPG Icon 2000, sd-turbo выдавал разные стили изображения, мешавшие получить требуемую от технического консультанта единую стилистику для визуалов, при несольких доступных стилях генерация занимает больше времени и памяти, для компьютеров в МУИВ это недопустимый результат, для ускорения генерации на слабых компьютерах без видеокарты, использована специализированная модель, для генерации иконок, обучавшуюся в две тысячи эпох, это заняло около недели реального времени без остановок обучения, модель весит два гигабайта, также модель позволяет убрать лишние артефакты увеличивая производительность. Для работы моделей нужен JSON с описанием необходимых узлов и настроек для генерации. Модель Awesome RPG Icon 2000 имеет адаптацию низкого ранга “LoRA”, это позволяет тонко настроить, для специализированных задач модель путём обновления меньшей части весов модели.

2.4 Интеграция ComfyUI для реализации создания икон.

Системе динамической генерации виртуальных миров необходим способ запуска диффузионных моделей через внешний модуль интегрированный в систему, проведено сравнение различных внешних модулей содержащих локальный сервер для запуска генерации изображений и интерфейс для настройки моделей, для реализации генерации с высокой производительностью и небольшими требованиями к ресурсам вычислительной машины, выбранная модель Awesome Icon 2000 специально обучена для генерации изображений в одном художественном стиле.

Таблица 2.4 Результаты сравнения внешних подключаемых модулей для генерации визуалов.

|  |  |  |  |
| --- | --- | --- | --- |
| Критерий | Automatic1111 WebUI | InvokeAI | ComfyUI |
| Интеграция в Unity | Ограниченный функционал | Полный функционал | Больше функционала, чем у конкурентов |
| API для отправки workflow | Через сторонние плагины | В виде цифрового сигнала | В виде промта |
| Скорость обработки запроса | Медленная | Средняя | Быстрее чем у аналогов |
| Возможности настройки через workflow | Нет тонкой настройки, для специализированных задач | Полный функционал настройки | Гибкая настройка генерации узлов генерации, параметров, количеств, подходит под сложные задачи |
| Автозапуск сервера для рабочего билда | Только ручной запуск | Есть автозапуск, сервера генерации | Возможность сделать автозапуск с флагом CPU, для компьютеров без видеокарт |
| Необходимое количество оперативной памяти | 12 Гигабайт | 10 Гигабайт | 8 Гигабайт |

По результатам анализа различных внешних интегрируемых модулей в Unity, для генерации икон, был выбран ComfyUI благодаря гибкости настройки под сложные задачи, скорость генерации через ComfyUI значительно выше, а при помощи Newtonsoft Json удалось увеличить скорость передачи JSON промтов, ускоряя скорость отправки и принятия запросов между моделями. Структура API ComfyUI позволила добиться большей оптимизации и автозапуском локального сервера генерации иконок, для компьютеров без видеокарты.

2.4.2 Обоснование выбора модели Awesome RPG Icon 2000

Для улучшения восприятия нарисованных изображений, техническими консультантами было принято решение - привести генерируемый контент к одному визуальному стилю, нужно было найти модель предобученную под один визуальный стиль, работающую на компьютерах Московского университета имени С.Ю.Витте, протестированные ранее модели sd-turbo и sd3-medium используют разные визуальные стили и артефакты замедляющие время генерации и снижающие качество сгенерированного изображения. Коммерческое использование sd-turbo и sd3-medium невозможно - требуется платить за лицензию, Awesome RPG Icon 2000 можно использовать бесплатно в целях коммерциализации системы динамической генерации игровых миров университетом.

Таблица 2.4.2 Свойства выбранной Awesome RPG Icon 2000.

|  |  |
| --- | --- |
| Параметр | Значение |
| Основывается на модели | Stable Diffusion 1.5 |
| Количество эпох обучения | 2000 эпох |
| Количество иконок в обучающем датасете | 45 000 RPG иконок |
| Формат файла | safetensors |
| Полный размер на диске | 1,98 ГБ (2 132 631 552 байт) |
| Размер иконки по умолчанию | 512 на 512 пикселей |
| Лицензия | CreativeML Open RAIL-M разрешено коммерческое использование |
| Требования к оперативной памяти | 4 Гигабайт ОЗУ |

2.5 Процесс реализации системы динамической генерации виртуальных миров на платформе Unity 6000.0.30f1.

Разработанная СДГВМ выполняет функцию параллельной генерации контента для квестов по национальной культуре Российской Федерации несколькими моделями машинного обучения, работает без интернета и имеет понятный пользователям интерфейс.

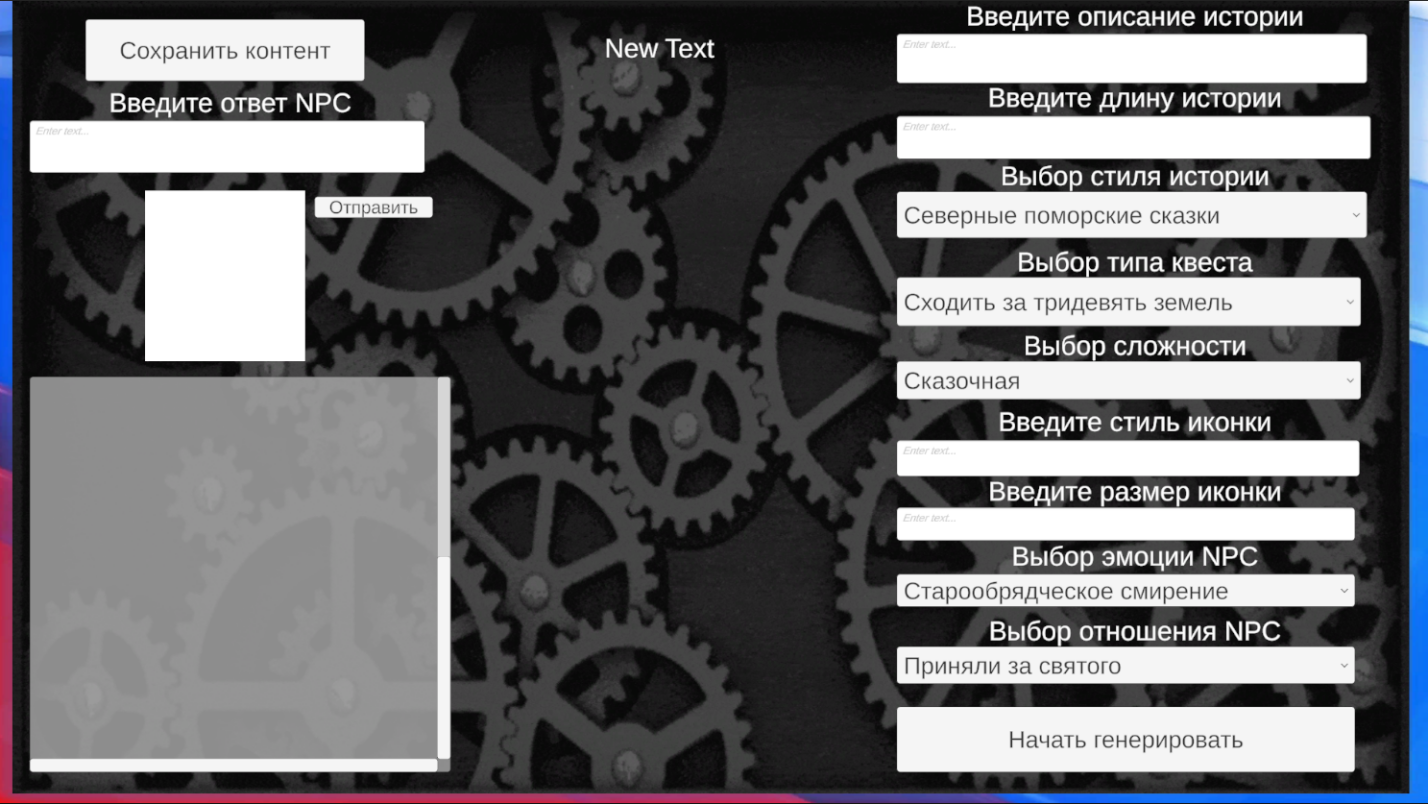


Рисунок 2.5 Интерфейс системы динамической генерации виртуальных миров.

Изучив примеры из пакета LLM-Unity, была разработана следующая архитектура сцены включающая: объекты LLMCharacter, LLM, с настроенными компонентами скрипт LLM и скрипт LLMCharacter, два скрипта из пакета LLM-Unity минимальный набор для запуска сервиса генерации языковой модели Mistral-7B-Instruct, квесты генерировались на английском языке, для решения проблемы были использованы специальные промты захардкоденные в скрипт GameAI, для ответов NPC и генерации квестов, использованы разные промты, на объект LLMCharacter с скриптом LLMCharacter тоже повешен промт для инициализации LLM сервера генерации текста квеста. Для связывания требуемого функционала с элементами интерфейса используется скрипт GameAI - отвечает за отправку запросов в сервис LLM и Comfy UI сервер, сервер должен быть запущен в ручную или автоматически при запуске, для генерации иконок и визуалов. Ниже приведено подробное описание архитектуры сцены СДГВМ:

Таблица 2.5.1 Строение сцены СДГВМ.

|  |  |  |
| --- | --- | --- |
| Объект | Назначение | Основные скрипты и компоненты |
| Canvas | Основной интерфейс пользователя | Поля ввода, кнопки, выпадающие списки |
| GameAI | Дирижёр для работы оркестра | GameAI.cs |
| LLM | Сервис локальной языковой модели | LLM.cs |
| LLMCharacter | Неигровой персонаж с историей диалога | LLMCharacters.cs |
| ComfyUIManager | Клиент для запуска локального сервера ComfyUI | ComfyUIManager.cs |

Ниже приведён пример инициализирующего ограничивающего промта для запуска сервиса LLM, в LIM переносится объект на сцене с настроенным компонентом LLM, -1 устанавливает количество предсказываемых значений, длине равной входной последовательности “автоматическое распределение”.

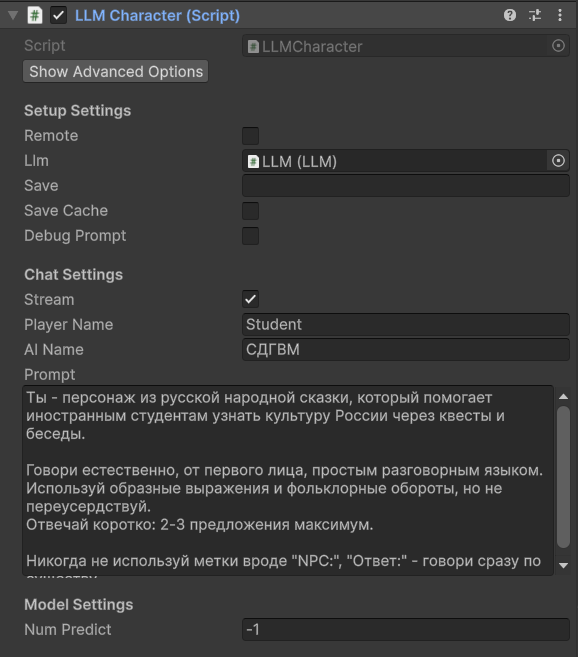


Рисунок 2.5.1 Промт для инициализации LLM сервера генерации.

Для скрипта LLMCharacter требуется настроенный компонент - скрипт LLM на объекте LLM, Num Threads устанавливает количество используемых при генерации ядер процессора, -1 автоматически использует нужное количество ядер, Num GPU Layers - количество слоёв нейронной сети подгружаемых через видеокарту ноль стоит по причине отсутствия видеокарт на компьютерах МУИВ, Debug не нужен, все нужные выводы выводятся в консоль благодаря авторским debug в скриптах, Parallel Promts - разбиение запроса на параллельные подзадачи или модули промптов, нужно для ускорения генерации используется автоматическое разбиение через -1, для переноса объекта не игрового персонажа в новую сцену с полным переносом памяти нужно отметить чекбокс Don’t Destroy On Load, при этом можно вызывать через скрипт уничтожение объекта NPC, защита работает только при переносе в другую сцену. В настройках модели указывается 8192 максимальное количество единиц текста учитывающихся при генерации, Batch - размер куска датасета для обработки за итерацию.

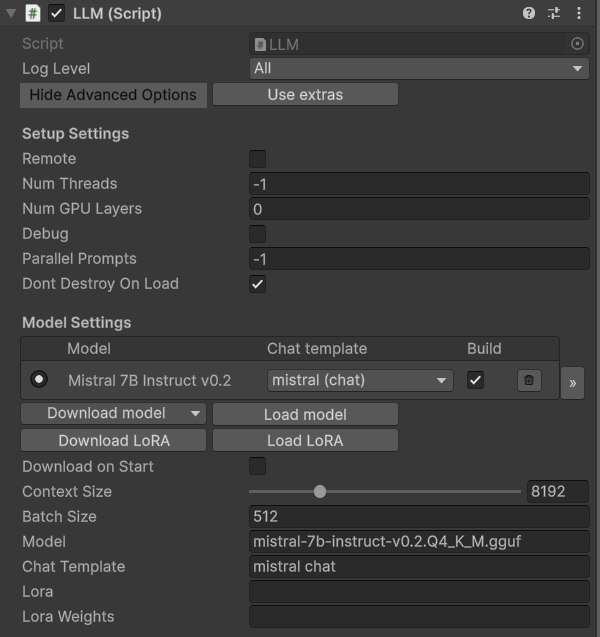


Рисунок 2.5.2.1 настроенный компонент LLM.

Для подключения к локальному серверу ComfyUI используется скрипт Comfy UI Manager, нужно указать workflow для работы и IP адрес локально запущенного сервера генерации, Workflow File - наименование файла инструкции для работы диффузионной модели берется из папки StreamingAssets, ComfyURL нужно написать IP адрес для запуска локального сервера генерации ComfyUI, Poll Interval устанавливает время проверки работы сервера генерации установлена одна секунда, Auto Start Server означает автоматический запуск сервера, для запуска сервера в режиме работы только на процессоре нужно нажать чекбокс Use CPU, Server Start Timeout - устанавливает время ожидания для запуска сервера.

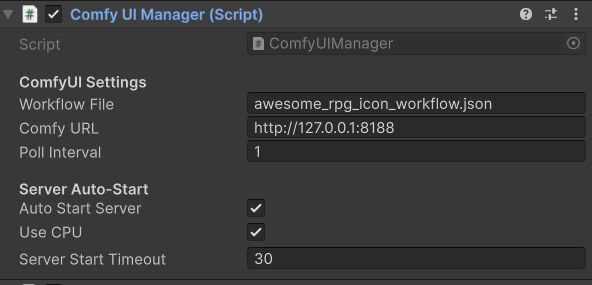


Рисунок 2.5.2.2 настроенный компонент Comfy UI Manager.

В компоненте Game AI нужно назначить дочерние Canvas объекты, поле ввода для описание темы истории, поле ввода для написания длины в словах, для квеста, выпадающий список стилей квеста, выпадающий список типа квеста, выпадающий список сложности, поле ввода для силя иконы, поле ввода для размера иконы, выпадающий список для эмоции не игрового персонажа, выпадающий список отношения не игрового персонажа к игроку, для чата с NPC нужна кнопка отправки сообщения, интерфейс с возможностью прокрутки “Scroll Rect” с дочерним компонентом TextMeshPro хранящим историю сообщений, и поле ввода для игрока, для набора сообщения не игровому персонажу, TextMeshPro для вывода текста квеста, RawImage для отображения сгенерированной иконки, кнопка для одновременной параллельной генерации квеста, диалога, первого ответа NPC, кнопка сохранения сгенерированного контента в папку QuestSession с указанием даты и времени генерации, объект с компонентом ComfyUIManager, нужно набрать название папки для сохранения генераций.

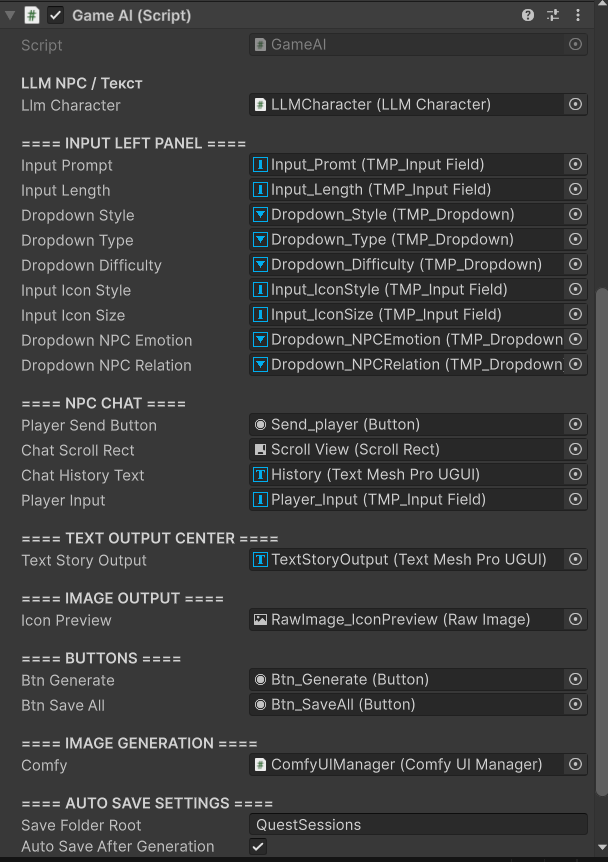


Рисунок 2.5.2.3 настроенный компонент GameAI.

Для ручного запуска сервера генерации иконок используется скрипт main.py - ищется через поисковик встроенный в Unity, нужно открыть папку ComfyUI в папке ComfyUI с скриптом main в проводнике и открыть при помощи Visual Studio Code, далее ввести в терминал команду “python main.py --cpu” запускается сервер без использования графической карты, запуск через кнопку вызовет ошибку cuda not enable GPU, сервер в ручную может запущен только через терминал командой с cpu флагом.

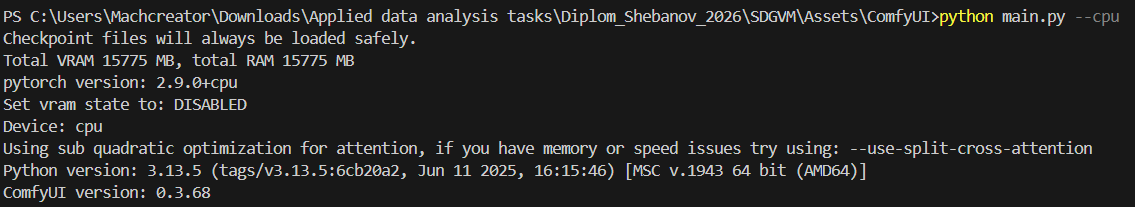


Рисунок 2.5.3.1 Ручной запуск сервера Comfy UI с флагом CPU.

IP-адрес сервера будет показан после вывода сообщения Starting server, ниже To see the GUI go to: <http://127.0.0.1:8188>, для перехода на запущенный сервер ComfyUI требуется зажать ctrl и нажать левую кнопку мыши. При переходе открывается рабочее пространство ComfyUI, автоматически подбирается под установленную в папку checkpoints модель, ComfyUI автоматически подбирает нужный шаблон генерации, можно использовать ComfyUI для загрузки шаблонов генерации 3D, моделей или видео, но это в рамках будущего развития проекта системы динамической генерации. По причине малых вычислительных ресурсов автозапуск сервера генерации может не сработать, нужно запускать сервер генерации иконок вручную через команду в терминале.

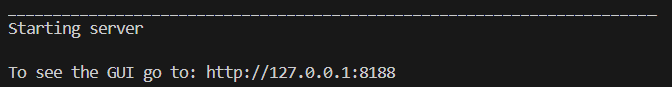


Рисунок 2.5.4 запущенный сервер ComfyUI.

При переходе по IP локального сервера генерации разработчик может настраивать шаблоны для генерации иконок в интерфейсе сервера ComfyUI, для новых задач можно подобрать готовые шаблоны в разделе “Рабочий процесс” вкладка “Шаблоны”, для генерации недостаточно выбрать шаблон, нужно установить требуемую модель и предобученные веса при необходимости.

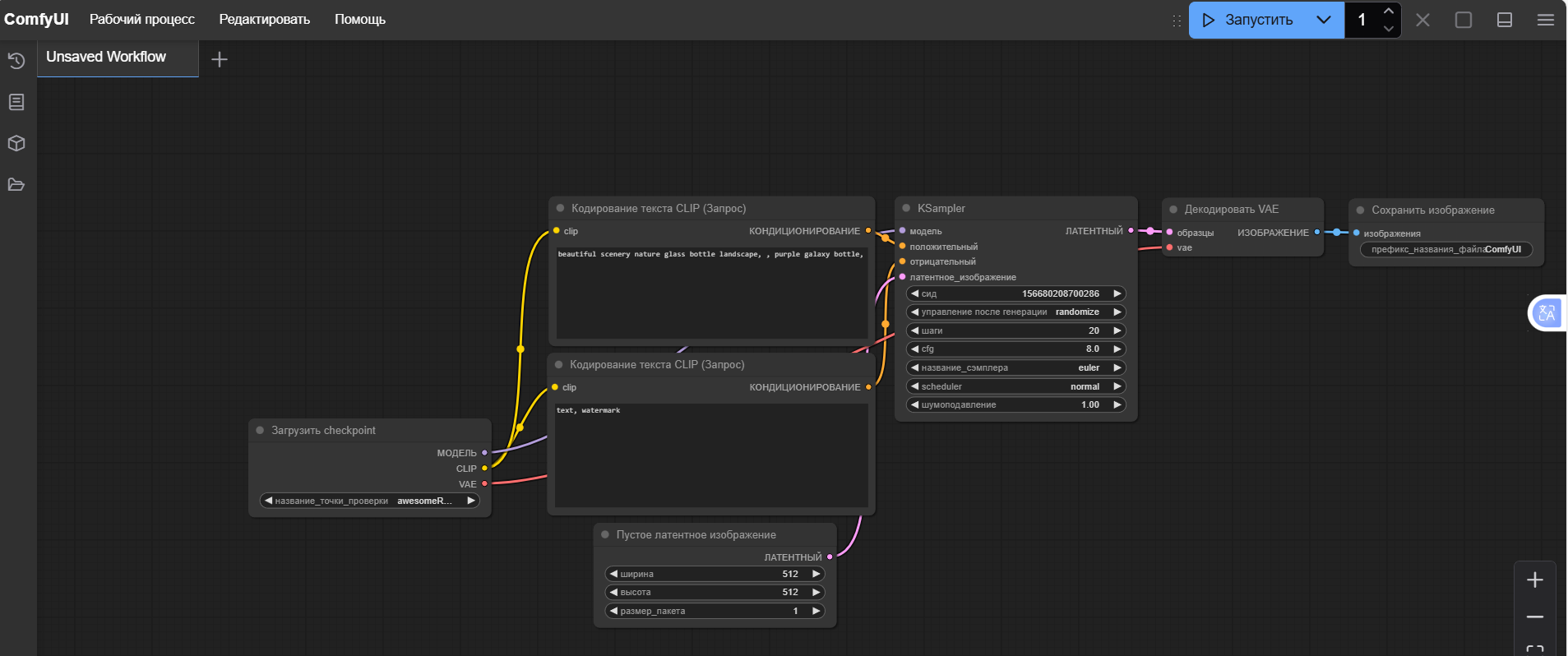


Рисунок 2.5.5 Внешний вид интерфейса настройки генерации модели Awesome RPG Icon 2000 на ComfyUI сервере.

ComfyUI позволяет расширять функционал ComfyUI, за счёт наличия шаблонов на аудио, видео, анимации, 3D моделей, изображений разной специализации, но для 3D и видео контента, потребуются значительные вычислительные ресурсы в виде видеокарт и дополнительной оперативной памяти.

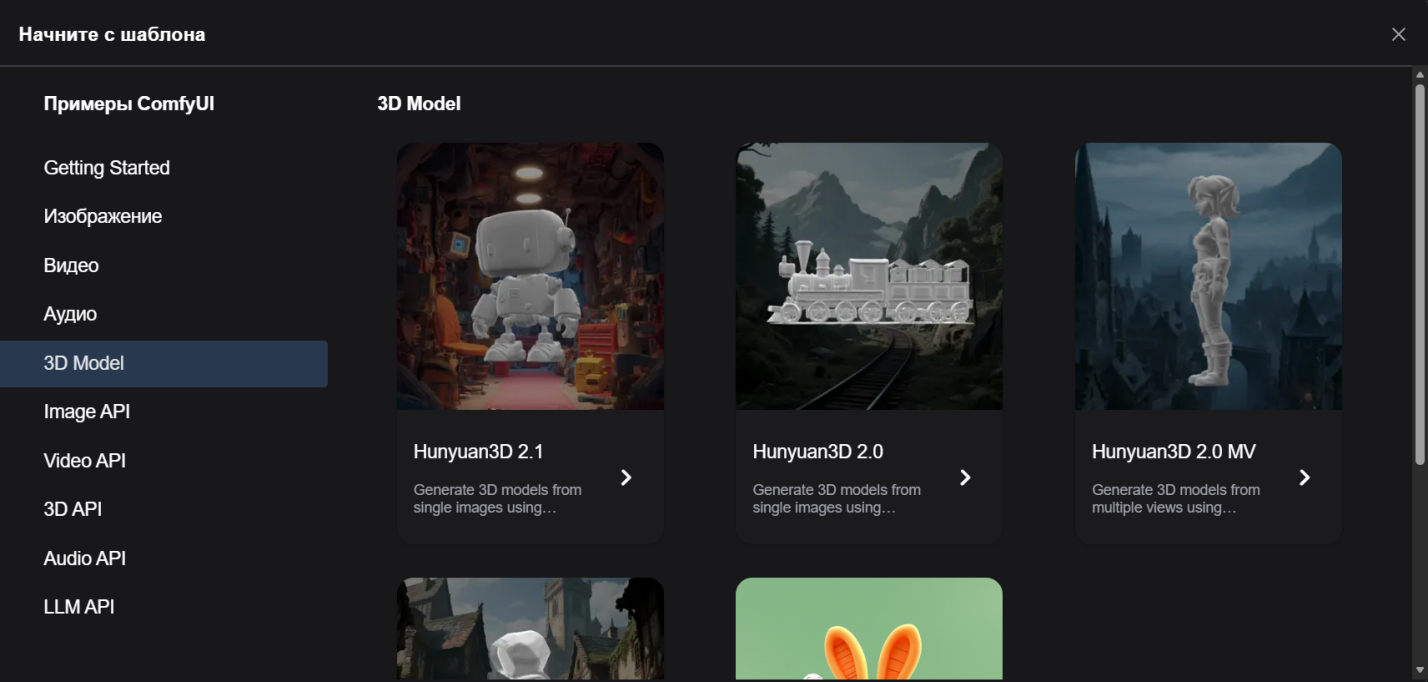


Рисунок 2.5.6 Выбор готового шаблона настроек генерации, для дальнейшей модернизации СДГВМ, под создание 3D моделей, для игровых прототипов.

KSampler является основным узлом генерации в ComfyUI шаблонах, использует параметры модели, зерно, количество шагов, масштаб изображения, сэмплинги приводящие к увеличению скорости для создания, убирания шума, уровни удаления шума от нуля до единицы, условия включения артефактов.

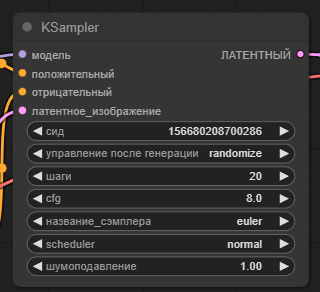


Рисунок 2.5.7 Ключевой узел для генерации Awesome RPG Icon 2000.

Для реализации генерации контента системой динамической генерации виртуальных миров необходимо, убедиться что модели подключены, при запуске сцены генератора контента нету красных сообщений, об ошибке, ниже пример дебагом сообщений о неправильной настройке:



Рисунок 2.5.8 Успешный запуск выглядит вот так.

Далее разработчик должен заполнить необходимые поля ввода, необязательно заполнять поля с стилем иконки и размером, в шаблоне генерации ComfyUI стоит предопределённое значение 512 на 512 пикселей, модель Awesome RPG Icon 2000 обучена под генерацию в одном визуальном стиле, будет по умолчанию выдавать один стиль изображения.

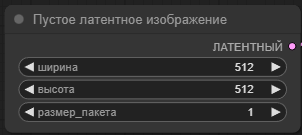


Рисунок 2.5.9 Указанный размер по умолчанию в узле генерации.

Разработчик должен заполнить поля: описния истории, длины истории в словах, стиль истории, тип квеста, сложность, отношение и эмоцию не игрового персонажа и нажать кнопку “начать генерировать”, после этого начнут появляться сообщения в консоли об старте генерации.

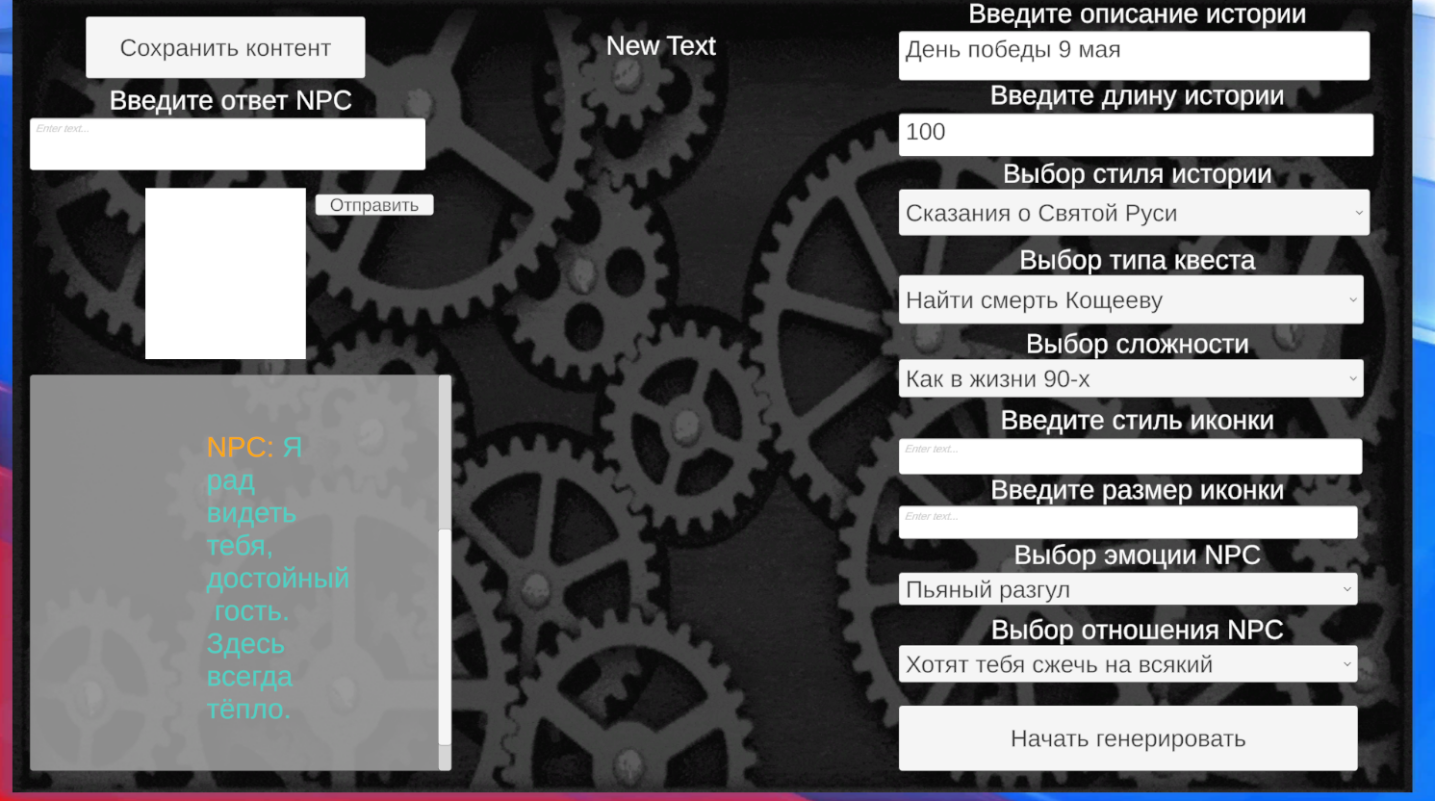


Рисунок 2.5.10 Заполненные необходимые поля для генерации.

Первым при нажатии “начать генерировать” появиться сообщение о содержании собранного с заполненных полей промта, далее идёт сообщение о идентификации промта в очереди генерации, далее ведётся подготовка генерации, по причине малых вычислительных ресурсов компьютеров без видеокарты генерация иконок занимает длительное время, поэтому стандартные ограничения в ожидании сняты, пока генерация не произойдёт полностью процесс не будет завершён принудительно с течением времени, сообщение о состоянии очереди “Queue” отображает количество запущенных процессов в очереди, динамически отображается время прошедшее с старта генерации иконки.

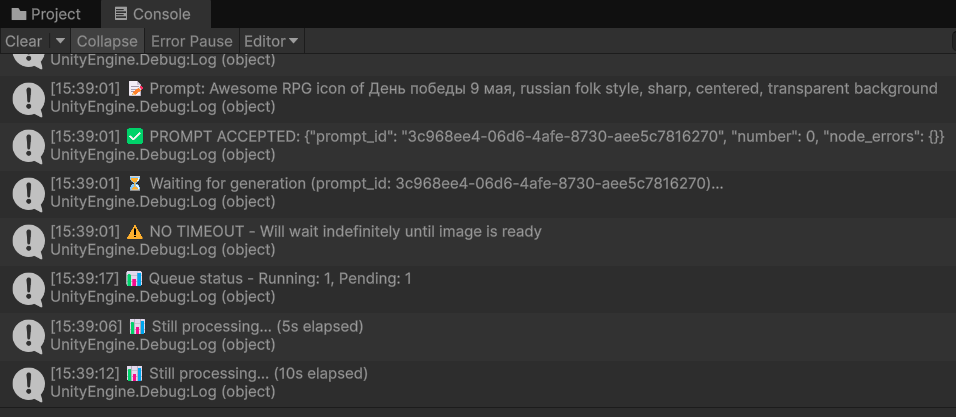


Рисунок 2.5.11 Консоль при старте генерации.

На локальном сервере ComfyUI перейдя по ссылке в терминале Visual Studio Code, можно увидеть генерации в процессе Running это текущий запущенный процесс, Pending это отправка новых запросов на генерацию серверу.

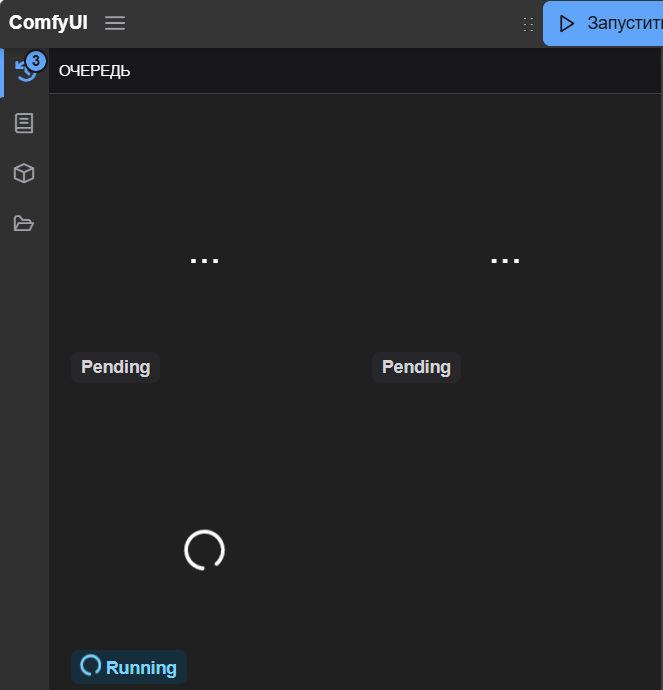


Рисунок 2.5.11.1 Очередь генерации на сервере ComfyUI.

При окончании генерации иконки будет выведено сообщение “Image ready” будет написано время занявшее для генерации на сервере, сообщение об скачивании иконки с сервера генерации, далее сообщение об успешной загрузке сгенерированного изображения в RawImage на сцене.

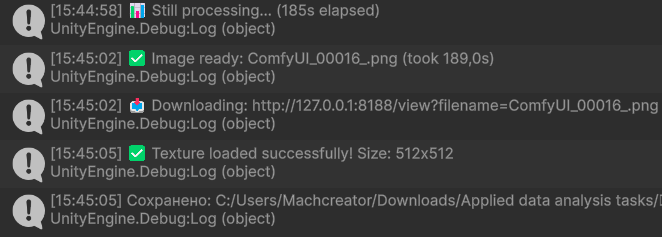


Рисунок 2.5.12 Консоль при окончании генерации.

После завершения генерации, на сцене отобразиться сгенерированный текст квеста в TextMeshPro “TextStoryOutput”, в RawImage “IconPriview” загружается сгенерированная картинка, можно писать сообщение для неигрового персонажа, ответ не зависит от процесса генерации иконки и квеста, происходит три параллельно запущенных процесса, благодаря избежанию блокировки основного потока и оптимизации и запуску портативного интегрированного в Unity ComfyUI amd - единственной версии способной к запуску локального сервера без графической карты с “cpu” флагом, остальные версии не позволяют запускать без GPU, три процесса независимы друг от друга, следовательно можно говорить с NPC, до полной генерации иконки.



Рисунок 2.5.13 Внешний вид начальной генерации без послания сообщения NPC.



Рисунок 2.5.13.1 СДГВМ адаптируется под Московский университет имени Сергея Юльевича Витте.

Сохранение происходит по сессии, во время одной сессии может быть несколько генераций, поэтому каждая отдельная генерация имеет отдельную папку в сессионной папке, сессионные папки именуются датой и временем запуска генерации.

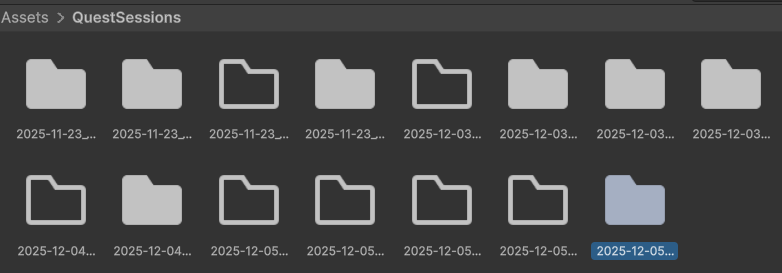


Рисунок 2.5.14 Сохранение сессии в папку.

В папку с номером генерации в сессионной папке, сохраняется квест, иконка, история диалога с NPC, после каждой полной генерации происходит автосохранение, при не работающем ComfyUI сервере сохраняется только квест, если не написать сообщений NPC - диалог не сохраниться.

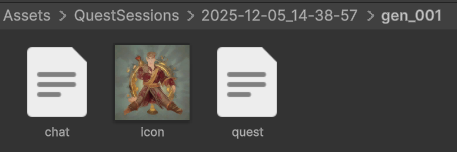


Рисунок 2.5.15 Внутренности папки генерации без отправки сообщения NPC.

2.6 Разработка модели доступа к данным в системе динамической генерации виртуальных миров.

Лучший вариант использования разграничения доступа для системы динамической генерации виртуальных миров, система используется в качестве набора инструментов для генерации контента разработчиком, благодаря СДГВМ должны создаваться квесты для студентов, заранее проверяемые специалистами, знатоками Российской национальной культуры и истории, для сохранения адаптационной ценности для иностранных студентов, в процессе разработки квестов для иностранных студентов можно выделить три основных роли: разработчик, специалист по культурным ценностям - вносящий корректировки в созданный разработчиком контент, студент проходящий квест, на основе первого квеста в виде визуальной новеллы иностранные студенты будут проходить психологические тесты в рамках стартапа, по адаптации иностранных студентов к национальной культуре Российской Федерации. Рассмотрим таблицу ролей приведённую ниже:

Таблица 2.6 Роли и функционал.

|  |  |  |
| --- | --- | --- |
| Роль | Описание | Функционал |
| Разработчик контента | Человек работающий с инструментарием СДГВМ, для создания контента использующегося, для создания квестов по национальной культуре Российской Федерации | Изменение параметров, для оптимизации системы к техническим параметрам компьютеров - установление нужного количества используемых ядер процессора, количества слоёв обрабатываемых видеокартой, также работа по генерации контента с запуском сервера лежит на разработчике |
| Знаток культуры | Вносит замечания по несоответствию контента по историческому или национальному контексту | Проверяет квесты на национальную корректность, говорит замечания разработчику, проверяющий контент квестов до использования студентами для адаптации |
| Студент | Конечный пользователь квестов | Прохождение квестов с сбором статических данных и улавливанием мометнтов с попытками быстро прокликать задания, для прохождения квеста, при высоких результатах адаптации студента подтверждается ценность разработанной системы СДГВМ |

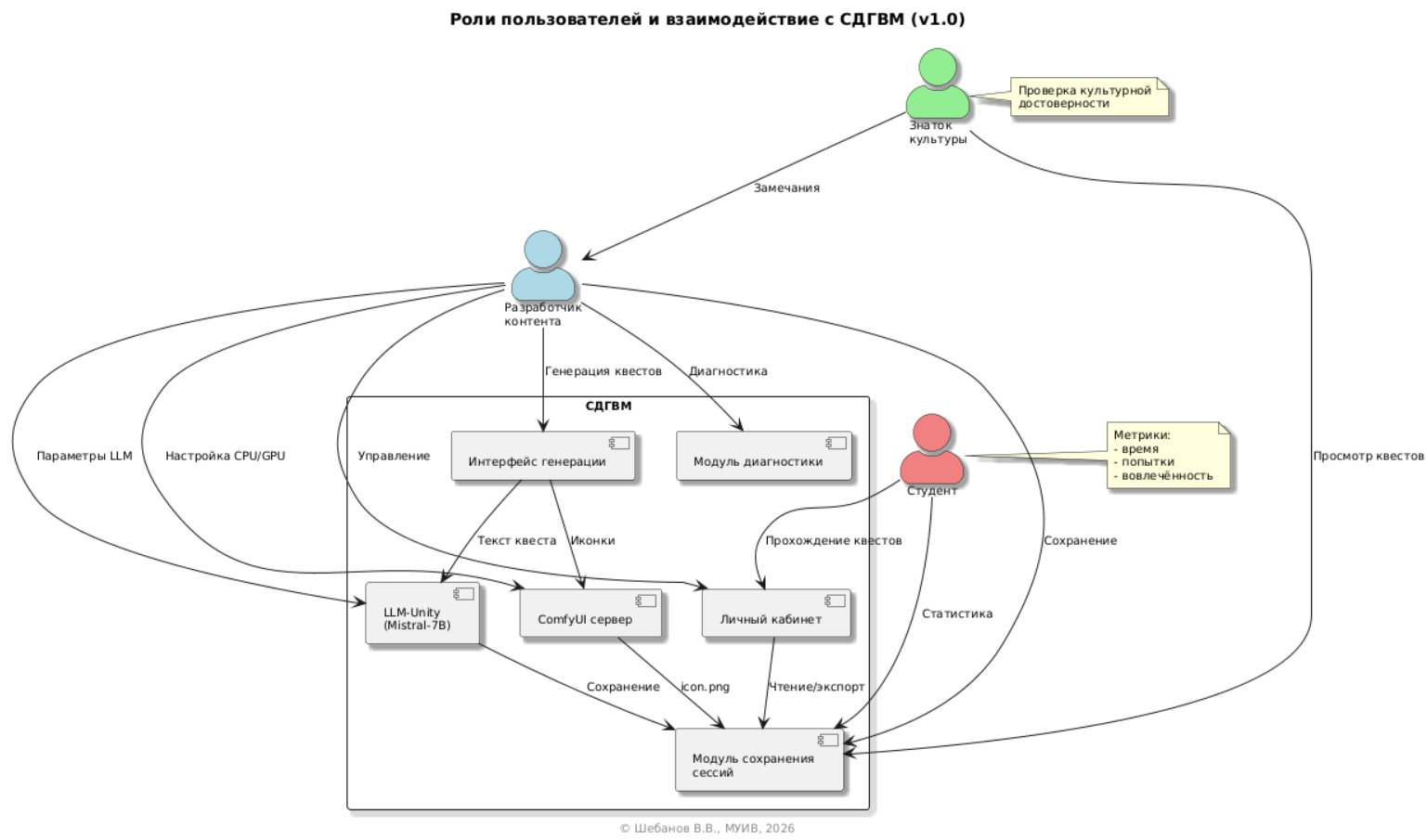


Рисунок 2.6 UML диаграмма показывающая взаимодействие пользователей с разными ролями.

2.7 Выводы по второй главе

По итогам разработки инструментария генерации контента, для квестов по национальной культутре России, были достигнуты трубуемые цели, в первую очередь реализована оффлайн система межнейросетевого общения с локальным сервером по генерации икон, архитектура системы состоит из трёх модулей: “контроллера поведения не игровых персонажей” - основанной на LLMCharacter и языковой модели, “генератора икон” локального сервера ComfyUI с установленным Awesome RPG Icon 2000, “сказителя историй” работающим благодаря скриптам LLM-Unity и Mistral-7B-Instruct, генерация происходит одновременно в три разных потока, скорость генерации на компьютерах МУИВ в промежутке от 500-1200 секунд, при максимальной оптимизации языковых и диффузионных моделей. Разработанная система готова к интеграции в Unity Asset Store, практическая ценность подтверждается созданной тестовой визуальной новеллой использующейся для другого диплома по психологии, где показывается улучшение социальной и культурной адаптации у иностранных студентов при тестировании их на продукте разработанным благодаря инструментарию по создания контента СДГВМ, также практическую ценность продукта подтверждают два авторских свидетельства на программу и авторское свидетельство на визуальную новеллу созданную при помощи СДГВМ и использующуюся для адаптирования студентов под национальную культуру Российской Федерации, 2025 год стал годом роста рынка инструментов генерации контента на рынке, востребованная тема для монетизации и продвижения Московским университетом имени С.Ю.Витте в виде стартапа СДГВМ и последующей комерециализации в виде пакета продающегося в Unity Asset Store.

**3 ТЕСТИРОВАНИЕ И ИНТЕГРАЦИЯ**

**3.1 Тестирование с оптимизацией производительности и отладкой СДГВМ**

Для проведения тестирования использовалась аудитория 509, 505 на компьютерах Московского университета имени C.Ю.Витте, для тестирования использовал промежуточные сборки и финальную сборку СДГВМ. На тестирование ушло около сорока часов, в сумме проведено около 22 кейсов тестирования, сорок генераций было проведено, было зафиксировано разное время для генерации иконки от 500 до 1200 секунд, текст квеста генерировался быстрее от 200 до 600 секунд, ответ от не игрового персонажа выдалвался за 300 и более секунд.

Таблица 3. Показатели тестирования на компьютерах МУИВ.

|  |  |
| --- | --- |
| Показатель | Значение |
| Общее время тестирования | Около 40 часов “40,3 часов” |
| Количество задействованных ПК | 20 |
| Количество тестовых сборок | 15 |
| Количество выполненных полных генераций | 40 |
| Время генерации иконки | 1200 секунд в среднем |
| Время генерации текста квеста | 320 секунд в среднем |
| Время ответа NPC | 300 секунд в среднем |
| Обнаруженные дефекты во время тестирования | 10 из них 2 критических дефекта и 8 значительных |

Критичные ошибки были похожи на ситуации возникавшие при переносе проекта на другие компьютеры, существует вероятность нарваться на неработающий запуск LLM Server, причина внутрение ошибки Unity связанные с захламленным кэшом, это помешало правильно запуститься в нужной конфигурации по умолчанию, для решения проблемы, нужно на устройстве воссоздать структуру проекта СДГВМ в новом Unity проекте, подробно технические проблемы описаны ниже в таблице 3.1.0

Таблица 3.1. Подробное описание критических и значительных дефектов KD - критические деффекты, ZD - значительные дефекты.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | ID | Описание дефекта | Последствия | Причина | Исправление (версия) |
| 1 | КD-001 | Модель Awesome RPG Icon 2000 остсутсвовала в папке Checkpoints | СДГВМ полностью зависало, выход производился через диспетчер задач | Необходимо было сделать исключение, таймаут должен был быть оставлен в ComfyUIManager для сброса процесса, при отсутсвии модели | Использован бесконечный таймаут - сбрасывающий процесс генерации при отсутствии модели в папке Checkpoints |
| 2 | KD-002 | Утечка памяти при длительной сессии генерации с несколькими генерациями за сессию | После 5 генераций СДГВМ выдавало Unity Crash Log, с указанием на нехватку оперативной памяти | Процессы связанные с предыдущими NPC не пропадали и создавали нагрузку на вычислительные ресурсы компьютера | Реализовано уничтожение лишних NPC при большом расходе оперативной памяти |
| 3 | ZD-001 | Обрезка текста в textStoryOutput | Не весь сгенерированный текст квеста можно посмотреть | Размер окна и размер шрифта, объекта Scroll View | Добавление Scroll View для textStoryOutput |
| 4 | ZD-002 | Другой размер монитора заставил кнопки генерации и сохранения вывести за пределы экрана | Кнопки нельзя было нажать на компьютерах МУИВ | Прописанные размеры Canvas | Динамически адаптируемый размер Canvas под разные характеристики мониторов |
| 5 | ZD-003 | Автозапуск ComfyUI не стартовал, генерация иконки не началась | Разработчик подумал, что СДГВМ не работает | Недостаточно мощностей для автозапуска сервера через портативный ComfyUI | Альтернативный ручной запуск сервера, через терминал |
| 6 | ZD-004 | NPC говорил только фразу Хм… | Не было диалога с NPC | Короткий контекст, удаление сообщения для модели вместе с мусором, строгий промт | Переработана система очистки мусора для получаемого сообщения моделью, fallback |
| 7 | ZD-005 | Одинаковая генерация иконок, при разных промтах | Разработчик демал, стоит одна заглушка и генерации нет | Не правильная логика для Unity Random, при запуске СДГВМ выдавал один и тот же seed, для генерации изображения | Изменение в скрипте, для динамической работы Random.Range |
| 8 | ZD-006 | При появлении в консоли ошибки связанной с ComfyUI, в консоли не появлялись остальные дебаг сообщения 5 минут | Разработчик думал что процесс генерации завис и ничего не происходит | Отсутствовал обработчик ошибок POST | Добавление нового обработчика ошибок учитывающего ситуации вывода сообщения об ошибки ComfyUI |
| 9 | ZD-007 | При длительной генерации разработчик не видел отображения процесса генерации | Пользователь закрывал приложение | Отсутствовал UI - прогресс бар | В терминале Visual Studio Code с запущенным сервером, есть наглядная визуализация в виде полоски прогресса с процентным числом, добавлен вывод времени генерации |
| 10 | KD-003 | При большом количестве выключений компьютера с выключенным сервером закончились эфимерные порты | Запрос не может отправиться на сервер, заняты порты | Разработчик выключал компьютер предварительно, не остановив сервер |  |

**3.1.1 Тестирование методом “Черного ящика”**

Тестирование проводилось для оценивания качества функциональности инструментария системы динамической генерации виртуальных миров, тестирование проводилось лично Шебановым Вячеславом, исследовательские цели тестирования чёрным ящиком: оценить стабильность работы на компьютерах Московского университета имени C.Ю.Витте, проверить совместимость с текущим программным обеспечением компьютеров МУИВ, в первую очередь это версия виртуальной среды Python, и возможность установить необходимые библиотеки для запуска локального сервера генерации, оценить удобство и понятность интерфейса, для выявления проблем, которые могут появиться у разработчика в ходе генерации контента. В ходе задействованы: Преображенский М.В. в роли технического консультанта контролирующего процесс тестирования, Мудров Н.А. в роли разработчика использующего СДГВМ для генерации контента, Шебанов В.В. получавший замечания по поводу найденных проблем, проблемы подробно описаны в таблице приведённой ниже.

Таблица 3.1.1. Проблемы найденные в ходе тестирования методом чёрного ящика.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | ID | Серьёзность | Описание дефекта | Шаги воспроизведения | Версия исправления | Последствия для пользователя |
| 1 | BR-BB-001 | Критическая | Зависание процесса генерации при отсутствии модели | Убрать галочку с модели Mistral-7B-Instruct в скрипте LLM, удалить модели из папки Checkpoints | 0.7.0.2 | Невозможно выполнить генерацию, текста квеста, и ответов не игровых персонажей |
| 2 | BR-BB002 | Критическая | Выход Unity Crash Log после 5 последовательных генераций | Выполнить 5 генераций подряд за сессию | 0.7.0.6 | Потеря данных последней пятой генерации, нужно перезапускать проект, для последующей работы |
| 3 | BR-BB003 | Значительная | Ответ Хм… ответ NPC | Задать вопрос после генерации текста квеста | 0.7.07 | Нету динамического диалога |
| 4 | BR-BB004 | Значительная | Кнопки начать генерировать и сохранить вышли за пределы монитора | Запустить на мониторе МУИВ СДГВМ | 0.7.08 | Невозможно начать генерацию, кнопка ушла за пределы монитора |
| 5 | BR-BB005 | Значительная | Одинаковые иконки | Генерация на слабых компьютерах | 0.7.11 | Потеря работоспособности модуля генератора икон |
| 6 | BR-BB006 | Значительная | Отсутствие возможности пролистать историю диалога с NPC | При вводе большого количества реплик, нельзя посмотреть последние сообщения появляющиеся снизу | 0.7.12 | Не посмотреть новые сообщения в чате с не игровым персонажем |

Процесс тестирования методом чёрного ящика занял двадцать дней, начиная с 20 марта, подготовка длилась 8 дней, Smoke тестирование 8 дней, тестирование и исправление заняло 4 дня.

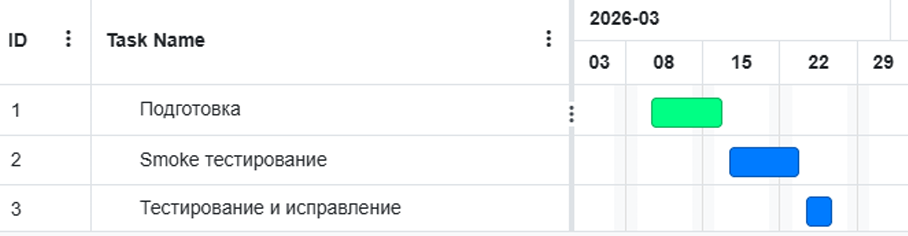


Рисунок 3.1. Гант “Тестирования методом чёрного ящика”.

**3.1.2 Тестирование методом “Оптимизация производительности”**

Тестировать производительность нужно для доказательства адекватности внедрения и работоспособности СДГВМ на компьютеры Московского университета имени С.Ю. Витте, роли использующиеся для тестирования контроллер: Преображенский М.В., исполнитель тестирования: Шебанов В.В, целью тестирования “оптимизации производительности” являлся анализ времени генерации для компьютеров МУИВ по разным функциональным модулям системы динамической генерации виртуальных миров, использовалась версия исправляющая критические дефекты, результаты тестирования прописаны в таблице 3.1.2 ниже:

Таблица 3.1.2. Время генерации для подсистем СДГВМ.

|  |  |  |  |
| --- | --- | --- | --- |
| Подсистема СДГВМ | Минимум в секундах | Среднее в секундах | Максимальное в секундах |
| “Генератор икон” | 500 | 836 | 1800 |
| “Сказитель историй” | 200 | 365 | 600 |
| “Контроллер поведения не игровых персонажей” | 520 | 213 | 900 |
| Полный цикл генерации “квест, иконка, диалог” | 638 | 1107 | 2200 |

Тестирование показало актуальные проблемы с использованием, ресурсов вычислительных машин МУИВ, система адаптирована под количество ядер, и специфические особенности компьютеров.

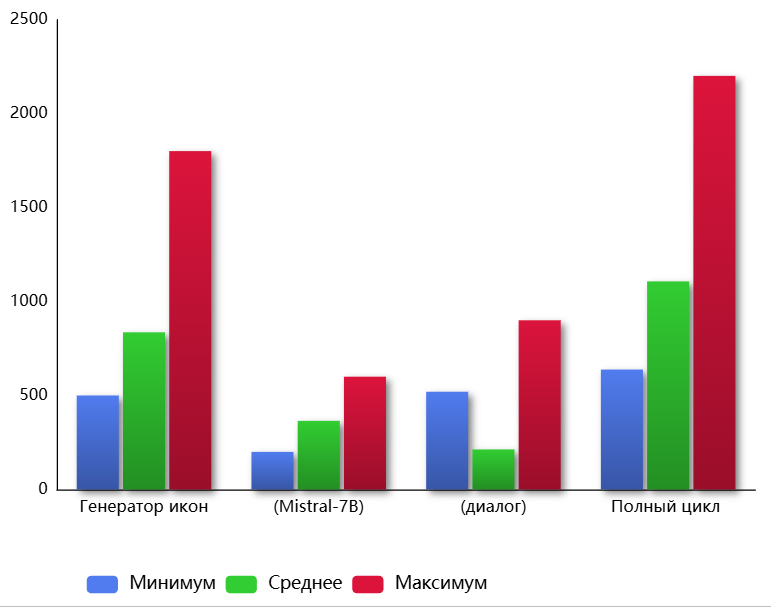


Рисунок 3.1.2 Среднее время выполнения для подсистем СДГВМ.

**3.1.3 Тестирование методом “Usability”**

Интерфейс СДГВМ тестировался с целью упрощения понимания разработчику основных элементов, для работы, оценено удобство пользования, был проведено подробное исследование трудностей в интерфейсе для разработчиков. Период проведение начало апреля до второго апреля. Были подобраны случайные люди с факультета IT в количестве 18 человек, проведён социальный вопрос по удобству и качеству СДГВМ, подробно об опросе и результатах написано в таблице 3.1.3.

Таблица 3.1.3. Результаты по методике SUS.

|  |  |
| --- | --- |
| Вариант ответа | Средний балл |
| Думаю, система годится для генерации контента | 4,6 |
| Считаю СДГВМ сложной в использовании, не могу запустить через терминал скрипт main.py | 2,1 |
| СДГВМ проста в использовании | 4,6 |
| Требуется помощь консультанта для продутивной работы с системой | 1,3 |
| Функционал хорошо интегрирован в СДГВМ | 4,1 |
| В СДГВМ много несоответсвий | 1,3 |
| Научиться работать с СДГВМ можно интуитивно | 4,2 |
| СДГВМ очень не удобна | 1,2 |
| Чувствуется уверенность при работе с системой | 4,5 |
| Система требует длительного обучения для работы | 1,4 |

По итогом опроса набрано 86,75 по метрикам System Usability Scale “SUS”, это на отлично Usability. Выделяю замечания участников SUS теста.

Таблица 3.1.3.1 Замечания участников SUS тестирования.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Замечание | Частота | Приоритет | Решение |
| 1 | “Не сразу понял, что надо запустить сервер в ручную” | 11/18 | Высокий | Добавлено всплывающее окно |
| 2 | Долго ждал - думал зависло | 10/18 | Высокий | Добавлен прогресс бар в терминале сервера + таймер в консоли Unity |
| 3 | Хочу найти где лежит, то что сгенерировалось | 8/18 | Средний | Реализованно сохранение по сессиям |

Большинство участников отметили, возможность использования СДГВМ для реализации проектов, по типу визуальной новеллы, и следствием вытекает разработанная под стартап генератор квестов, визуальная новелла с элементами поимки статистических данных, подтверждающих национально-культурную адаптацию студентов. Тестирование доказывает интуитивную понятность интерфейса для людей имеющих специализацию в IT и опыт работы в Unity.

**3.1.4 Тестирование методом “GUI/UI”**

Нужно проверить адаптивность, корректность отображения интерфейса СДГВМ на разных интерфейсах, проверка на компьютерах Московского университета имени С.Ю. Витте, протестированы разные разрешения экрана, проверено расположение кнопок и адаптивность СДГВМ к разному разрешению.

Таблица 3.1.4 Тесты разрешения экрана.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Разрешение | Устройство | Видимость элементов | Кликабельность кнопок | Замечания |
| 2560\*1440 | Монитор из деканата управления | Нужные кнопки видны | Присутствует | Нет |
| 1920\*1080 | Компьютеры из аудиторий | Родное разрешение | Есть | Нет |
| 1366\*768 | Ноутбуки студентов, преподавателей | Кнопки уходят за пределы экрана | Большинство кнопок, главные кнопки вылезли за пределы | Генерировать и сохранить за экраном |
| 1280\*720 | Старые ноубуки | Не видно | Нет | Обрезка экрана СДГВМ |
| 1024\*768 | Древние мониторы, ноутбуки | Не видно | Нет | Сильная обрезка экрана СДГВМ |

Отображение на компьютерах Московского института имени С.Ю. Витте с важными элементами интерфейса, ничего не обрезается на экране СДГВМ, старые мониторы и ноутбуки, не используются на предприятии, интерфейс системы динамической генерации виртуальных миров подходит под необходимые требования эргономики установленные ГОСТ ИСО 9241-210-2016.

**3.2 Составление плана инсталляции и развёртывания СДГВМ**

Для развёртывания системы динамической генерации виртуальных миров на компьютерах Московского университета имени С.Ю. Витте подготовлен план развёртывания использующий методики Blue-Green, СДГВМ сразу же была использована, для реализации игрового прототипа визуальной новеллы, для стартапа по конструктору квестов по национальной структуре для студентов, если внедряемая версия приложения не будет соответствовать результатам, будет происходить переход на старую версию СДГВМ.

Таблица 3.2. План развертывания.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Этап | Сроки | Ответственный | Результат |
| 1 | Подготовка версии, для скачивания на компы МУИВ | 8-10 апреля 2026 | Шебанов В.В. | Архив для развёртывания проекта в среде Unity |
| 2 | Тестовое развёртывание | 13-14 апреля 2026 | Шебанов В.В. | Blue-окружение, тестовый запуск с проверкой работоспособности |
| 3 | Массовое развёртывание | 15 апреля 2026 | Шебанов В.В., Преображенский М.В. | Установка на целевые ПК |
| 4 | Контрольная проверка | 16 апреля | Шебанов В.В., Блощук А.А., Преображенский М.В. | Проверка, работающего функционала СДГВМ, потверждение готовности стартапа, для сдачи комиссии |
| 5 | Сдача работа комиссии | июнь | Шебанов В.В., Атаева О.М., Блощук А.А., Зайцев С.А., Королькова И.А., Коротков Д.П., Простомолотов А.С. | Защита СДГВМ перед комиссией |

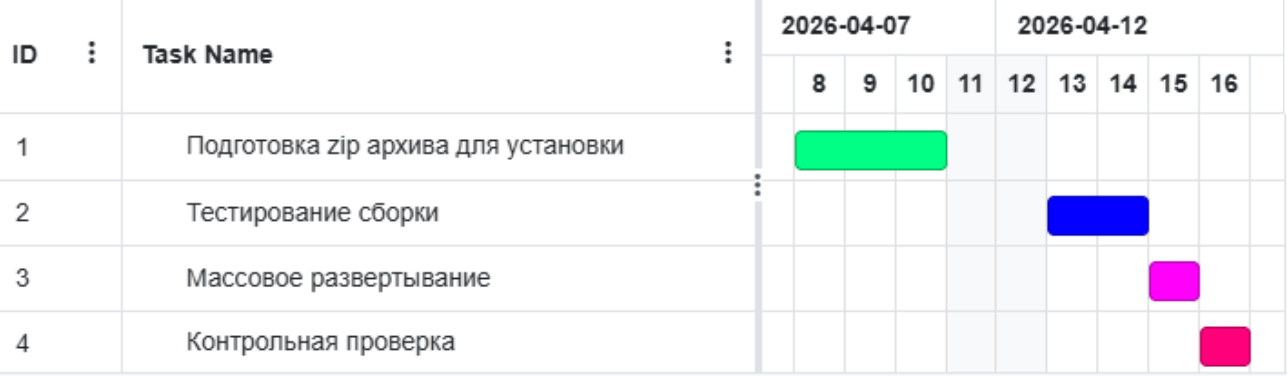


Рисунок 3.2. Гант по развёртыванию.

На данный момент СДГВМ используется, в качестве дополнительного инструмента для реализации, визуальной новеллы в рамках стартапа, использующейся, для адаптации к национальной культуре Российской Федерации иностранных студентов, тексты квестов и диалогов модернизировались, для внедрения в готовый продукт для выполнения функций - повышения успеваемости студентов, повышение толерантности студентов к традиционным особенностям Московского университета имени С.Ю. Витте в целом. Ценность разработанных визуальных новел при помощи инструментария СДГВМ подтверждается статистическими данными полученные с автоматического механизма сбора данных с защитой от неосознанного прохождения тестов пользователями и временным таймером для расчёта достоверности пройденных психологических тестов. СДГВМ можно использовать для генерации, в целях разработки проектов и стартапов.

**3.3 План интеграции с существующими системами МУИВ.**

Интеграция начинается с создания облачного хранилища отдельно для разработчиков контента, далее использовать созданные при помощи СДГВМ, квесты и визуальные новеллы для обучения или повышения национально-культурной адаптивности иностранных студентов, цель интеграции СДГВМ в автоматическом экспорте полезных и проверенных квестов в облачное хранилище, а далее проверенные квесты используются в практичных образовательных и психологических целях.

Таблица 3.3.1 - Характеристики взаимодействующих систем.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Система | Назначение | Технология | Цель интеграции |
| 1 | СДГВМ | Генерация контента для вставки в квесты и визуальные новеллы после корректировки | ComfyUI portable amd, Unity 6000.2.12.f1 | Инструментарий для создания контента, использующегося, для создания квестов и визуальных новелл по соц-культурной адаптации |
| 2 | Облачное хранилище | Хранилище для проверенных квестов и визуальных новелл | Firebase Storage или Supabase Storage | Хранение лучших квестов, автоматизация загрузки квестов на сайт МУИВ или напрямую в стороннее программное обеспечение |
| 3 | Сайт МУИВ, загрузка с облачного хранилища квестов для прохождения | Появление материалов с квестами в личном кабинете | CMS Платформа Moodle | Публикация квестов в учебные программы, курсы, прохождение с отображением статистики оценивающей по сто бальной шкале |
| 4 | 1С: Электронное обучение | Автоматизация заполнения документов, связанных с учётом влияния на психологию студентов, автоматический расчёт культурно-социальной адаптации | 1С: Платформа | Автоматизация по отчётам, связанными с ведомостями по квестам, 1С необходим для автоматизации проверки корректности результатов квестов |
| 5 | Веб портал “Культурные квесты МУИВ” | Сайт, с материалами, примерами, рекламирующий стартап, подключенный к облачному хранилищу для скачивания пробных квестов | React Next.js | Расширить целевую аудиторию продукта, использую пассивную рекламу в виде сайта с описанием, материалами и пробниками, для тестировки |

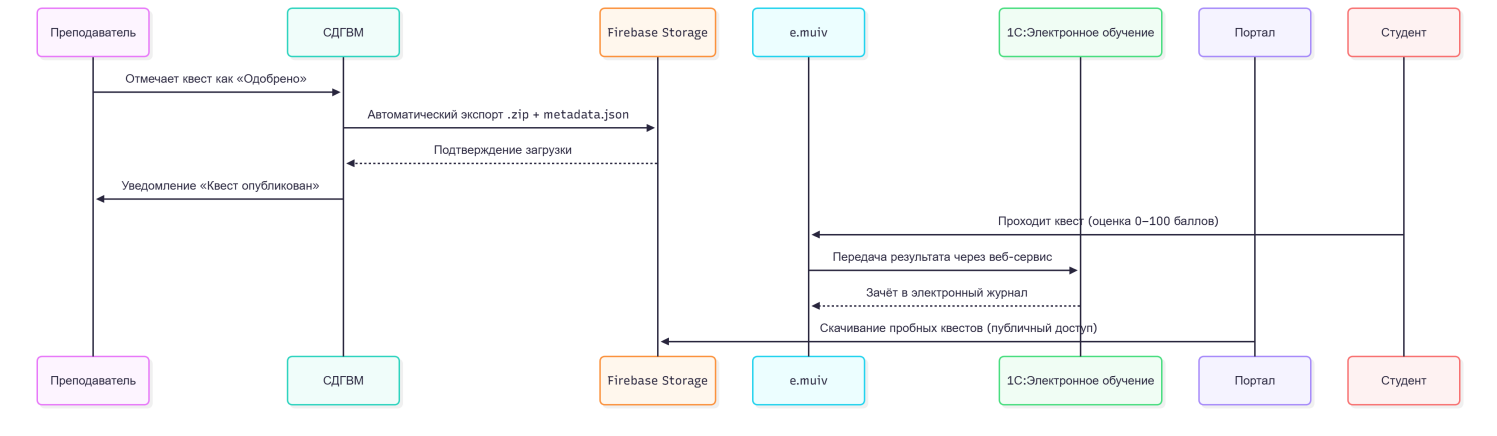


Рисунок 3.3. Диаграмма последовательности для взаимодействующих систем.

Интеграция нужна для реализации цели стартапа “Инструментарий создания культруно - образовательных квестов” - повышение уровня толерантности к Российской национальной культуре, для иностранных студентов. Firebase Storage используется в качестве первоночального варианта хранилища, дальше планируется использование собственного хранилища данных на MinIO “VPS”, при достаточном выделении средств.

**3.4 Формирование модели обновлений и технической поддержки**

Для обновления системы динамической генерации виртуальных миров Разработана модель на основе семантического использования разных типов версий для внедрения: в первую очередь патчи, изменения официальных библиотек LLM-Unity включая скрипты LLMCharacter, для увеличения совместимости с специализированными задачами, версии СДГВМ для внедрения разделяются на мажорные с несовместимыми изменениями и минорные имеющий новый функционал.

Таблица 3.4. План реализации семантической системы обновлений.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Версия | Сроки выпуска | Основные изменения | Тип обновления |
| 1 | 1.1.0 | Июнь 2026 | Прогресс-бар, улучшение диалога с NPC | Минорное |
| 2 | 1.2.0 | Сентябрь 2026 | Интеграция с e.muiv, загрузка квестов с облачного хранилища на сайт e.muiv, для использования квестов в образовательной программе | Минорное |
| 3 | 2.0.0 | Декабрь 2026 | Клиент-серверная архитектура, облачное хранилище, веб-клиент | Мажорное |
| 4 | 2.1.0 | Март 2027 | Подключение к 1С для автоматизации отчётности, проверки корректности | Минорное |
| 5 | 3.0.0 | Декабрь 2027 | Сайт с СДГВМ, возможно мультиплеерные квесты, для группового прохождения,  Лаунчер позволяющий скачивать обновления, поддержка VR | Мажорное |

Для обновления системы динамической генерации виртуальных миров будет использован Git Hub репозиторий, или интегрировать облачное хранилище, для специально разработанного лаунчера - позволяющего скачать установщик exe для обновления или обновлять через установку новых файлов напрямую без установщика через лаунчер, обязательно создаётся резервная копия старой версии, для быстрого возврата при наличии ошибок и несовместимостей, мажорные и минорные версии могут содержать баги и ошибки, поэтому при ошибках должен происходить автоматический или ручной откат к предыдущим версиям.

Для технической поддержки СДГВМ для пользователей продуманы различные пути решения возникающих проблем: Руководства администратора и пользователя, Telegram бот для обращений: t.me/WonkFarmbotHelper, обучающие презентации находящиеся в папке Unity проекта СДГВМ, создание специальной почты только под сообщения о просьбы технической поддержке со стороны пользователей СДГВМ.

Таблица 3.4.2. - соглашение об уровне обслуживания системы динамической генерации виртуальных миров.

|  |  |  |  |
| --- | --- | --- | --- |
| Тип обращения | Время реакции | Время решения | Канал |
| Критическая ошибка | 1 час | 24 часа | Telegram |
| Значительная ошибка | 4 часа | 5 рабочих дней | Telegram или почта |
| Консультация | 24 часа | - | Почта |

Смоделированная система поддержки и обновлений гарантирует эволюцию СДГВМ на протяжении жизненного цикла программного обеспечения.

**3.5 Руководства администратора и пользователя**

Основываясь на ГОСТ 34.602-89 и РД 50-34.698-90 основных требований к руководству пользователям и техническому заданию разработана инструкция по эксплуатации и установки системы динамической генерации, в приложение 3 вынесено руководство администратора, в приложение 4 руководство пользователя.

**3.6 Выводы по третьей главе**

Проведены комплексные тестирования по производительности, удобству использования, методу чёрного ящика, адаптивности GUI, на компьютерах университета генерация происходит за большее время, чем планировалось изначально, разработан план развёртывания на компьютерах по методу Blue-Green, продуман дальнейший план интеграции стартапа в среду МУИВ, используя квесты в курсах и образовательных планах, спланирован жизненный цикл и эволюция системы динамической генерации виртуальных миров, включающий систему семантического контроля версий с разделением на мажорные версии ломающие совместимость и минорные версии содержащие новый функционал, воссоздано руководство для пользователей и администратора, техническое задание, приложение с кодом основного функционала программы. Главное благодаря разработанному инструментарию был реализован стартап конструктор квестов по национальной культуре Российской Федерации, доказавший эффективность и значимость - доказано при помощи проходения иностранными студентами психологических квестов, результаты показывают рост уровня социально - культурной адаптации увеличивая повышение успеваемости и ускоряя прирост связей в общении с русскими студентами.

Экономическая целесообразность внедрения СДГВМ в Московский университет имени С.Ю. Витте подтверждается расчётами прибыли основанными на реальных данных полученных в ходе исследования рынка и основных возможностей монетизации СДГВМ.

Расчёт годовой выручки от продажи SaaS подписок на СДГВМ:

= 187200 рублей, где

является количество активных студентов, цена подписки за год, общая часть платящих клиентов.

Выручка от продажи B2B лицензий вузам:

= 120000 рублей, где - общее число вузов покупающих лицензию на СДГВМ, стоимость одной проданной лицензии (цена за лицензию).

Прибыль от рекламы:

= 43200 рублей, переменные в формуле означают:

это показы одной тысячи реклам, V - Просмотры выбранного для размещения рекламы сайта, в среднем за месяц;

Общая годовая выручка:

= 350400 рублей, где R выручка с продажи SaaS подписок, выручка от продажи лицензий вузам, - прибыль от рекламы;

Маржинальная прибыль расчитывается:

= 262800 рублей, где является долей себестоимости;

Прибыль от использования внедрённого проекта СДГВМ составит около 350400 рублей.

Целевая аудитория:

, где - затраты на рекламу;

рублей - затраты на запуск стартапа;

готовые эксплутационные расходы;

Постоянная прибыль в год t рассчитывается по формуле “Денежный конвеер”:

;

является ставкой дисконтирования;

Чистая приведённая стоимость за пять лет:

;

Индекс доходности находиться по формуле:

;

Внутреняя норма дохода составляет: ;

Модификация для IRR расчитывается по формуле:

;

Дисконтированный срок окупаемости составил ;

Рассчитать за 5 лет коэффициент от возврата инвестиций можно по формуле:

;

Чистый дисконтированный доход за пять лет ищется по формуле:

Доход за пять лет будет около миллиона рублей.

При детальном анализе возможных рекламных каналов, выявлена основная целевая аудитория: кураторы по международному общению в вузах, разработчики игрового контента в стиле визуальных новелл и квестов для студентов, пользователи конечного контента в основном студенты должны, быть заинтересованы в прохождении обучающих квестов, тогда вузы по запросам иностранных студентов.

# **4 ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ПРОЕКТА “СДГВМ” КАК СТАРТАПА**

**4.1 Обзор рынка генерации контента для создания квестов и позиционирование “СДГВМ”**

Рынок генеративной индустрии в 2026 году продолжает расширятся параллельно тренду искусственного интеллекта, глобальная задача цифровизации образования, включающая: автоматизацию процессов создания контентов, для образовательных квестов и программ обучения студентов, адаптирующих под особенности культурно-национального контекста локально Московского университета имени Сергея Юльевича Витте, так и под контексты социумов, других университетов и глобально Российской Федерации в целом. Экспоненциальный рост рынка генерации контента в сегменте игр, развивается, по причине повышения реиграбельности игровых проектов и снижению стоимости разработки игровой продукции при помощи генерации контента. СДГВМ относится к сегменту генерации игровых ассетов, ссылаясь на официальные источники - Industry Research, анализ рынка показал общую стоимость 1,71 миллиардов долларов, растущую в геометрической прогрессии по формуле , где - объём рынка в год - прогнозный размер; начальный объём рынка в 2026 году 1942,7 миллионов долларов, по Industry Researchs. r - годовой темп роста, t - количество лет после базового года 2026. Для 2030 года стоимость рынка по формуле будет составлять, V = 4 386,4 \* 1,3119 ≈ 5 757,1 миллионов долларов, рост стоимости рынка произойдёт в 2,96 раз.

Таблица 4.1 Прогноз по формуле на 2026-2035 года.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Год | Добавленное количество лет к 2026 | Коэффициент роста (1 + r)^t | Объём рынка, миллионов долларов | Округлено, в миллиардах долларов |
| 2026 | 0 | 1,0000 | 1941,8 | 1,94 |
| 2027 | 1 | 1,3119 | 2547,5 | 2,55 |
| 2028 | 2 | 1,7208 | 3342,0 | 3,34 |
| 2029 | 3 | 2,257 | 4386,4 | 4,39 |
| 2030 | 4 | 2,960 | 5757,1 | 5,76 |
| 2035 | 9 | 16,728 | 32483,3 | 32,48 |

Таблица рассчитана по формуле , для разных годов, стоимость рынка генерации игрового контента, растёт по причине развития тренда искусственного интеллекта, и применения в образовательном процессе студентов контента сгенерированного подобными СДГВМ программами и адаптации к культурному контексту, разных национальности, для увеличения уровня успеваемости и социальных взаимодействий в университетах. Также в причины роста сегмента генерации контента игровой индустрии, в удешевлении новых моделей искусственного интеллекта, по причине цифровизации образования повышается спрос на адаптивные квесты и культурно - специфичный контент, вузы и инди-студии используют инструменты генерации с большей вероятностью, для ускорения и удешевления процессов разработки новых квестов, тренд на локальные решения по типу Cursor, Windsurf, Continue, Codium растёт по причинам отсутсвия лишних уязвимостей, для конфиденциальности кооперативных систем, на предприятиях, СДГВМ работает оффлайн аналогично популярным решениям, загруженные локально модели, позволяют локально запускать сервис по генерации контента. Далее приведены аналоги СДГВМ использующиеся, непосредственно в образовательном процессе.

Таблица 4.1.1 Сравнительный анализ инструментов генерации квестов и нарратива.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Инструмент | Тип контента | Локальный или нет | Поддержка на русском | Цена | Ограничения | Конкуренция с СДГВМ (степень) |
| Inworld AI | NPC-диалоги, поведение | Облако | Слабая | 20-200 долларов в месяц | Дорого, цензура, интернет требует VPN | Высокая |
| Charisma.ai | Интерактивные истории | Облако | Нет | 99 долларов в месяц | Enterprise-фокус, дорого | Высокая |
| Scenario | 2D/3D - ассеты | Облако + API | Слабая | 29-149 долларов в месяц | Нет-нарратива | Средняя ассеты |
| Ludo.ai | Идеи, квесты, прототипы | Облако | Нет | Бесплатный тариф с ограничениями +20-100 долларов в месц | Слабый русский, облако | Высокая |
| NovelAI | Текстовые истории | Локально и облако | Средняя | $10-25 в месяц | Нет Unity интеграции, нет визуала | Средняя |
| Rosebud AI | 2D-игры, персонажи | Облако | Слабая | Бесплатная | Ограниченная кастомизация | Средняя |
| СДГВМ | Квесты, диалоги, иконки | Оффлайн | Под российскую национальную культуру | Монетизация с прибылью для МУИВ | Время генерации на слабом ПК | Это СДГВМ |

При проведении тщательного анализа конкурентов в сегменте рынка игровой генерации, видно что СДГВМ занимает уникальную нишу, как интегрированный инструмент для генерации адаптивного контента, использующегося в образовательных квестах, повышающих социальную толерантность иностранных студентов к культуре Московского университета имени Сергея Юльевича Витте, и культурным особенностям Российской Федерации, с множеством национальностей и разнообразного сказочного фольклора. СДГВМ является уникальным оффлайн инструментом генрации контента, для генерации адаптивного контента, использующегося в образовательных квестах, адаптирующих студентов под социум МУИВ и других вухов “при ручной настройке контекстуальных особенностей”, для увеличения успеваемости, и уровень социальных взаимодействий, приоритет отдан культурной адаптации, под национальный контекст. Выбор платформы Unity, основывается на интеграции с образовательным процессом МУИВ по направлению бизнес информатика, специальность: игровая компьютерная индустрия, Unity позволяет сделать низкий порог входа, для освоения, существующие экономические системы позволяют максимально коммерциализировать проект сделанный на платформе Unity, через возможные площадки и Unity Asset Store - позволяющий продовать инструментарий, для генерации в виде отдельного пакета Unity.

Таблица 4.1.2 TAM-SAM-SOM для проекта “СДГВМ”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Показатель | Описание | Объём рынка в млн. долларов | Доля занимаемая “СДГВМ” | Обоснование |
| TAM | Весь глобальный рынок инструментов ИИ, для генерации ассетов | 2000 - 3000 | - | Industry Research, Grand View + MRFR |
| SAM | Доступный рынок: оффлайн решения, связанные с образованием | 250 - 600 | - | 17% от TAM, санкции и фокус на нишевый сегмент |
| SOM | Реальная доля СДГВМ в первые пять лет | 0,5 - 3,0 | 0,1-0,5 % от SAM | Ниша культурной адаптации, вузы РФ “B2B”, Unity Asset Store |

Выводы следующие - подобный СДГВМ пакет в Unity Asset Store, зарабатывает 5 тысяч долларов в месяц, имея стоимость 20 долларов, вузы могут покупать лицензию за 750 долларов в год, востребованная ниша - адаптации иностранных студентов пустая, но востребованная по причине притока большого числа иностранных студентов в Российскую Федерацию около трёхсот тысяч студентов в год.

Таблица 4.1.3 Целевые сегменты клиентов и экономическая ценность.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Сегмент | Размер | Потребности | Ценность СДГВМ в доходе | Ожидаемая цена |
| Соло и инди разработчики | Около 49 753 активных клиентов | Дорого нанять художников, сценаристов, сокращенное время на прототипы | Экономия 500 - 5000 человек-часов экономия 300 тысяч рублей на проект, | 990 - 4990 рублей разово, или 490 рублей месячная подписка |
| Малые студии (2 - 10 человек) | 5 - 8 тысяч в РФ/СНГ | Нужна быстрая итерация квестов/ассетов, ограниченный бюджет | Сокращение цикла разработки на 40%, быстрая монетизация проектов | 1990 - 9990 рублей в виде разовой выплаты + покупка модулей по отдельностей |
| Вузы кафедры, геймдев, психология, культурология | 300 вузов в РФ и 100 вузов в СНГ | Нужно создавать сотни адаптивных квестов ежегодно, ручная работа методистов | Экономия 1-5 миллионов рублей в год, на подрядчиках и методистах | B2B - лицензия 50 тысяч рублей в год |
| Зарубежные вузы НКО (адаптация мигрантов) | Нишевый, 50 - 200 учреждений | Культурно-спецефичный контент на русском; оффлайн-решения | Уникальность ниши порождает премиум цену | 500 долларов лицензия |

Из анализа видно СДГВМ занимает уникальную нишу в сегменте клиентов, что повышает стоимость разрабатываемого проекта, для продажи и монетизации.

4.1.4 Конкурентный анализ с экономическим акцентом.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Инструмент | Цена | Оффлайн | Культура | Интеграция | Аудитория | Слабые  места |
| Inworld AI | 200 долларов в месяц | Нет | Слабая | Есть | Средние AAA студии | VPN, цензура, нет культуры |
| Charisma.ai | 99 долларов в месяц | Нет | Нет | API | Нарративные проекты | Дорого, не подходит под образование, слабая кастомизация |
| Scenario | 149 долларов в месяц | Частично | Слабая | API | Ассеты | Нет нарратива, для квестов |
| NovelAI | 25 долларов в месяц | Да | Средний | Нет | Текст | Нет возможности интегрировать в Unity |
| Ludo.ai | 100 долларов в месяц | Нет | Слабый | Нет | Идеи и прототипы | Плохая оптимизация под русский язык, нет оффлайн |
| СДГВМ | 490 рублей в месяц | Да | Отличный | Полная | Инди + вузы РФ/СНГ | Время генерации на слабом железе |

СДГВМ, наиболее подходит, для решения локальной проблемы Московского университета имени Витте, для создания образовательных квестов, по российской национальной культуре, СДГВМ опередил конкурентов, в ценообразовании и адаптации под контекст образования в российских вузах, благодаря интеграции, под Unity и совместимости с кастомизацией системы, под российский фольклор и язык. СДГВМ позиционируется в качестве доступного инструментария, для генерации адаптивных образовательных ассетов, в виде квестов, иконок и диалогов, СДГВМ ориентирован на нишу в которой, не доступны облачные аналоги, по причине санкций, в нишу входят Российские вузы, нуждающиеся в социально адаптивных генерациях, для образования и привязки иностранных студентов к университетской культуре и социуму, для увеличения количества социальных взаимодействий и успеваемости студента.

Таблица 4.1.5 Юнит-экономика.

|  |  |  |  |
| --- | --- | --- | --- |
| Метрика | Инди-разрабочик “подписка, разовая покупка” | B2B вуз, кафедра “годовая лицензия” | Источник значений и обоснование |
| ARPU “средний доход от одного пользователя” в год | 1200 рублей - разовая покупка, подписка 2490 рублей в год | 90 000 рублей | Общая выручка за период/ количество платящих пользователей |
| CAC “привлечение клиента” | 500 рублей один клиент | 25 000 рублей | Суммарные маркетинговые затраты/ количество привлеченных клиентов |
| LTV “жизненная ценность клиента” за три года | 3000 рублей | 250 000 рублей |  |
| Пожизненная ценность клиента LTV / CAC | 3,0 - 8,5x | 5,0 - 12,0x |  |
| Валовая рентабельность | 88-96% | 82% | (Выручка - прямые затраты/выручка) \* 100% |
| Коэффициент оттока клиентов в год | 37% | 25% | Ушедшие клиенты/ клиенты за начало периода |
| Срок окупаемости | 5 месяцев | 6 месяцев |  |
| Коэффициент дополнительной выручки, на каждый рубль | 0,9 | 0,7 | ROMI = Выручка от маркетинга - затраты на маркетинг/ Затраты на маркетинг |
| Правило “маржа прибыли больше 40%” | 47% | 51% |  |

Основные показатели юнит-эконимики приведенные в таблице расчитываются по формулам: Коэффициент дополнительной выручки: ; Срок окупаемости: , где CAC - стоимость привлечения одного клиента, ARPU - средний доход от одного пользователя в год , Churn - отток клиентов, за период: ; Валовая маржа: , Пожизненная ценность клиента , где LTV “жизненная ценность клиента за период” , CAC “привлечение клиента” . По таблице юнит экономики можно сделать следующие выводы: инструментарий СДГВМ наиболее подходит, для решения локальной задачи помощи иностранным студентам в развитии навыков саморегулирования, ответственности, сгенерированный контент обладает качествами, для осуществления моральной поддержки иностранных студентов и адаптации под социум Московского университета имени Витте, также в отличие от облачных аналогов, СДГВМ развёртывается полностью локально, оффлайн решение сделанное под контекст социума университета, у инструментария уникальная ниша в виде доступного генератора ассетов, для квестов, использующихся для повышения социальной адаптации, развития ответственности и саморегулирования у иностранных студентов, также квесты повышают уровень межкультурных взаимодействий и общую успеваемость и индивидуальный рейтинг в Московском университете имени Сергея Юльевича Витте. При анализе возможностей СДГВМ в выбранной рыночной нише, юнит-экономика демонстрирует высокую рентабельность проекта: пожизненная ценность клиента, в диапазоне от 5 до 12, срок окупаемости 6 месяцев, следовательно у проекта быстрая самоокупаемость маркетинговых вложений, валовая маржа 82% по причине нулевых затрат на копирование, целесообразность позиционирования “СДГВМ” как доступного инструментария, для генерации контента, использующегося в условии санкций, для создания образовательных квестов, повышающих уровень саморегулирования, ответственности, толерантности к чужой культуре и повышающий уровень социальной поддержки иностранных студентов.

**4.2 Разработка бизнес модели проекта**

СДГВМ разрабатывается в качестве стартапа, в нише инструментария генерации адаптивного контента, использующегося для образовательных квестов, специфика конечного продукта предполагает оффлайн использование, с упором на культуру российской федерации, имеет низкий порог входа, готов к интеграции в образовательный процесс в виде практик для специализации “Игровая компьютерная индустрия” и возможность исследования применения искуственного интеллекта в образовании и социальной адаптации для педагогических направлений в Московском университете именни Сергея Юльевича Витте, изначально система разрабатывалась для решения локальной проблемы МУИВ, с адаптацией новых студентов, к особенностями культуры университета и ознакомлением с биографией Сергея Юльевича Витте, но в дальнейшем при анализе ниши рынка генерации, выяснилась уникальность продукта СДГВМ, в контексте интеграции с культурно-национальным контекстом Российской Федерации, проект востребован в нише СНГ, России, целевой аудитории университетов и кафедр - около трехсот, также соло или инди разработчики нуждаются в подобных унифицированных инструментариях подобных СДГВМ, лучшим выбором экономической модели будет - гибридная бизнес модель, включающая коммерциализацию по возможным путям реализации монетизации и продажу дополнительного функционала или возможностей. Vip функции, для B2B лицензий, ежемесячные подписки, для отдельных лиц в виде разработчиков, годовые подписки, для институтов и кафедр. Гибридная модель выбрана по причине: быстрого набора целевой аудитории потребителей СДГВМ, устойчивости в доходе от СДГВМ при использовании гибридной бизнес модели, “бесплатный доступ к базовым функциям привлечет больше потенциальных заказчиков”, в условиях роста рынка ИИ инструментария генерации контента, для геймдева растёт, потребность в автоматизации процессов создания учебных материалов, для образовательной программы с учетом роста тренда ИИ и цифровой трансформации образовательных процессов, успешным будет продвижение инструментария генерации адаптивного контента СДГВМ, через Unity Asset Store, что свойственно для гибридной бизнес модели, если брать в условие санкции, то оффлайн режим генерации, имеет преимущество, над облачными решениями ограничеными для использования в регионах СНГ и России.



Рисунок 4.2.1 Классический шаблон бизнес модели Остервальда.

Таблица 4.2.1 Ключевые метрики успеха и этапы монетизации.

|  |  |  |  |
| --- | --- | --- | --- |
| Этап 2026-2028 | Цели по пользователям | Выручка в рублях | Основные действия |
| 2026 запуск | 500 скачиваний через Asset Store + 5 вузов пилотов | 100 тысяч рублей | Релиз бесплатного базового функционала, пилот в МУИВ |
| 2027 рост | 1500 скачиваний, 50 активных лицензий B2B | 220 тысяч рублей | Обновления 2.0, интеграция с Moodle/1C |
| 2028 масштаб | 3000 скачиваний, 150 активных лицензий | 400 тысяч рублей | Мажорная версия с облаком и экспортом в СНГ |

Благодаря уникальности ниши СДГВМ обладает рядом преимуществ: низкой стоимостью привлечения клиентов, за счёт использования Unity Asset Store, полная защита от санкций, благодаря использованию локально загруженных моделей, без использования API и облаков ограниченных санкциями, нулевые затраты на R&D на старте, за счёт синергии с дипломом. Также существуют значительные минусы за счёт уникальности проекта: долгая генерация на не производительных компьютерах, по причине использования локальных вычислительных ресурсов - требуется постоянная оптимизация, высокая конкуренция от бесплатных вариаций локальных ресурсов в виде нейросетей использующихся, для генерации контента, но “СДГВМ имеет ориентацию на русский фольклёр и культурную составляющую Российской Федерации”, также низкая цена на старте проекта, возможные скидки, для привлечения новых клиентов, существенный минус проекта.

Гибридная бизнес модель использует комбинацию актуальных направлений коммерциализации проекта, с опорой на устойчивое развитие, при позиционировании на российском рынке.

**4.3 Модель коммерциализации “СДГВМ” и ценообразование в Unity Asset Store**

Выбрана ведущая глобкльная площадка в области монеизации дополнительных пакетов Unity, для создания прототипов и полноценных игровых приложений, на платформе уже размещено около тринадцати тысяч работ, использующихся десятью миллионами разработчиков, ежемесячно с платформы скачивают 5 миллионов ассетов и пакетов, а лучшие работы зарабатывают по пятьдесят тысяч долларов в месяц. В преимущества Unity Asset Storre входит: низкаий порог входа, в виде бесплатной публикации проекта, основные пользователи совпадают с потенциально рассматриваемыми потребительскими сегментами “СДГВМ”, рейтинги и рекомендации включенные в платформу, позволяют продвигать СДГВМ и выводить в топ по ключевым тэгам: культурные ассеты, генератор квестов, диалогов, икон, оффлайн генератор. Unity Asset Store позволит продовать дополнения к СДГВМ и премиум версии, с улучшенным функционалом, также Unity Asset Store доступен в России и можно оплачивать, через карты МИР и VK Pay, полностью обходит ограничения по облачным сервисам.

В коммерциализации СДГВМ используются дополнительные каналы продвижения: прямые продажи университетам, сообщества в социальных сетях, сайт с демо версией СДГВМ, для лидогенерации по продажи лицензий.

Таблица 4.3 Список версий СДГВМ и структура пакетов.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Версия | Описание | Целевая аудитория | Цена в рублях на старте | Дополнительно |
| Бесплатная с базовым функционалом | Базовая генерация ограничена пятью попытками в день | Тестировщики, инди разработчики, соло разработчики на пробу | Бесплатно | Ознакомительная версия, для привлечения лидов. |
| Стандартный функционал | Неограниченная базовая генерация по попыткам во время действия купленной лицензии | Инди - разработчики, соло-девы | 2 490 рублей | Основной продукт |
| Pro | + Улучшенное качество генерации, большая оптимизация, при генерации благодаря использованию графического ускорения и оптимизации работы процессора | Продвинутые инди разработчики, небольшие студии | 4 990 рублей разово | + DLC - модули |
| Education и B2B лицензия | Неограниченное использование на кафедрах, техподдержка, кастомизация под вуз, использование датасетов русской культуры | Вузы, кафедры “геймдева, психологии, культурологии” | 99 000 - 299 000 рублей в год | Прямые продажи |
| DLC модули | - LoRA - модели под стилистику конкретного университета, датасеты квестов под выбранную стилистику, расширенные модели NPC поведения (подстроенные и обученные под требования заказчика) | Продвинутые студии, вузы, кафедры, соло разработчики, инди разработчики. | 2 990 рублей за модуль | Дополнительный доход |

Средняя цена на инструментарии генерации контента, подобного СДГВМ составляли от восьмиста до девяти тысяч рублей, за разовую покупку B2B лицензии.

Таблица 4.3.1 Прогноз дохода от разных каналов монетизации.

|  |  |  |  |
| --- | --- | --- | --- |
| Канал | 2026 | 2027 | 2028 |
| Unity Asset Store | 250 000 рублей | 2 500 000 рублей | 4 000 000 рублей |
| Подписки | 50 000 рублей | 1 000 000 рублей | 2 000 000 рублей |
| B2B - лицензии вузам | 100 000 тысяч рублей | 1 500 000 рублей | 3 000 000 рублей |
| Итого | 400 000 рублей | 2 000 000 рублей | 9 000 000 рублей |

Диплом как стартап СДГВМ, имеющий сначала ценность, только для Московского университета имени Сергея Юльевича Витте, оказался монетизируемым, для целой ниши клиентов, на рынке геймдева и генерации квестового контента при помощи искусственного интеллекта, ценность проекта, будет повышаться с течением эволюции, развития и проведения новых исследований в области генерации контента искусственным интеллектом. SEO - оптимизация, для веб страницы Unity Asset Store, должны включать минимум семь - восемь примеров генерации визуалов, скриншоты проходящей генерации, отзовы с пилотного тестирования программы, обязательно должны быть ежемесячные логи разработчиков, с информацией над чем ведётся работа, что получается, что нет, планируемые DLC, выход новых дополнений привлечёт больше потенциальных клиентов и удержит старых, посты в группах ВК и других социальных сетях, также повлияют положительно на динамику SEO продвижения СДГВМ в Unity Asset Store, в постах можно размещать промо - акции, скидки, так привлечение пойдёт быстрее. Реальные значения органического трафика, можно посчитать по формуле: ,при поставления ,CTR(p) - ожидаемый процент кликов по ссылке, (p) - позиция в выдаче, a, b, c, d - параметры нелинейной регрессионной кривой “эмпирические значения полученные путём адаптации модели к данным, на больших выборках из датасетов кликов”,далее объём органического трафика расчитывается по формуле: , где - ежемесячный поисковый объём по i-му ключевому слову, - прогнозируемая позиция по слову, CTR() - коэффициент кликабельности с позиции , - доля трафика, которая приходит на русскоязычный сегмент константа = 0,7. Стоимость привлечения одного посетителя при использовании SEO: , - все затраты на SEO за период, где - все затраты на SEO за период, - органический трафик за тот же период, CR - конверсия из посетителя в покупателя.Показатель жизненной ценности клиента расчитывается по формуле: , где ARPU - средний чек,k - коэффициент допродаж DLC, r - возврат. Прогноз роста органического трафика вычисляется по формуле: , где - начальный трафик, g - месячный темп роста, t - время в месяцах, K - предельный трафик, r - скорость роста.

Таблица 4.3.2 Прогноз органического трафика по ключевым словам.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Ключевое слово | Поиск | Прогноз | CTR | Ожидаемый трафик | Конверсия в покупку | Новые клиенты |
| Генератор квестов unity | 1100 - 1500 руб | 4-6 | 7,5-9,0 | 90-130 | 1,2 - 2,0 | 1-3 |
| Offline ai quest | 300-600 руб | 5-8 | 4,0-6,5 | 20-40 | 1,5 | 0-1 |
| Генератор диалогов npc | 800-1200 руб | 5-7 | 5,0-7,0 | 50-80 | 1,8 | 1-2 |
| Ai rpg icons generator | 400 - 700 руб | 6 - 9 | 3,5-5,5 | 20 - 40 | 1,0 | 0-1 |
| Генератор иконок unity ai | 500-900 руб | 4-7 | 6,0 - 8,5 | 40-70 | 1,5 | 1-2 |
| Итого по ключам | ~3100 - 5000 руб | - | - | 220-360 | - | 3-9 |

**4.4 Прогноз объёма рынка подбор целевого сегмента под “СДГВМ”**

Прогноз объёмов рынка инструментов генерации игрового контента, для образовательных квестов, включая генерацию 2D и 3D контента, адаптивных диалогов NPC, нарратива, а также направленность на конкретный контекст адаптации, под среду Московского университета имени Сергея Юльевича Витте, можно оказать точные цифры стоимости рассматриваемой рыночной ниши, для коммерциализации проекта, составляет 1 942 731 182 доллара США, по оценкам официальных источников Technavio “Generative AI in Gaming Market Growth Analisis - Size and Forecast 2025-2029 [Электронный ресурс] // Technavio. - URL: <https://www.technavio.com/report/generative-ai-in-gaming-market-industry-analysis> указал близкое значение к ранее показаному числу объёма рынка 1,94 миллиарда долларов, AI Game Generator Market Size [Электронный ресурс] // Market.us. - 2025. - URL: <https://market.us/report/ai-game-generator-market/> (дата обращения 2026), при взятии среднегодового темпа роста выбранного сегмента рынка, получается 31%, по предоставленным выше источникам. Расчитать формулу экспоненциального роста: , где: = 1942 миллионов долларов стоимость сегмента рынка “СДГВМ”, r = 0,28 \* 100 = 28%, t - период времени после 2026 года.

Таблица 4.4 Общий объём целевого рынка СДГВМ.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Год | t | Коэффициент  (1 + 0,28)^t | Объём рынка в миллионах долларов | Округлённо в миллиардах долларов |
| 2026 | 0 | 1,000 | 1942 | 1,94 |
| 2027 | 1 | 1,280 | 2486 | 2,49 |
| 2028 | 2 | 1,638 | 3182 | 3,18 |
| 2029 | 3 | 2,097 | 4073 | 4,07 |
| 2030 | 4 | 2,684 | 5213 | 5,21 |
| 2035 | 5 | 8,916 | 17320 | 17,32 |

По результатам анализа, выявлена классическая экспоненциальная положительная динамика стоимости рынка, что связано с ростом тренда на искусственный интеллект, коэффициент роста увеличивается в два раза, спустя период в три года, к 2035 году, увеличится в девять раз, стоимость рынка будет составлять 17,32 миллиарда долларов. При захвате скромной доли рынка в 0,00001% СДГВМ принесёт около 1732 доллара в год - пессимистичный сценарий, при базовом сценарии прибыль от СДГВМ в год будет в три раза выше, чем 1732 доллара, объём рынка в 2026 году 1,94 миллиарда долларов, следовательно пессимистичный сценарий предполагает 900 долларов в год, честные цифры выявленные при помощи обработки информации из реальных источников.

Таблица 4.4.1 Сравнение прогнозов TAM “общий объём целевого рынка” по разным источникам.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Источник | Отчёт | Увеличение объёма рынка в (%) | Объём на 2026 год в млрд. долларов | Объём на 2030 год, млрд долларов | Объём на 2035 году, млрд. долларов | Примечание |
| Technavio | Generative AI in Gaming, 2025 | 39,2 | ~1,94 | ~7,85 | - | 2025-2029 |
| Market.us | AI Game Generator Market, 2025 | 29,2 | 2,0-2,5 | ~10-12 | ~21,26 | [2] |
| Grand View Research | AI in Gaming Market, 2025 | 36,1 | ~3,28 | - | 51,26 | [3] |
| Собственный расчёт | - | 28,0 | 1,94 | 5,21 | 17,32 | Усреднение источников |

SAM “доступный сервисируемый сегмент рынка”, лучшая часть по специфике СДГВМ, за счёт особенностей по типу: работы генерации, без интернета, культурной адаптации иностранных студентов, контекст национальной культуры Российской Федерации, упор на среду Московского университета имени Сергея Юльевича Витте, для повышения уровня ответственности и самоконтроля студентов, повышения уровня социальной поддержки и адаптации под социум МУИВ. От общего рынка TAM, разобранного ранее, SAM действительно можно охватить и обслужить СДГВМ, с учётом ограничивающих факторов, СДГВМ подходит, для слабо развитых регионов России, за счёт отсутствия интернета, или обеспечения конфиденциальности данных, необходимо использовать защиту конфеденциальных данных, - обычные облачные решения и API, не подходят, нужно использовать СДГВМ - работающую полностью без интернета, также благодаря фокусу на образование и среду Московского университета имени С.Ю.Витте, генерация адаптивных образовательных квестов, и ассетов происходит учитывая национальный фольклор и культуру, университетский социум МУИВ и правила поведения в МУИВ, адаптацию иностранных студентов за счёт повышения самоконтроля и ответственности студентов, социальной поддержки иностранных студентов, за счёт сгенерированного контента при помощи СДГВМ, использующегося в образовательных квестах.

В результате получается что СДГВМ в искусственном интеллекте, для геймдева и EdTech, займёт в нише контекста выше перечисленных особенностей 20%, от всего EdTech в поднише генератора контента, для образовательных квестов Российской Федерации, по аналитике статей AI in Gaming Market Size, Share & Trends Analysis Report By Component (Hardware, Software), By Application “PC Gaming, Console Gaming, Mobile Gaming”, By Region, And Segment Forecasts, - 2025. - URL:

<https://www.grandviewresearch.com/industry-analysis/ai-gaming-market-report>

Дата обращения 18.01.2026, в статье показан совокупный среднегодовой темп роста выбранного сегмента рынка на 36,1% предсказана стоимость сегмента искуственного интеллекта в генерации игрового контента на 2033 год - 3,28 миллиардов долларов.

Таблица 4.4.2. Фильтры и коэффициенты для расчёта SAM от TAM.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Год | SAM, млн долларов | SOM пессемистичный | SOM базовый | SOM оптимистичный | Доля от SAM |
| 2026 | 233-350 | 0,01-0,03 | 0,05-0,18 | 0.23-1,05 | 0,02-0,05 |
| 2027 | 298-450 | 0,015-0,045 | 0,06-0,23 | 0,30-1,35 | 0,02-0,05 |
| 2028 | 382-580 | 0,019-0,058 | 0,08-0,29 | 0,38-1,74 | 0,02-0,05 |
| 2029 | 489-740 | 0,024-0,074 | 0,10-0,37 | 0,49-2,22 | 0,02-0,05 |
| 2030 | 625-940 | 0,031-0,094 | 0,13-0,47 | 0,63-2,82 | 0,02-0,05 |
| 2035 | 2080-3120 | 0,10-0,31 | 0,42-1,56 | 2,08-9,36 | 0,02-0,05 |

Видно подтверждение ранее сказанных слов про доступный сервисируемый сегмент рынка и потенциал СДГВМ в коммерциализации, на нише рынка сегмента EdTech, для генерации образовательного контента, развивающего ответственность, самоконтроль и толерантность к культуре Московского университета имени Сергея Юльевиича Витте, лекгость обслуживания СДГВМ и экспоненциальный рост выбранной ниши рынка с 233 миллионов долларов в 2026 году, до 940 миллионов долларов в 2030 году, выявленые плюсы в работе системы генерации без интернета, русскоязычный контекст, адаптация к культуре, социальная поддержка иностранных студентов, адапптация к среде МУИВ, делают выбранную нишу ёмкой, затрагивающую смешанную целевую аудиторию, по гибридной бизнес модели, сегмент смешанной целевой аудитории достаточно ёмкий и расширяется, в проекте инструментария СДГВМ, по генерации контента, есть смысл по причине возможной коммерциализации и получению денежных средств, хоть изначально проект задумывался, для решения локальной проблемы адаптации к социуму Московского института имени Сергея Юльевича Витте, но в дальнейшем, проект показал возможности для коммерциализации в разных формах, продажа в виде пакета, через Unity Asset Store, продажа B2B лицензий, для соло и инди геймдева, разовые продажи кафедрам и университетам России и СНГ “контекст на народный фольклор и адаптации к среде конкретного университета, легко подогнать под любой вуз, нужно просто рассказать СДГВМ об особенностях и контексте выбранного университета”. Даже при пессимистичном сценарии выручка даст ощутимый результат в 2026 году 900 долларов, 2030 году 2800 долларов, для университетского стартапа, достижимо сделать 30 продаж в Unity Asset Store, 3 продажи B2B лицензий. При оптимистичном сценарии, доход выходит уже приличный: при курсе рубля равном 90 рублей, ожидается доход от 1 миллиона рублей до 2,2 миллионов рублей “после 5 лет развития проекта”. Стратегия в базовом сценарии - фокус на узкой нише, такая стратегия позволяет уменьшить затраты, на масштабирование и минимизировать конкуренцию, для сохранения высокой маржинальности и прибыли от стартапа. Ниша СДГВМ, остаётся почти не занятой, за счёт ориентации на культурный контекст, социум МУИВ, адаптацию иностранных студентов - это необычно, для ИИ стартапа EdTech, действительно мало конкуренции, даже при пессимистичном сценарии проект окупается за два года и приносит прибыль Московскому университету имени Сергею Юльевичу Витте. Значимым стратап СДГВМ делает не только оценка дохода, но и социальная ценность - проявляющаяся в увеличения уровня саморегулирования, ответственности, социальной поддержки, успеваемости, толерантности у иностранных студентов к социуму Московского университета имени Сергея Юльевича Витте, таблица 4.4.2 показала рост, выбранной ниши EdTech с упором, на адаптацию иностранных студентов, проект СДГВМ экономически целесообразен и решает локальную проблему Московского университета имени Сергея Юльевича Витте, в социальной адаптации новых иностранных студентов, также возможны исследования на основе применения искуственного интелекта, для генерации контента, исполльзующегося в образовании, для цифровизации образования в МУИВ и повышения конкуренции с другими вузами, в скорости адаптации студентов, что важно, для увеличения рейтинга устойчивого развития университета - “требуется для улучшения оценки экспертами уровня цифровизации и устойчивого развития Московского университета имени Сергея Юльевича Витте - СДГВМ нужен, как продукт, показывающий заинтересованность института в устойчивом развитии и социальной поддержке иностранных студентов”.

**4.5 Расчёт затрат на привлечение целевой аудитории**

Можно разделить целевую аудиторию СДГВМ, на два сегмента: B2B - вузы, кафедры, “кастомизация под контекст университета заказчика” - годовые лицензии с продлением и дополнениями, для соло и инди разработчиков B2C “Unity Asset Store или продажа лицензий с дополнениями”, формула для расчёта затрат на привлечение клиента, будет выглядеть: , ключевой индикатор жизнеспособности , на каждый потраченный доллар на привлечения, приходится 3 доллара дохода полученного от приличенного клиента; эффективность вложений, показатель окупаемости затрат на маркетинг рассчитывается по формуле: , а срок окупаемости рассчитывается по формуле: . Должны использоваться следующие виды каналов привлечения: таргетированная реклама -подходит, под контекст гибридной бизнес модели и выбранной узкой ниши сегмента, ориентирующийся на целевой аудитории, состоящей в основном из кафедр, университетов, соло и инди разработчиков, в большей степени таргетированная реклама затронет инди-разработчиков, для реализации подойдут социальные сети - Telegram Ads, ВКонтакте, Яндекс.Директ.

Для B2B подойдут вебинары, демонстрации в других вузах, холодные письма, основной целевой аудиторией, для прямых продаж будут - вузы стран СНГ и России, а также кафедры относящиеся, к психологии и IT.

Unity Asset Store основной канал коммерциализации, для организации органического трафика, с применением SEO продвижения и отзывов Unity Asset Store, канал применяется, для B2C, также необходимо применять продвинутые методы машинного обучения и применение ансамблевых методов, для мониторинга и управления отзывами клиентов “классификация отзывов, определение определение эмоциональной тональности текста”, для увеличения динамики положительного роста рейтинга СДГВМ, в Unity Asset Store, а также профессиональный подход к монетизации стартапа “применение машинного обучения, способствует увеличения прибыли от стартапа, для университета”. Ансамблевые методы объединяют несколько простых моделей в одну мета модель, для значительного увеличения точности предсказания, классификации комментариев и отзывов, купивших СДГВМ клиентов, являющихся инди-разработчиками, при этом прибыль от стартапа растёт, необходимо показать математическую формулу прироста к прибыли далее: , где: - точность ансамблевой модели, - точность базовой модели, - количество привлечённых клиентов, ARPU - средний доход на клиента, (1 - Churn) - доля удержанных клиентов. При улучшении точности предсказания класса отзывов, увеличивается прибыль на 30 % за счёт возможности реагировать, на изменение тональности отзывов и устранять проблему раньше, персонализации контента и сегментации аудитории на сегменты. Применение ансамблевых методов машинного обучения, особенно важно, по причине использования гибридной бизнес модели, для СДГВМ, организация и управления органическим трафиком важно, для увеличения прибыли, с канала Unity Asset Store, а также использование ансамблей увеличит качество работы с B2B клиентским сегментом включая вузы, кафедры, повысив общее доверие к проекту СДГВМ, машинное обучение является драйвером устойчивого развития стартапа, повышая прибыль университету, для нескольких каналов реализации проекта.

Контент маркетинг будет реализован через Telegram - каналы, группы в ВК, посты в группах по геймдеву, также большая база опубликованных статей, по разработке и улучшению СДГВМ, позволяет распространять сведения о ценности стартапа, и привлекать новых потребителей, для покупки подписки или единоразовой лицензии на СДГВМ.

Для повышения пользовательского интереса к продукту, необходимо выпускать платные DLC, в Unity Asset Store, наличие реферальных программ и скидок - влияет положительно на привлечение новых клиентов, необходимо: проводить акции в социальных сетях, розыгрыши лицензий за участие в конкурсах, выделены следующие подходящих виды конкурсов: “конкурс репостов”, “конкурс по лайкам на комментарий”, “конкурс с голосованием”, “квесты - поиск подсказок в предыдущих постах, выполнение заданий, с ответами на вопросы”, “конкурс на лучшую работу - например человек делает реализацию генерации через ComfyUI, проверяется качество, оптимизация и скорость генерации, победителем оказывается лучший”.

Таблица 4.5.1 Структура затрат на привлечение аудитории.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Канал | Запкуск стартапа (2026 год) | Развитие стартапа 2027 год | Масштабирование 2028 год | Итог за 3 года | Доля в бюджете (%) |
| Органика Unity Asset Store (SEO, скриншоты, отзовы) | 50 000 | 80 000 | 120 000 | 250 000 | 18 |
| Таргетинг ВК, Telegram, Яндекс.Директ | 150 000 | 300 000 | 500 000 | 950 000 | 68 |
| Контент - маркетинг “посты, статьи, видео” | 30 000 | 60 000 | 100 000 | 190 000 | 14 |
| Прямые продажи B2B (письма, звока, вебинары) | 20 000 | 50 000 | 80 000 | 150 000 | 11 |
| Реферальные программы и скидки | 10 000 | 30 000 | 50 000 | 90 000 | 6 |
| Итого затрат на маркетинг | 260 000 | 520 000 | 850 000 | 1 630 000 | 100 |

При анализе таблицы 4.5.1 отчетлива видна динамика роста маркетингового бюджета, опираясь на полный жизненный цикл стартапа СДГВМ, таргетированная реклама забирает большую часть затрат, но необходима, для быстрого привлечения инди и соло разработчиков, прямые продажи кафдрам и университетам обеспечивают стабильный ежемесячный регулярный доход, а Unity Asset Store, требует меньше вложений, по сравнению с остальными каналами.

Таблица 4.5.2 Расчёт САС по сегментам.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Сегмент | Период | Затраты на маркетинг в рублях | Привлечено клиентов | САС рублей | ARPU годовой в рублях | LTV за 3 года в рублях | LTV  CAC  ratio | Payback в месяц |
| Инди - разработчики B2C | 2026 | 200 000 | 400 | 500 | 1 200 - 2 490 | 3 000 - 7500 | 6-15 | 2-5 |
| Инди-разработчики B2C | 2027 | 400 000 | 1000 | 400 | 1 200 - 2 490 | 3000 - 7500 | 7,5-18,75 | 2-4 |
| Вузы, кафедры | 2026 | 60 000 | 3-5 | 12 000 - 20 000 | 90 000 | 180 000 - 250 000 | 9 - 20 | 2 - 3 |
| Вузы кафедры | 2027 | 120 000 | 10-15 | 8 000 - 12 000 | 90 000 | 180 000 - 250 000 | 15 - 31 | 1 - 2 |
| Общий | 2026 | 260 000 | 405 | 642 | - | - | - | - |

Видно, различие стоимостью привлечения сегмента B2B и B2C, цена привлечения B2C ниже, а B2B выше, благодаря таргетингу и органическому трафику, через рейтинги и отзывам на Unity Asset Store, соотношение пожизненной ценности клиента и стоимости привлечения клиента, для двух сегментов превысило порог три к пяти - доход получаемый от привеченных клиентов больше в три и пять раз, чем траты на привлечение этих клиентов, это показатель эффективности гибридной бизнес модели, клиент окупается за период меньший пяти месяцев, СДГВМ как стартап, даёт преимущество в быстром возвращении вложений, что решает задачу реинвестирования прибыли в проект, для улучшения СДГВМ и увеличения коэффициента полезного действия влияния сгенерированного контента, на задачу повышения уровня адаптивности к социуму конкретного университета, быстрое инсвестирование средств в СДГВМ решает главные задачи конструктора квестов: увеличения уровня ответственности и саморегулирования, успеваемости, чувства уверенности, эмоциональной поддержки, толерантности к социуму Московского университета имени Сергея Юльевича Витте.

Таблица 4.5.3 Сценарии маркетингового бюджета и САС.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Сценарий | Бюджет 2026 | Бюджет 2027 | Бюджет 2028 | Итого 3 года | CAC B2С рублей | CAC B2B рублей | LTV, CAC |
| Пессимистичный (Высокий САС) | 400 000 | 800 000 | 1 200 000 | 2 400 000 | 800 - 1000 | 20 000 - 30 000 | 4 - 8 |
| Базовый | 260 000 | 520 000 | 850 000 | 1 630 000 | 400 - 500 | 8 000 - 12 000 | 10 - 20 |
| Оптимистичный (низкий САС) | 150 000 | 300 000 | 500 000 | 950 000 | 200 - 300 | 5 000 - 8 000 | 15 - 30 |

Глядя на три основных сценария развития маркетинга СДГВМ, можно сказать, что даже пессимистичный сценарий обеспечивает жизнеспособность проекта, CAC для B2C увеличен до 1000 рублей, для B2B до 20 000 рублей, но соотношение пожизненной ценности клиента и стоимости привлечения клиента будет четыре к восьми - для B2C и шесть к двенадцати, для B2B, необходимый порог жизнеспособности проекта “соотношение ценности к стоимости привлечения” ≥ 3, значения четыре к восьми и шесть к двенадцати, превышают порог, получается на один затраченный рубль на привлечение получается четыре или восемь рублей, для B2C и шесть или двенадцать рублей, для B2B, получается положительный денежный поток гарантируется, формула подтверждающая расчёты: , подставим в формулу числа - получим:

Также стартап СДГВМ имеет короткий срок окупаемости: , подставляем числа из таблиц: , да срок окупаемости приблизительно равен 5 месяцам, как было сказано ранее. Следовательно, жизнеспособность СДГВМ как стартапа, позволяет развивать проект, даже при пессимистичном сценарии, устойчивость стартапа к внешним рискам, привлекает дальнейшее финансирование, для масштабирования и улучшения проекта, в рамках получения прибыли Московским университетом имени Сергея Юльевича Витте в рамках стартапа.

**4.6 Организационный план и структура затрат для “СДГВМ”**

Распределение ответственностей по ролям, основные этапы жизненного цикла и этапы развития проекта в качестве стартапа - СДГВМ, входят в организационный план и структуру затрат стартапа. Можно выделить ряд отличительных особенностей стартапа СДГВМ: сложный этап запуска, по причине ограниченных ресурсов в виде человеческих затрат и технических возможностей “персональные компьютеры, на которых ведётся разработка без видеокарты”, финансовые ресурсы также ограничены, улучшение системы ведётся на личные средства автора данной дипломной работы, на старте сложна интеграция с кафедрами психологии и IT, для внедрения системы официально в образовательный процесс, привязкой результатов психологического тестирования толерантности к МУИВ в личном кабинете студента, это необходимо, для решения поставленных задач конструктору квестов - адаптация студентов под социальную среду Московского университета имени Сергея Юльевича Витте, интеграция с личным кабинетом студентов проблематична, по причине не достаточной совместимости созданных ресурсов сайта на Moodle, во время разработки сайта участвовали студенты, есть проблематичные места интеграции конструктора квестов с метриками, для оценки студентов, метрики нужны чтобы понять увеличился, ли уровень адаптации студента, или нет, при отрицательных результатов, применяются особые методики воздействия, заставляющих принять правильную модель поведения “толерантная модель”, СДГВМ коммерциализируется по гибридной бизнес-модели, затрагивающей разные каналы монетизации, включающие B2C лицензии используя платформу Unity Asset Store, B2B лицензии, для университетских кафедр в других вузах, также особенностью будет являться акцент на повышение адаптивности иностранных студентов, к правилам поведения в социуме Московского университета имени Сергея Юльевича Витте, влияние на уровень ответственности, эмоциональной поддержки, саморегулирования через проходимые студентами квесты, сделанные на основе контента сгенерированного СДГВМ.

Планируется что этап запуска продлится три года, на этапе “запуска” проект реализуется в формате “университетского стартапа”, рекомендуется следующий набор команды: Научный руководитель, технический консультант, разработчик СДГВМ, психолог для использования методик адаптации студентов, маркетолог, разработчик визуальной новеллы, руководитель проекта. Далее показана таблица ролей, для стартапа.

Таблица 4.6.1 Матрица ролей и ответственности в проекте “СДГВМ”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Роль | Основные обязанности | Ключевые задачи в проекте | Связь с задачами адаптивности студентов | Уровень вовлеченности |
| Научный руководитель | Научное руководство, контроль соответсвия целям университета, утверждение дорожной карты | Утверждение концепции квестов, контролирует научную новизну, помогает в реализации статей | Обеспечивает методологическую основу, для увеличения навыков саморегулирования, ответсвенности, толерантности у иностранных студентов | Постоянно |
| Технический консультант | Консультация по архитектуре и интеграции моделей ONNX, LMM-Unity, ComfyUI, оптимизация производительности, | Техническое руководство разработкой, выбор моделей, решение технических проблем, оптимизация под слаюое железо | Техническая реализация адаптивных NPC, обесспечивающих эмоциональную поддержку и обратную связь | Высокий уровень вовлеченности ближайшие 5 лет |
| Разработчик СДГВМ | Полная разработка системы, включая код, интеграцию моделей, интерфейс, тестирование, обновление и поддержка после релиза | Реализация всей функциональности: генерация квестов, диалогов, иконок, оффлайн режим | Прямая реализация механик, влияющих на саморегулирование и ответственность через развилки и последствия | Постоянно |
| Психолог | Разработка и валидация психологических методик адаптации, подбор стимулов в квестах, анализ влияния на студентов, интерпритация результатов тестирования | Создание сценариев квестов, направленных на адаптацию, оценка эффективности по методикам | Прямая ответсвенность за эмоциональную поддержку, толерантность, саморегулирование и ответсвенность | Высокий в пилотной фазе |
| Маркетолог | Продвижение в Unity Asset Store, SEO, таргетированная реклама, контент-маркетинг, работа с отзывами, ML-анализ тональности | Организация органического трафика, запуск таргета, акции, конкурсы, мониторинг репутации | Косвенная - чем больше охват, тем больше студентов получт квесты для адаптации | Высокий |
| Разработчик визуальной новеллы | Создание демонстрационных образовательных визуальных новелл на базе контента, сгенерированного СДГВМ, интеграция сгенерированного контента, тестирование на студентах | Реализация готовых квестов в формате визуальной новеллы, проверка связанности нарратива и визуалов | Прямая демонстрация эффекта квестов на эмоциональную поддержку и толерантность к социуму МУИВ | Поддержка |
| Руководитель проекта | Общее управление, координация команды, планирование сроков, контроль бюджета, взаимодейтствие с МУИВ и партнёрами, защита интересов университета | Стратегическое планирование, дорожная карта, отчётность, привлечение ресурсов | Координация всех усилий для достижения адаптивности и социальной миссии университета | Постоянно |

В рамках университетского стартапа, нужно соблюсти порог минимального количества необходимого количества участников, структура команды, основная нагрузка лежит на разработчике СДГВМ, техническом консультанте, психолог необходим для корректировки конструктора квестов под педагогические и адаптивные цели: толерантность к социуму МУИВ и развития ответственности, эмоциональной поддержки, саморегулирования, маркетолог и технический консультант поддерживают техническую устойчивость и нужны для масштабирования монетизации стартапа конструктора квестов, это увеличивает прямую прибыль Московскому университету имени Сергея Юльевича Витте, выбранная конфигурация команды, позволяет минимизировать затраты ресурсов и принести больше прибыли университету.

Затраты разделены на три разных типа, разовые - инвестиции при запуске стартапа, переменные колеблющиеся, от количества клиентов, постоянные “фиксированные” - затраты не зависящие от объёма продаж, ниже приведена структура затрат.

Таблица 4.6.2 Структура затрат “СДГВМ”.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Категория затрат | Запуск стартапа | Рост стартапа | Масштабирование стартапа | Стабилизация стартапа | Итого за 5 лет | Доля |
| Разработка и обновления | 150 000 | 200 000 | 150 000 | 100 000 в год | 800 000 | 18 |
| Маркетинг и продвижение | 260 000 | 520 000 | 850 000 | 600 000 в год | 3 080 000 | 68 |
| Техническая инфраструктура | 80 000 | 50 000 | 30 000 | 20 000 в год | 220 000 | 5 |
| Юридические расходы | 50 000 | 20 000 | 10 000 | 5000 в год | 95 000 | 2 |
| Административные | 30 000 | 40 000 | 50 000 | 40 000 в год | 200 000 | 4 |
| Непредвиденные расходы | 30 000 | 50 000 | 80 000 | 60 000 в год | 320 000 | 7 |
| Итого затрат | 600 000 | 880 000 | 1 170 000 | 825 000 в год | 4 715 000 | 100 |

Затраты СДГВМ похожи на затраты обычного цифрового стартапа, маркетинг занимает 68% долю и это нужно для увеличения продаж через Unity Asset Store, а затраты на улучшение СДГВМ уменьшаются с каждым годом, в итоге общий бюджет получился на 4,7 миллионов рублей за пять лет, будет покрыт полностью “при максимальном достижимом объёме рынка генерации контента, для образовательных квестов” в 0,02% от всего мирового рынка, следовательно стартап имеет высокую рентабельность и высокую живучесть.

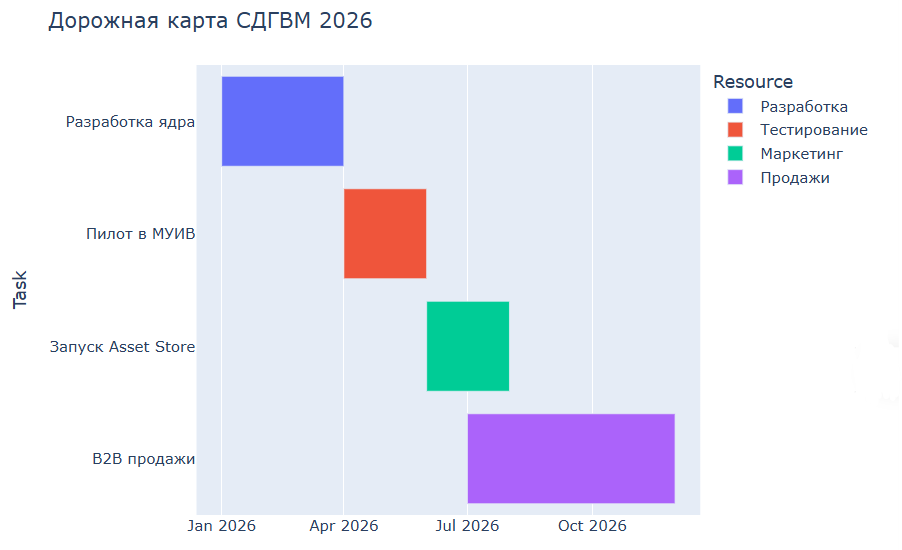


Рисунок 4.6.1 диаграмма Ганта, для развития стартапа.

Стартап конструктор квестов, экономически устойчив и значим, для решения локальной проблемы Московского университета имени Сергея Юльевича Витте, в адаптации к социуму иностранных студентов, увеличению уровня эмоциональной поддержки, демонстрация правильных моделей поведения, для студентов, ознакомления с правилами поведения в МУИВ, стартап имеет потенциал в дальнейшем развитии и способен приносить пользу в решении локальных проблем и прибыль университету, от продаж в Unity Asset Store и прямых продаж другим университетам.

**4.7 Расчёты инвестиций**

Проект СДГВМ, является университетским стартапом, где базовый функционал, реализован в виде дипломной работы, первоначальные инвестиции в проекте, направлены на маркетинг и масштабирование, для получения университетом прибыли, от стартапа, также инсвестиции направлены, на улучшение функционала, обновления мажорные и минорные, поддержку работоспособности СДГВМ, на протяжении жизненного цикла стартапа, масштабирование в выбранной узкой рыночной нише, важно для увеличения дохода от стартапа университетом, ставка дисконтирования принята, на уровне 25%, по причине не стабильности в мире, проблем с экономикой, рисков связанных с внешними источниками “потеря стабильного дохода, пропорции ценность клиента на стоимость привлечения будут меньше порога” и инфляции рубля, можно поделить возможные инвестиции, на два типа: ежегодные операционные затраты на поддержку работоспособности, улучшения, маркетинг “инфраструктурная поддержка” и единоразовое стартовое вложение, для запуска стартапа конструктор квестов.

Таблица 4.7.1. Объём инвестиций по годам.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Направление для инвестиций | Запуск | Рост | Масштаб | 2029 | 2030 | Итого за 5 лет |
| Разработка и обновления моделей | 150 000 | 200 000 | 150 000 | 100 000 | 100 000 | 700 000 |
| Маркетинг и продвижение | 260 000 | 520 000 | 850 000 | 600 000 | 600 000 | 2 830 000 |
| Техническая инфраструктура | 80 000 | 50 000 | 30 000 | 20 000 | 20 000 | 200 000 |
| Юридическая инфраструктура | 50 000 | 20 000 | 10 000 | 5 000 | 5 000 | 90 000 |
| Юридические расходы | 60 000 | 90 000 | 130 000 | 100 000 | 100 000 | 480 000 |
| Административные и резерв | 600 000 | 880 000 | 1 170 000 | 825 000 | 825 000 | 4 300 000 |

В сумме объём инвестиций, получился 4,3 миллиона рублей, основное направление для инвестиций - маркетинг и продвижение, для реализации стратегии на основе использования платформы Unity Asset Store, при помощи таргетинга. Благодаря локальной работе без использования API, и сетевых инфраструктур и использованию университетских вычислительных машин, также, благодаря существующим двум патентам, на отдельные модули конструктора квестов, генератор СДГВМ и части с визуальной новеллой имеющую название “конструктор квестов РФ”, юридические расходы меньше, но необходимы, для решения возникающих проблем из-вне, для привлечения специализированных юристов.

Таблица 4.7.2 Прогноз денежных потоков “при базовом сценарии”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Год | Доход в рублях | Расходы в рублях | Чистый поток денежных средств | Кумулятивный поток денежных средств |
| 2026 | 400 000 | 600 000 | -200 000 | -200 000 |
| 2027 | 2 000 000 | 880 000 | 1 120 000 | 920 000 |
| 2028 | 4 000 000 | 1 170 000 | 2 830 000 | 3 750 000 |
| 2029 | 6 000 000 | 825 000 | 5 175 000 | 8 925 000 |
| 2030 | 8 000 000 | 825 000 | 7 175 000 | 16 100 000 |

Стартап выйдет в плюс по доходам, не сразу, а только к 2027 году, к концу 2030 году ожидается прибыль в шестнадцать миллионов рублей, при базовом сценарии, что подтверждает наличие высокой доходности.

Таблица 4.7.3 Прогноз денежных потоков “при пессимистичном сценарии”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Год | Доход в рублях | Расходы в рублях | Чистый поток денежных средств | Кумулятивный поток денежных средств |
| 2026 | 250 000 | 780 000 | -530 000 | -530 000 |
| 2027 | 1 100 000 | 1 144 000 | - 44 000 | -574 000 |
| 2028 | 2 200 000 | 1 521 000 | 679 000 | 105 000 |
| 2029 | 3 300 000 | 1 072 500 | 2 227 500 | 2 332 500 |
| 2030 | 4 400 000 | 1 072 500 | 3 327 500 | 5 660 000 |

Пессимистичный сценарий показал снижение доходов на 50%, расходы увеличены на 30%, конструктор квестов выйдет в плюс по доходам к концу 2028 года, а кумулятивный поток станет положительным только к 2030 году и составит 5,66 миллионов рублей, Медленный рост и масштабирование стартапа при пессимистичном сценарии, все равно оставляет проект жизнеспособным: происходит накопление прибыли к 2030 году, и нету значительных убытков при планировании пессимистичного сценария.

Таблица 4.7.4 Прогноз денежных потоков “при оптимистичном сценарии”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Год | Доход в рублях | Расходы в рублях | Чистый поток денежных средств | Кумулятивный поток денежных средств |
| 2026 | 600 000 | 540 000 | 60 000 | 60 000 |
| 2027 | 3 500 000 | 792 000 | 2 708 000 | 2 768 000 |
| 2028 | 7 000 000 | 1 053 000 | 5 947 000 | 8 715 000 |
| 2029 | 10 500 000 | 742 500 | 9 757 500 | 18 472 500 |
| 2030 | 14 000 000 | 742 500 | 13 257 500 | 31 730 000 |

При оптимистичном сценарии кумулятивная прибыль к 2030 году достигнет 31,74 миллионов рублей, доходы выше, чем в базовом сценарии на 50%, и на 80% чем в пессимистичном сценарии, это продемонстрировало возможный потенциал стартапа конструктор квестов, при оптимистичном сценарии.

Таблица 4.7.5 Сценарии эффективности денежных потоков, при разных сценариях “NPV, IRR, PI,DPP”.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Сценарий | Инвестиции за 5 лет рублей | NPV | IRR | PI | DPP | Вывод |
| Пессимистичный | 6 000 000 | + 2 150 000 | 32% | 1,36 | 4,1 | Жизнеспособен |
| Базовый | 4 300 000 | + 9 850 000 | 68% | 3,29 | 2,8 | Высокодоходен |
| Оптимистичный | 3 200 000 | + 18 400 000 | 105% | 6,75 | 2,1 | Очень привлекателен |

В пессемистичном сценарии положителен NPV положительный, IRR превысил ставку дисконтирования, дисконтированный срок окупаемости составил чуть больше четырёх лет. Особенно привлекательными являются оптимистичный и базовый сценарий, показавшие выдающуюся доходность, благодаря такой высоким показателям доходности, стартап увеличивает привлекательность, для монетизации. Базовый сценарий, показал преимущество в высокой окупаемости привлечённых клиентов, таргетинговой рекламой, каждый вложенный рубль в привлечение новых клиентов, приходится 3,29 рублей.

**4.8 Анализ рисков и чувствительности**

Анализ рисков - обязательная переменная в формуле успеха стартапа СДГВМ, здравая оценка финансовых и операционных показателей проекта, позволит избежать возможных проблем в будущем, при масштабировании проекта, необходимо заранее принять меры, для снижения возникновения и реализации возможных рисков, также классификация рисков по источникам и степени влияния - ключевая стратегия, для предотвращения проблем у стартапа конструктор квестов. Риски были оценены, по степени ущерба и вероятности возникновения.

Таблица 4.8.1 Матрица рисков проекта “СДГВМ”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Риск | Вероятность | Ущерб | Уровень риска | Меры минимизации |
| Медленный рост продаж в Unity Asset Store | 5 | 5 | 25 (высокий) | SEO-оптимизация, ML-анализ отзывов, акции, конкурсы, регулярные обновления |
| Высокий САС из-за роста стоимости рекламы | 5 | 4 | 20 (высокий) | Увеличение доли органики, реферальные программы, партнёрства с геймдев - сообществами |
| Низкая конверсия B2B - продаж | 4 | 5 | 20 (высокий) | Пилотные внедрения в МУИВ, демонстрации, кейсы успеха, публикации статей |
| Технические проблемы (медленная генерация на слабом ПК) | 5 | 4 | 20 (высокий) | Постоянная оптимизация моделей (GGUF, CPU-only), документация пот требованиям железа |
| Негативные отзывы и низкий рейтинг в Asset Store | 4 | 5 | 20 (высокий) | ML - мониторинг тональности, оперативные ответы, бесплатные обновления, конкурсы за отзывы |
| Санкции блокировка Asset Store в РФ | 3 | 6 | 18 (высокий) | Оффлайн-решение как преимущество, альтернативные каналы (прямые продажи, Telegram, ВК) |
| Конкуренция от бесплатных локальных LLM | 4 | 4 | 16 (средний) | Уникальность ниши (культурная адаптация + образование), интеграция с Unity, квесты для вузов |
| Отток пользователей | 4 | 4 | 16 (средний) | Регулярные обновления, DLC, персонализация, поддержка, реферальные бонусы |
| Задержки в обновлениях моделей | 3 | 4 | 12 (средний) | План обновлений, резервные модели, участие студентов в тестировании |
| Юридические риски (патент, лицензии) | 2 | 5 | 10 (средний) | Регистрация ПО в Роспатенте, проверка лицензий моделей |
| Низкая мотивация команды | 3 | 3 | 9 средний (средний) | Вовлечение студентов МУИВ, публикации статей, участие в конференциях |
| Технические сбои на стороне пользователя | 4 | 2 | 8 (низкий) | Подробная документация, FAQ, видео -инструкции, техподдержка через Telegram |

Самыми опасными рисками являются: медленный рост в Unity Asset Store, высокая стоимотсь привлечения клиентов, а также низкая конверсия B2B лицензий, но существуют способы предотвратить или частично минимизировать риски, до конкретного уровня.

Для прогнозирования изменения ключевых параметров “NPV, IRR, кумулятивный CF” нужно провести анализ чувствительности, ниже в таблице приведены значения чувствительности, для ключевых параметров.

Таблица 4.8.2 Чувствительность ключевых показателей к изменению параметров.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Параметр | Изменение | NPV миллионов рублей | IRR % | Кумулятивный CF в миллионах рублей | Комментарий |
| Базовый сценарий | 0% | +9,85 | 68 | +16,1 | Исходный |
| Доход -30% (медленные продажи) | -30% | +4,12 | 42 | +7,8 | Всё ещё положительный NVP |
| Доход +50% (вирусный росты) | +50% | +18,4 | 105 | +31,7 | Очень высокая доходность |
| Расходы +50% (рост САС) | +50% | +2,15 | 32 | +5,66 | Жизнеспособен |
| Churn +20% (отток пользователей) | +20% | +6,20 | 51 | +11,2 | Умеренное снижение |
| Ставка дисконтирования +10% (35%) | +6,80 | - | - | - | NPV снижается, но > 0 |

СДГВМ продемонстрировал, высокую устойчивость к негативным изменениям, преимущество реализуемого стартапа в живучести и защите от отклонений ключевых экономических показателей включая снижение дохода или роста расходов на 50%, чистая приведённая стоимость остаётся в норме, даже при сильных отклонениях, таблица подтверждает факт “высокой жизнеспособности проекта”.

Можно отметить, экономические и внешние риски не поставят под угрозу задачи решения локальной проблемы Московского университета имени Сергея Юльевича Витте в адаптации иностранных студентов, под социум и правила университета, благодаря устойчивости и живучести стартапа, работа над продуктом, не будет прекращена, и будет прямолинейно продолжаться, это стратегически важный и ценный проект, не теряющей своей ценности, при любых обстоятельствах, влияющий на решение проблем адаптации студентов, под Московский институт имени Сергея Юльевича Витте.

**4.9 Оценка эффективности внедрения в МУИВ**

Внедрение СДГВМ в инфраструктуру Московского университета имени Сергея Юльевича Витте, на кафедру информационных систем, нужно для решения локальных проблем университета в адаптации иностранных студентов под социум университета, автоматизацию процесса создания образовательного контента “геймификация обучения”, но главная цель это получения прибыли от проекта университетом, повышение рейтинга устойчивого развития университета мировыми экспертами и критиками, коммерчески успешный и живучий стартап как диплом, будет приносить прибыль Московскому университету имени Сергея Юльевича Витте, эффективность стартапа проявляется в укреплении имиджа университета и повышении качества образовательного процесса, социальная адаптация иностранных студентов реализуется через квесты насыщенные культурой и историей Российской Империи, помогает глубоко погрузить студентов в контекст университета, понять правила и традиции социума, для увеличения уровня самоконтроля, ответственности, успеваемости, у студента формируется правильная модель поведения, а геймификация снижает психологический барьер при обучении. Также система проще в обслуживании, благодаря оффлайн технологиям генерации при помощи локального сервера ComfyUI c Awesome RPG Icon 2000 и языкового сервиса LLM - Unity, в который загружены локальные модели Mistral - 7B - Instruct, требует меньше затрат, для технической поддержки, не требует затрат на лицензии или API сервиса, не зависит от Google Classroom, зарубежных LMS и облачных сервисов. Проведена оптимизация, для работы на компьютерах МУИВ, квантованные модели, распределение ресурсов, работа без видеокарт, что позволяет внедрить инновационные технологии ИИ, без масштабной модернизации аппаратного парка Московского университета имени Сергея Юльевича Витте. Стартап конструктор квестов, содержит в себе обширную базу, для научных исследований, NLP и генеративного дизайна, также психологические научные исследования, влияние сгенерированного нейросетями контента, на адаптацию к социуму, или использование искусственного интеллекта в образовательной среде, влияние на другие аспекты жизни иностранных студентов и студентов МУИВ, стартап создаёт исследовательскую среду и обширную базу, для научных статей. Для оценки экономической эффективности, нужно рассмотреть процесс создания набора состоящего из пятидесяти обучающих квестов, например для курса адаптации иностранных студентов, первый вариант: ручное создание, на разработку сценария и ветвления диалогов требуется четыре часа, поиск и подбор визуальных ассетов, “включает процесс проверки прав” заёмёт два часа, внесение в LMS и вёрстка займёт приблизительно час времени, в сумме один квест при ручном создании займёт около семи часов времени, при ставке преподавателя 330 рублей в час, затраты университета составят , что дороже варианта, без использования СДГВМ: Генерация сценария и диалогов 3 варианта на выбор, 15 минут, генерация иконок и ассетов 15 минут, исправление смысловых ошибок и аппробация сгенерированного контента преподавателем 30 минут, итого в сумме на один квест получается 1 час, по формуле получается лей, получается экономия ровно в четырнадцать раз, , это ещё не вся экономия, при использовании СДГВМ, траты на электроэнергию при потреблении 0,5 кВт, лей, получается общая экономия составит лей, в рамках одного курса получается сэкономить 119 825 рублей и 300 часов рабочего времени, для преподавателей, СДГВМ не только деньги экономит, но и освобождает время работы у преподавателей, сэкономленное время можно использовать в пользу, для научной и педагогической работы, не стоит забывать что стартап, делается с целью получения прибыли Московским университетом имени Сергея Юльевича Витте, по гибридной бизнес - модели проекта, основными каналами получения денежной прибыли, будут Unity Asset Store и прямые продажи другим университетам, кафедрам, дополнительный доход от стартапа будет составлять около миллиона рублей в год, окупаемость проекта составит два или три года, при базовом сценарии, а также улучшение имиджа университета и научная база, это критерии делающие конструктор квестов и СДГВМ стратегическим важным стартапом, для устойчивого развития и цифровой трансформации Московского университета имени Сергея Юльевича Витте.

**4.10 Выводы по экономической части**

На основе проведённого анализа разработки и внедрения СДГВМ, в Московский институт имени Сергея Юльевича Витте, сделаны следующие выводы: годовая экономия в 14 раз, по сравнению с ручным трудом, для вуза при создании сто адаптирующих к социуму МУИВ квестов, будет больше 150 000 рублей, расчётный срок окупаемости проекта составляет, менее 5 месяцев, после выхода из стадии MVP, но это при условии коммерциализации, через платформу Unity Asset Store, а также СДГВМ соответствует критическим факторам успеха организации, решает локальную проблему адаптации студентов, под социум Московского университета имени Сергея Юльевича Витте , и способствует цифровой трансформации при помощи автоматизации создания цифровых “виртуальных квестов”, также стартап имеет потенциал для коммерциализации, как самостоятельный стартап, существует два авторских свидетельства на программу, подтверждающие новизну и права университета, на коммерциализацию.

Стартап конструктор квестов, является экономически устойчивым, проектом, для монетизации, за счёт предварительного анализа возможных рисков, для минимизации возможного ущерба, а также внедряемая динамическая система генерации виртуальных миров, оптимизирована для работы на вычислительных машинах Московского университета имени Сергея Юльевича Витте, не требуется модернизации оборудования, и не требуется покупка лицензий, используются локально установленные модели, со свободным использованием.

Также нужно указать на рост выбранной рыночной ниши генерации образовательного контента EdTech, на данный момент размер мирового рынка составляет около 17,32 миллиардов рублей, ожидаемый доход к 2028 году будет 1-2 млн рублей, при наличие маркетинга для продаж через Unity Asset Store и прямых продаж, другим университетам и кафедрам.

Низкая стоимость привлечения клиента, и высокая окупаемость, окупаемость клиента за пять месяцев, машинное обучение при использовании добавляет к окупаемости 450 тысяч в год, минимальная команда состоит из семи, бюджет в основном идёт на маркетинг - 68% на маркетинг, инвестиции в стартап составляют 4,3 миллиона рублей за пять лет, даже пессимистичный сценарий, показывает жизнеспособность СДГВМ, значения чистой приведенной стоимости составляли 2,15 миллионов рублей.

При внедрении СДГВМ, наблюдается большая экономия денежных и временных средств, 300 часов работы преподователей - разработчиков виртуально образовательных квестов, и 119 850 рублей на 100 сделанных квестов.

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения выпускной квалификационной работы реализован и подготовлен к коммерциализации полноценный стартап - продукт решающий локальную социальную задачу университета, с высоким потенциалом к масштабируемости дохода, на рынке EdTech СДГВМ, используемый для создания контента адаптирующих к социуму Московского университета имени Сергея Юльевича Витте квестов, в виде визуальных новелл, используемых для увеличения адаптации иностранных студентов к элементам культуры Российской Федерации, увеличения уровня эмоциональной поддержки, саморегулирования, успеваемости и ответственности, произведена интеграция больших языковых моделей Mistral-7B-Instruct квантованную до GGUF формата, оптимизированную под работу на вычислительных машинах университета, через пакет LLM-Unity, для генератора икон интегрирована диффузионная модель Awesome RPG Icon 2000, для интеграции использовался портативный ComfyUI версии для amd позволяющий запускать генерацию без видеокарты, продуман жизненный цикл СДГВМ включающий эволюцию инструментария, используя мажорное и минорное версионирование, стартап системы динамической генерации виртуальных миров имеет дисконтированный срок окупаемости пять месяцев, прибыль от стартапа составляет 984210 рублей, инструментарий СДГВМ перспективен в условиях растущей потребности генераций контента на рынке геймдева, практическую значимость и научную новизну стартапа подтверждает разработанный проект визуальной новеллы использованный для адаптации иностранных студентов, психологическое тестирование показало положительную динамику прироста толерантности к национальной культуре Российской Федерации, стартап СДГВМ готов для использования, полученные в итоге квесты дорабатываются до уровня образовательных материалов при помощи знатоков истории и культуры, полученный в итоге стартап будет развиваться в дальнейшем, благодаря наличию возможностей исследовать аспекты психологии иностранных студентов и возможности коммерциализации СДГВМ в качестве пакета Unity Asset Store, продажа лицензии на программу другим университетам.

**СПИСОК ЛИТЕРАТУРЫ**

1. Achiam J. GPT-4 Technical Report / J. Achiam [и др.] // arXiv:2303.08774. – 2023.

2. Anthropic. Constitutional AI: Harmlessness from AI Feedback / Y. Bai [и др.] // arXiv:2212.08073. – 2022.

3. Bangor A. Determining what individual SUS scores mean: Adding an adjective rating scale / A. Bangor, P. Kortum, J. Miller // Journal of Usability Studies. – 2009. – Vol. 4, № 3. – P. 114–123.

4. Brown T. B. Language Models are Few-Shot Learners / T. B. Brown [и др.] // Advances in Neural Information Processing Systems. – 2020. – Vol. 33. – P. 1877–1901.

5. Chen M. Evaluating Large Language Models Trained on Code / M. Chen [и др.] // arXiv:2107.03374. – 2021.

6. Chiang W.-L. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%\* ChatGPT Quality / W.-L. Chiang [и др.] // LMSYS Org. – 2023.

7. Chowdhery A. PaLM: Scaling Language Modeling with Pathways / A. Chowdhery [и др.] // arXiv:2204.02311. – 2022.

8. Chung H. W. Scaling Instruction-Finetuned Language Models / H. W. Chung [и др.] // arXiv:2210.11416. – 2022.

9. Clark P. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge / P. Clark [и др.] // arXiv:1803.05457. – 2018.

10. Cobbe K. Training Verifiers to Solve Math Word Problems / K. Cobbe [и др.] // arXiv:2110.14168. – 2021.

11. Devlin J. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / J. Devlin [и др.] // NAACL-HLT. – 2019. – P. 4171–4186.

13. Dettmers T. QLoRA: Efficient Finetuning of Quantized LLMs / T. Dettmers [и др.] // NeurIPS. – 2023.

14. Du Z. GLM: General Language Model Pretraining with Autoregressive Blank Infilling / Z. Du [и др.] // ACL. – 2022.

15. Fedus W. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity / W. Fedus [и др.] // arXiv:2101.03961. – 2021.

16. Hoffmann J. Training Compute-Optimal Large Language Models / J. Hoffmann [и др.] // arXiv:2203.15556. – 2022.

17. Hu E. J. LoRA: Low-Rank Adaptation of Large Language Models / E. J. Hu [и др.] // arXiv:2106.09685. – 2021.

18. Иванов А. В. Применение генеративных нейронных сетей в образовании: обзор 2020–2025 / А. В. Иванов, Е. С. Петрова // Вестник МГТУ им. Н.Э. Баумана. – 2025. – № 2. – С. 45–58.

19. Jiang A. Q. Mistral 7B / A. Q. Jiang [и др.] // arXiv:2310.06825. – 2023.

20. Kaplan J. Scaling Laws for Neural Language Models / J. Kaplan [и др.] // arXiv:2001.08361. – 2020.

21. Köpf A. OpenAssistant Conversations – Democratizing Large Language Model Alignment through Open Collaboration / A. Köpf [и др.] // arXiv:2304.07327. – 2023.

22. Козлов Д. С. Интеграция ComfyUI в игровые движки для генерации 2D-ассетов / Д. С. Козлов // Программные продукты и системы. – 2024. – № 3. – С. 78–85.

23. Lewis M. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension / M. Lewis [и др.] // ACL. – 2020.

24. Ли Ч. Stable Diffusion в образовательных приложениях: опыт 2023–2025 / Ч. Ли, С. Ким // International Journal of Educational Technology. – 2025. – Vol. 17. – P. 201–215.

25. Liu H. LLaMA-Adapter: Efficient Fine-tuning of Language Models with Zero-init Attention / H. Liu [и др.] // arXiv:2303.16199. – 2023.

26. Loshchilov I. Decoupled Weight Decay Regularization / I. Loshchilov, F. Hutter // ICLR. – 2019.

27. Luccioni A. S. Stable bias: Evaluating societal representations in diffusion models / A. S. Luccioni [и др.] // NeurIPS Workshop. – 2023.

28. Николаев С. В. Экономическая эффективность внедрения генеративного ИИ в вузы / С. В. Николаев // Экономика образования. – 2025. – № 1. – С. 56–67.

29. Nkomo L. M. Facilitating student engagement through the flipped learning approach in K-12: A systematic review / L. M. Nkomo [и др.] // Computers & Education. – 2021. – Vol. 167. – Art. 104184.

30. Ouyang L. Training language models to follow instructions with human feedback / L. Ouyang [и др.] // NeurIPS. – 2022.

31. O’Regan L. Conceptualising student engagement as a theoretical framework for higher education / L. O’Regan, M. Brown // All Ireland Journal of Higher Education. – 2025. – Vol. 17, № 1. – P. 45–62.

32. Петрова Е. Н. Культурная адаптация иностранных студентов через геймификацию / Е. Н. Петрова, В. А. Смирнов // Высшее образование в России. – 2024. – № 6. – С. 89–97.

33. Podell D. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis / D. Podell [и др.] // arXiv:2307.01952. – 2023.

34. Radford A. Learning Transferable Visual Models From Natural Language Supervision / A. Radford [и др.] // ICML. – 2021. – P. 8748–8763.

35. Ramesh A. Zero-Shot Text-to-Image Generation / A. Ramesh [и др.] // ICML. – 2021. – P. 8821–8831.

36. Rombach R. High-Resolution Image Synthesis with Latent Diffusion Models / R. Rombach [и др.] // CVPR. – 2022. – P. 10684–10695.

37. Сидоров П. А. Генерация образовательного контента с помощью Mistral-7B в русскоязычной среде / П. А. Сидоров // Информатика и образование. – 2025. – № 4. – С. 112–120.

38. Smith J. The Role of AI-Generated Content in Cultural Adaptation Programs / J. Smith, M. Garcia // Journal of Cross-Cultural Psychology. – 2024. – Vol. 55. – P. 678–690.

39. Touvron H. LLaMA: Open and Efficient Foundation Language Models / H. Touvron [и др.] // arXiv:2302.13971. – 2023.

40. Touvron H. Llama 2: Open Foundation and Fine-Tuned Chat Models / H. Touvron [и др.] // arXiv:2307.09288. – 2023.

41. Васильев И. П. Автоматизация создания квестов для Moodle с помощью LLM / И. П. Васильев // Дистанционное и виртуальное обучение. – 2025. – № 5. – С. 34–42.

42. Villamañe M. Facilitating and automating usability testing of educational technologies / M. Villamañe [и др.] // Computer Applications in Engineering Education. – 2024. – Vol. 32, № 1. – Art. e22678.

43. Wang Y. LLM-Unity: Integrating Large Language Models into Unity Game Engine / Y. Wang, X. Liu // Journal of Artificial Intelligence Research. – 2024. – Vol. 79. – P. 123–145.

44. Wei J. Emergent Abilities of Large Language Models / J. Wei [и др.] // TMLR. – 2022.

45. Zamfirescu-Pereira J. D. Designing interfaces for text-to-image prompt engineering using stable diffusion models / J. D. Zamfirescu-Pereira [и др.] // arXiv:2303.11519. – 2023.

46. Zhang T. Avatar Generation from Text using Stable Diffusion and LLM / T. Zhang, Y. Li // IEEE Access. – 2024. – Vol. 12. – P. 45678–45689.

47. Zhang T. Qwen-VL: A Frontier Large Vision-Language Model with Versatile Abilities / T. Zhang [и др.] // arXiv:2308.12966. – 2023.

48. Zhou C. LIMA: Less Is More for Alignment / C. Zhou [и др.] // NeurIPS. – 2023.

**Техническое задание на создание информационной системы**

**ГЕНЕРАЦИЯ ДИНАМИЧЕСКИХ ИГРОВЫХ МИРОВ ПРИ ПОМОЩИ НЕЙРОСЕТЕЙ**

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

**ТЗ - 34.602-2020**

Москва 2026 год

**СОДЕРЖАНИЕ**

[1 ОБЩИЕ СВЕДЕНИЯ 5](#_Toc68" \o "#_Toc68)

[1.1 Обозначение и полное наименование инструмента генерации контента 5](#_Toc69" \o "#_Toc69)

[1.2 Номер договора (контракта) 5](#_Toc70" \o "#_Toc70)

[1.3 Наименования организации-заказчика и организаций- участников работ 5](#_Toc71" \o "#_Toc71)

[1.4 Перечень документов, на основании которых создается система 5](#_Toc72" \o "#_Toc72)

[1.5 Плановые сроки начала и окончания работы по созданию системы 5](#_Toc73" \o "#_Toc73)

[1.6 Источники и порядок финансирования работ 5](#_Toc74" \o "#_Toc74)

[1.7 Порядок оформления и предъявления заказчику результатов работ по созданию системы 6](#_Toc75" \o "#_Toc75)

[1.9 Перечень нормативно-технических документов, методических материалов, использованных при разработке ТЗ 6](#_Toc76" \o "#_Toc76)

[1.9 Определения, обозначения и сокращения, обозначения и сокращения. 7](#_Toc77" \o "#_Toc77)

[2. НАЗНАЧЕНИЕ И ЦЕЛИ СОЗДАНИЯ СИСТЕМЫ 9](#_Toc78" \o "#_Toc78)

[2.1 Назначение системы 9](#_Toc79" \o "#_Toc79)

[2.2 Цели создания системы 9](#_Toc80" \o "#_Toc80)

[3 ХАРАКТЕРИСТИКА ОБЪЕКТА АВТОМАТИЗАЦИИ 11](#_Toc81" \o "#_Toc81)

[3.1 Краткие сведения об объекте автоматизации или ссылки на документы, содержащие такую информацию 11](#_Toc82" \o "#_Toc82)

[3.2 Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды 12](#_Toc83" \o "#_Toc83)

[4. ТРЕБОВАНИЯ К СИСТЕМЕ 14](#_Toc84" \o "#_Toc84)

[4.1 Требования к структуре 14](#_Toc85" \o "#_Toc85)

[4.1.1 Описание подсистем динамического генератора «СДГВМ» 14](#_Toc86" \o "#_Toc86)

[4.1.2 Разработанный функционал подсистем генератора. 15](#_Toc87" \o "#_Toc87)

[4.1.3 Логические цепочки для входных данных. 15](#_Toc88" \o "#_Toc88)

[4.1.4 Возможность интеграции новых инструментов в «СДГВМ» 16](#_Toc89" \o "#_Toc89)

[4.1.5 Виды режимов работ инструмента генерации «СДГВМ» 16](#_Toc90" \o "#_Toc90)

[4.1.5.1 Возможности Unity Console для диагностирование и логирования ошибок 17](#_Toc91" \o "#_Toc91)

[4.1.5.2 Дальнейшая эволюция и перспективы системы генерации «СДГВМ» 17](#_Toc92" \o "#_Toc92)

[4.1.5.3 Обзор набора исполняющего персонала для оперирования «СДГВМ» 18](#_Toc93" \o "#_Toc93)

[4.1.6 Необходимые умения и знания для персонала генерирующего при помощи «СДГВМ» контент 18](#_Toc94" \o "#_Toc94)

[4.1.7 Данные о количественных и качественных показателях «СДГВМ» для платформы Unity 19](#_Toc95" \o "#_Toc95)

[4.1.8 Решение проблем с случаями сбоев на локальной машине генерации инструмента «СДГВМ» 20](#_Toc96" \o "#_Toc96)

[4.1.9 Безопасность и защита генерируемого контента инструментом «СДГВМ» 20](#_Toc97" \o "#_Toc97)

[4.1.10 Требования к визуальному дизайну, цветовой схеме и эстетике UI 20](#_Toc98" \o "#_Toc98)

[4.1.11 Совместимость инструмента генерации «СДГВМ» с другими платформами и OS 21](#_Toc99" \o "#_Toc99)

[4.1.12 «СДГВМ» – локальная работа в Unity 21](#_Toc100" \o "#_Toc100)

[4.1.13 Сохранение творческого потенциала сгенерированного контента 22](#_Toc101" \o "#_Toc101)

[4.1.14 Осуществление сохранения сгенерированного контента в случаях технических неисправностей 22](#_Toc102" \o "#_Toc102)

[4.1.15 Защита локальной машины с «СДГВМ» от воздействий с внешней стороны 22](#_Toc103" \o "#_Toc103)

[4.1.16 Патентование уникальных алгоритмов генерации при помощи авторского свидетельства 22](#_Toc104" \o "#_Toc104)

[4.1.17 Важные факторы совместимости и стандартизации «СДГВМ» 23](#_Toc105" \o "#_Toc105)

[4.1.18 Необходимый для удобства пользования функционал системы генерации «СДГВМ» 23](#_Toc106" \o "#_Toc106)

[4.2 Требования к функциям (задачам), выполняемым системой 23](#_Toc107" \o "#_Toc107)

[4.2.1 Модульная подсистема «СДГВМ» для генерации контента 24](#_Toc108" \o "#_Toc108)

[4.2.1.1 Блок «Сказитель историй» генерация квестов 24](#_Toc109" \o "#_Toc109)

[4.2.1.2 Блок «Контроллер поведения не игровых персонажей» генерация диалогов 24](#_Toc110" \o "#_Toc110)

[4.2.1.2.1 Математический анализ контролера поведения Mistral 7B – Instruct 24](#_Toc111" \o "#_Toc111)

[4.2.1.2.2 Обоснование выбора языковой модели для генерации квестов и диалогов 27](#_Toc112" \o "#_Toc112)

[4.2.1.2.3 Внутренее строение Mistral 7B - Instruct 27](#_Toc113" \o "#_Toc113)

[4.2.1.3 Блок «Генератор икон» генерация визуальных активов 28](#_Toc114" \o "#_Toc114)

[4.2.1.3.1 Реализация генерации иконок и спрайтов через Comfy UI 28](#_Toc115" \o "#_Toc115)

[4.2.1.3.2 Обоснование выбора моделей sd turbo и sd3 medium, для генерации контента 28](#_Toc116" \o "#_Toc116)

[4.2.1.4 Блок «Самоучитель с обратной связью» обучающие материалы, фидбэк 29](#_Toc117" \o "#_Toc117)

[4.2.1.5 Блок «Инструмент для автоматического создания интерфейса «СДГВМ»» 29](#_Toc118" \o "#_Toc118)

[4.2.1.5 Блок «Автоматизированное тестирование» проведение нескольких генераций с выводом графиков и рекомендаций 30](#_Toc119" \o "#_Toc119)

[4.2.2 Разработка Модульной подсистемы «СДГВМ» для генерации игрового контента 30](#_Toc120" \o "#_Toc120)

[4.3 Требования к видам обеспечения 34](#_Toc121" \o "#_Toc121)

[4.3.1 Требования к математическому обеспечению системы 34](#_Toc122" \o "#_Toc122)

[4.3.2 Требования к информационному обеспечению системы 34](#_Toc123" \o "#_Toc123)

[4.3.3 Требования к лингвистическому обеспечению системы 35](#_Toc124" \o "#_Toc124)

[4.3.4 Требования к программному обеспечению системы 36](#_Toc125" \o "#_Toc125)

[4.3.5 Требования к техническому обеспечению 36](#_Toc126" \o "#_Toc126)

[4.3.6 Требования к метрологическому обеспечению 36](#_Toc127" \o "#_Toc127)

[4.3.7 Требования к организационному обеспечению 37](#_Toc128" \o "#_Toc128)

[4.3.8 Требования к методическому обеспечению 37](#_Toc129" \o "#_Toc129)

**[5. СОСТАВ И СОДЕРЖАНИЕ РАБОТ ПО СОЗДАНИЮ СИСТЕМЫ](#_Toc130" \o "#_Toc130)** [39](#_Toc130" \o "#_Toc130)

[6 ПОРЯДОК РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ 41](#_Toc131" \o "#_Toc131)

[6.1 Виды, состав, объем и методы испытаний системы 41](#_Toc132" \o "#_Toc132)

[6.2 Общие требования к приемке работ по стадиям 41](#_Toc133" \o "#_Toc133)

[6.3 Статус приёмочной комиссии 42](#_Toc134" \o "#_Toc134)

[7 ТРЕБОВАНИЯ К СОСТАВУ И СОДЕРЖАНИЮ РАБОТ ПО ПОДГОТОВКЕ ОБЪЕКТА АВТОМАТИЗАЦИИ К ВВОДУ СИСТЕМЫ В ДЕЙСТВИЕ 43](#_Toc135" \o "#_Toc135)

[8 ТРЕБОВАНИЯ К ДОКУМЕНТИРОВАНИЮ 45](#_Toc136" \o "#_Toc136)

[9 ИСТОЧНИКИ РАЗРАБОТКИ 47](#_Toc137" \o "#_Toc137)

# 1 ОБЩИЕ СВЕДЕНИЯ

**1.1 Обозначение и полное наименование инструмента генерации контента**

Разработанный прототип ИИ инструмента генерации динамических игровых миров имеет название «Система динамической генерации виртуальных миров».

Условное обозначение Аббревиатура : СДГВМ «Генерация адаптивных миров».

**1.2 Номер договора (контракта)**

Настоящее Техническое задание разработано в рамках выполнения работ по Государственному контракту № ГК-178-ОФ/Д01, заключённому 7 ноября 2011 года.

**1.3 Наименования организации-заказчика и организаций- участников работ**

Заказчик:Московский институт имени С.Ю. Витте

Место нахождения:115432, Москва, второй Кожуховский проезд, дом двенадцать, строение один

Исполнитель:Шебанов Вячеслав Викторович

Место нахождения:141700, г. Долгопрудный, ул. Первомайская д.42.

**1.4 Перечень документов, на основании которых создается система**

Руководство по разработке больших языковых моделей LLM-Unity, спецификация моделей поведения игровых персонажей ONNX, различная документация Unity в области интеграции Python кода в проект для обучения адаптивных моделей поведения игровых персонажей.

**1.5 Плановые сроки начала и окончания работы по созданию системы**

Плановый срок начала работ – 17 сентября день подтверждения темы диплома.

Плановый срок окончания работ – 8 июня 2026 года.

**1.6 Источники и порядок финансирования работ**

Финансирование выходит из самостоятельного распределения ресурсов из личных средств разработчика, следовательно использовано программное обеспечение в публичном доступе и доступные инструменты генерации контента.

**1.7 Порядок оформления и предъявления заказчику результатов работ по созданию системы**

Функционирующая система в виде Unity проекта «динамической генерации виртуальных миров» интегрирована с платформой Unity 2D, предоставлена университету в установленный срок техническим заданием, демонстрационная версия передаваемого инструмента включает код программы, документацию, Unity проект для запуска, предоставляемая версия программы доступна через github. Работу принимает комиссия, в соответствии с положениями учебной организации, испытания и окончательная приемка показаны в шестом разделе технического задания, сдача документации, отчета и инструкций происходит на первичных этапах и регулируются требованием к документированию.

**1.9 Перечень нормативно-технических документов, методических материалов, использованных при разработке ТЗ**

Инструмент «система динамической генерации виртуальных миров» разработан с учетом методических и нормативных стандартов и документаций:

ГОСТ ИСО/МЭК 12207-2010 – используется для формирование инструмента начиная от начального прототипа с минимальным функционалом во время жизненного цикла проекта;

ГОСТ 34.602-89. Стандарт помог структурировать проект инструмента на Unity в соответствии с профессиональными нормами, поставил четкое определение функционала системы генерации и принципы реализации функций генерации игрового контента.

ГОСТ Р ИСО/ МЭК 25010-2015. Организовал процесс тестирования в Unity для соответствиям критериям оценки и ожиданиям клиентов, система приводится в соответствие стандартам оценки комиссии.

ГОСТ 34.601-90. Адаптирует создание инструмента с стадии прототипа до рабочего варианта с документацией и организацией всех этапов разработки.

ГОСТ 19.701-90 задает стандарты для создания диаграмм EPC, As Is, BPMN, To Be;

ГОСТ Р ИСО 9241-11-2014. Помогает интерфейсу в инструменте генерации соответствовать всем требованиям удобства.

**1.9 Определения, обозначения и сокращения, обозначения и сокращения.**

Таблица 1. Определения, обозначения и сокращения.

|  |  |  |
| --- | --- | --- |
| № | Сокращение | Расшифровка |
| 1 | СДГВМ | Система динамической генерации виртуальных миров |
| 2 | ONNX | Открытый обмен нейронными сетями (Open Neural Network Exchange) |
| 3 | NPC | Управляемый искусственным интеллектом персонаж (Non Player Character) |
| 4 | Visual Forge | Бесплатный инструмент создания иконок и спрайтов |
| 5 | GitHub | Система контроля управления версиями инструмента генерации адаптивных игровых миров |
| 6 | Прототип | Имитация функционала и внешнего вида готового продукта, существуют для тестов и сбора обратной связи помогая сэкономить деньги и временные ресурсы (Prototype) |
| 7 | Ассет | Деталь для сбора прототипа, (текстуры, иконки, спрайты, модели, эффекты, материалы) (Asset) |
| 8 | Инди-разработчик | Небольшая студия или один человек, не способный позволить лучшие альтернативы реализуемого инструмента (Indie Dev) |
| 9 | Пользовательский интерфейс | Набор кнопок и полей ввода для взаимодействия пользователей с инструментом генерации (UI – User Interface) |
| 10 | Спрайт | Картинка для визуализации иконок объектов и аватаров персонажей (Sprite) |
| 11 | Квест | Задание для игрока в виртуальном мире |
| 12 | ML-Agents | Инструмент интегрируемый в Unity для обучения не игровых персонажей |
| 13 | Unity | Платформа для создания игровых прототипов |
| 14 | Промт | Запрос для нейросети записываемый в виде текста (Prompt) |
| 15 | Игровой движок | Комплекс инструментов для создания игр (Game Engine) |
| 16 | Билд | Готовая версия игрового прототипа для запуска в виде .exe |
| 17 | Реиграбельность | Желание игроков поиграть в игру ещё раз (replay value) |
| 18 | Канвас | Область для размещения элементов UI |
| 19 | Unity Play Mode | Режим тестирования игровой сцены в Unity |
| 20 | Слайдер | Переключатель значения в выпадающем списке в интерфейсе (Slider) |
| 21 | Трансформеры | Вид нейронных сетей, преобразующих n-граммы в эмбеддинги, итеративно генерируют текст в квестах или в диалоге, следующее слова выбирают использованием голов внимания |
| 22 | Эмбеддинги | N-граммы преобразованые в векторный числовой вид для вычисления и анализа логической связанности в генерируемом тексте |
| 23 | Метрика BLEU | Метрика замеряет качество сгенерированного текста с эталонным образцом |
| 24 | N-граммы | В «СДГВМ» параметры в виде тэгов являющимися классами, для выполнения классификации нейронной сетью, в виде биграмм |
| 25 | Токены | Классифицирующийся текст находящийся в файлах JSON |
| 26 | Головы внимания | Используются нейронной сетью для определения слова подходящего под контекст для следующего шага генерации текста |
| 27 | Языковая модель | В проекте «СДГВМ» в роли языковой модели выступает трансформер Mistral 7B – Instruct обученный под чат-бот |
| 28 | Кросс энтропийный лосс | Используется в качестве функции минимизации потерь, для обучения языковой модели |
| 29 | HuggingFace | Сайт с публичным доступом к готовым моделям для скачивания |
| 30 | Gguf | Оптимизирующий использование ресурсов формат файла модели Mistral 7B - Instruct |
| 31 | Mistral 7B Instruct | Выбранная языковая модель превосходящая конкурентов в производительности, качества логической связанности и скорости, подходящая для интеграции в платформу Unity |
| 32 | Stable Diffusion | Интегрированная в Unity модель использующаяся для генерации изображений |
| 33 | Unity Sprite Generator | Пакет в Unity использующийся для автоматизации генерации спрайтов |
| 34 | Unity Raw Image | Создаваемый компонент для UI, позволяет вставить сгенерированное изображение |
| 35 | Квантизация | Процесс преобразования модели нейронной сети в gguf формат, для уменьшения потребления ресурсов |
| 36 | ONNX Runtime | Позволяет запускать ONNX в Unity используя режим Play mode |
| 37 | Dropdown | Выпадающий список в иерархии объектов канваса графического интерфейса пользователя |
| 38 | TensorFlow | Библиотека Python использующаяся при разработке модели Mistral 7B - Instruct |
| 39 | BERT | Не подходящая альтернативная модель трансформера, для генерации текста |
| 40 | LLaMA | Альтернатива Mistral 7B - Instruct, уступающая выбранной языковой модели в производительности и оптимизации |
| 41 | NumPy | Библиотека необходимая для запуска сервера генерации визуалов, работает с многомерными массивами |
| 42 | Einops | Библиотека созданная на основе нотации Эйнштейна, служит для манипуляций с тензорами «необходима для запуска сервера, через скрипт main.py» |
| 43 | Psutil | Библиотека позволяющая управлять запущенными процессами, библиотека реализует функции UNIX, позволяет создавать интерфейс с информацией о процессах и системе в целом. |
| 44 | Pytorch | Основная библиотека для реализации машинного обучения, служит для построения нейросетей, содержит больше функционала, чем у TensorFlow |
| 45 | TensorFlow | Библиотека на которой построена Mistral-7B-Instruct |
| 46 | Torchvision | Библиотека для реализации компьютерного зрения в «СДГВМ», требуется для запуска сервера генерации визуалов |
| 47 | Torchaudio | Библиотека для реализации работы с аудио-данными в «СДГВМ», требуется для запуска сервера генерации визуалов |
| 48 | Torchsde | Библиотека для реализации оптимизации модели через стахостические дифференциальные уравнения, позволяет обучать sd-turbo и sd3-medium по принципам GAN «генеративно состязательных сетей» |
| 49 | PyAV | Библиотека позволяет управлять деталями в обработке видео и аудио материалов, без прямой необходимости взаимодействия с низкоуровневыми API |
| 50 | Aiohttp | Библиотека для работы с сервером, необходимого для работы подсистемы генерации визуалов «Генератор Икон» |

# 2. НАЗНАЧЕНИЕ И ЦЕЛИ СОЗДАНИЯ СИСТЕМЫ

**2.1 Назначение системы**

Инструмент «СДГВМ» используется в целях ускорения создания игровых прототипов и уменьшения денежных затрат на разработку, служит для автоматизации генерации адаптивных виртуальных миров при помощи подключаемых модулей и инструментов включающих нейронные сети, к платформе Unity, реализуемый инструмент включает функционал:

Создание квестов под заданные параметры и характеристики пользователем.

Формирование адаптивной модели поведения ONNX у не игровых персонажей, адаптирующиеся под действия игрока и динамическая реакция на изменения произошедшие в игровом мире.

Синтез ассетов для визуализации иконок квестовых предметов, аватарок персонажей, текстур и тайлов ландшафта виртуального мира.

​Предоставление инструментов настройки и тестировки сгенерированного контента.

​Удобный интерфейс дающий создавать виртуальные игровые миры без знаний платформы Unity.

Автоматизация генерации приводящая к снижению временных затрат и трудовых ресурсов, повышает конкурентоспособность клиентов инди-разработчиков на рынке увеличивая скорость создания игр для монетизации.

Возможность обновлять генератор интегрируя новые инструменты и модули в систему для удовлетворения запросов клиентов и увеличения качества инструмента.

Система позволяет корректировать качество сгенерированных игровых миров для удовлетворения клиента и увеличения пригодности при помощи возможности тестирования генерации контента.

**2.2 Цели создания системы**

Основными целями проекта являются:

− Создание контента при помощи автоматизации ускоряющей скорость разработки прототипов, повышая конкурентоспособность на рынке во время цифровой трансформации и автоматизации;

− Дать доступный инструмент для одиночных разработчиков и малых студий, проект инструмента генерации не стремиться обогнать существующие решения в области автоматизации создания игровых и виртуальных миров;

− Демонстрация интеграции нейросетей для решения академических задач в области направления обучения разработки игровых приложений;

− Задать поток новых идей для развития способностей людей создавать инновационные идеи;

− Решение проблемы реиграбельности в современных играх, создание адаптации в играх к изменениям и действиям игрока, возникновение новых ситуаций должно разжечь в игроках интерес перепройти одну и ту же игру.

Увеличить рентабельность для инди-разработчиков за счет понижения затрат на разработку игрового прототипа.

Привлечение новых студентов на направление разработка игровых приложений при помощи удобного интерфейса и понятных инструкций с обучающими материалами.

Обеспечение качества сгенерированного контента благодаря созданию метрик уровня логической связи в квестах, для соответсвия потребностям пользователей.

# 3 ХАРАКТЕРИСТИКА ОБЪЕКТА АВТОМАТИЗАЦИИ

**3.1 Краткие сведения об объекте автоматизации или ссылки на документы, содержащие такую информацию**

Инструмент «СДГВМ» подлежит модернизации благодаря автоматизации ручного процесса создания виртуальных игровых прототипов, процесс включает в себя автоматизацию создания адаптивных квестов, ассетов для визуализации объектов на игровой сцене, динамического поведения NPC зависящее от событий в игровом мире и действий игрока. Автоматизируемый процесс затрачивает много времени и средств личного капитала разработчиков.

Рассмотрены аналоги «СДГВМ» в виде инструментов и готовых решений генерации виртуальных миров:

− в 2024 год MapMagic 2 пакет из Unity Asset Store, инструмент для автоматизации создания ландшафтов и уровней не содержит в себе генерацию квестов и адаптивного поведения NPC, также нельзя сделать визуализацию игрока и не игровых персонажей;

− в 2019 год AI Dungeon генерирует квесты и текстовые приключения, не способен создавать визуализацию или адаптивные модели поведения не игровых персонажей.

2019 год Dungeon Crawler RPG может генерировать подземелья, и не может квесты и адаптивных NPC.

2011 год Tracery и Ink инструмент генерации квестов, нет прямой интеграции с Unity и другими игровыми движками.

2010 год MiddleVR платформа для генерации миров в виртуальной реальности, генерирует виртуальную реальность в 3D, нет открытого обмена нейронными сетями, нельзя создать адаптивных NPC, нельзя создать адаптивные квесты.

Аналоги демонстрируют не полный функционал инструмента «СДГВМ» реализующий генерацию динамических миров при помощи нейросетей, актуальность разрабатываемого подтверждается отсутствием альтернатив с аналогичным функционалом, реализуемый генератор контента доступен в репозитории: https://github.com/Slavik993/Diplom\_Shebanov\_2026.git.

**3.2 Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды**

Инструмент генерации динамических виртуальных миров адаптирован для персональных рабочих станций с операционной системой Linux, Windows 10-11, MacOS, с установленной платформой Unity, для работы модулей с нейросетями ONNX и LLM-Unity предоставлены технические требования рабочей станции:

Процессор Intel Core i3 (4 ядра, 3.0 ГГц минимум).

Оперативная патмять: 16 ГБ Оперативного запоминающего устройства с частотностью 3200 МГц (DDR4).

Подключение к инету, через Wi-Fi или витую пару.

Технологии и инструменты разработки включают:

LLM-Unity библиотека: позволяет интегрировать большие языковые модели в платформу Unity, необходима для автоматизации создания квестов и диалогов с не игровыми персонажами, для адаптивных виртуальных миров.

ONNX библиотека: снижает нагрузку на систему и ускоряет обработку запросов, используются для обучения адаптивной модели поведения не игровых персонажей.

PyTorch, TensorFlow библиотеки: обучение с подкреплением для создания качественных моделей поведения не игровых персонажей, выполнение межпроцессорных взаимодействий и сложных вычислений.

Запуск сервера через скрипт main.py, генерации визуалов «иконок, аватарок, спрайтов» не возможен без библиотек: numpy, einops, psutil, torch, torchvision, torchaudio, torchsde, av, aiohttp. запуск сервера осуществляется через команду в терминале python -u "c:\Users\Machcreator\Downloads\Applied data analysis tasks\Diplom\_Shebanov\_2026\Diplom\_2026\_Dynamic\_World\_Generation\Assets\ComfyUI-master\main.py" --cp «нельзя использовать просто кнопку запуска в Visual Studio Code», при запуске кнопкой произойдет ошибка, на компьютерах в МУИВ нет графических карт – причина появления ошибки, нужно использовать команду с флагом «cp» – запуск через процессор, без использования видеокарты.

Условия эксплуатации:

LLM-Unity необходим запущенный сервер для постоянной генерации контента и обработки больших данных.

Рекомендуется использовать платформу Unity последней версии.

Преимущества: ускорение создания игровых прототипов, повышение конкурентоспособности, снижение затрат ресурсов.

Ограничения: LLM-Unity необходим запущенный сервер для сложных сценариев.

# 4. ТРЕБОВАНИЯ К СИСТЕМЕ

**4.1 Требования к структуре**

**4.1.1 Описание подсистем динамического генератора «СДГВМ»**

Инструмент «СДГВМ» предназначенный для автоматизации создания игровых прототипов, состоит из подсистем: «Создатель историй» – генерирует квесты для игрока, использует инструмент LLM-Unity на локальной машине. Модели ONNX для управления поведением не игровых персонажей при принятии различных решений игроком и изменений внутри игрового мира - «Система управления не игровыми персонажами». Генерацией иконок предметов, аватарок персонажей занимается подсистема «Генератор икон» используются нейросети из доступных источников или локально расположенных на машине.

«СДГВМ» запускает подсистемы в реальном времени в режиме Unity Play mode, в интерфейсе инструмента можно настроить параметры генерации – сложность, жанр, в текстовых полях будут отображаться сгенерированные квесты и диалоги неигровых персонажей. Пользователь получает доступ к функционалу через проект на платформе Unity в режиме Unity Play mode на главной сцене «Сцена\_Динамического\_Генератора».

Инструмент «СДГВМ» состоит из функциональных подсистем:

1) «Сказитель историй»;

2) «Контролер поведения не игровых персонажей»;

3) «Генератор икон»;

4) «Управление Unity проектом» – inspector, project settings;

5) «Персонализация контента» – интерфейс в режиме Unity Play mode;

6) «Мониторинг качества контента» – для соответствия стандартам;

7) «Сохранение результатов» – сохранение созданных миров;

4.1.2 Разработанный функционал подсистем генератора.

Для выполнения узко специализированных функций генерации динамических виртуальных миров инструмент «СДГВМ» использует подсистемы:

Подсистема «Сказитель историй» исполняет функции генерации квестов с указанием параметров репутации игрока и сложности квеста, использует инструмент LLM-Unity для генерации историй в реальном времени.

Подсистема «Контролер поведения не игровых персонажей» использует модель ONNX для адаптации поведения под квесты и действия пользователя в сцене.

Подсистема «Генератор икон» нужен для визуализации героев, предметов, окружения где происходят события, включает в себя нейросети для генерации картинок и спрайтов.

Подсистема «Управление Unity проектом» изменение параметров физики, камеры, освещения, настройка объектов на сцене, управление общими настройками, скачивание ассетов и новых модулей через Unity магазин ассетов и менеджер пакетов.

Подсистема «Мониторинг качества контента» используется для корректировки генерируемого «СДГВМ» контента, нужен для соблюдения ожиданий пользователя.

Подсистема «Сохранение результатов» предназначена для сохранения полученных результатов генерации «СДГВМ» в виде готовых игровых сцен с наполнением виртуального игрового мира, в виде файлов формата .unity, далее сцену можно экспортировать в другие проекты юнити.

4.1.3 Логические цепочки для входных данных.

На вход подсистеме «Сказитель историй» подается JSON с указанными параметрами в LLM-Unity, информационный обмен обеспечивается через локальный вызов события через скрипт C#, далее происходит генерация текста и парсинг создающий разный выбор и разные цели, потом дерево квеста экспортируется для редактирования в Unity, тем самым обеспечивается синхронизация между работой подсистем.

Подсистема «Генератор икон» использует JSON подающийся на вход локальной нейросети генератора «Stable Diffusion ONNX» c «Unity Sentis» генерирующего изображения, при помощи функции в скрипте сгенерированная картинка преобразуется в спрайт, далее спрайт рисуется на Unity Raw Image или UI Image, сгенерированная картинка сохраняется в ассеты, имея формат «png», при возникновении ошибок генерации будет вывод в консоль Unity.

«Контролер поведения неигровых персонажей» использует на входе XML файл, функция в скрипте использует ONNX для генерации возможных реакций не игровых персонажей, далее в UI обновляются элементы реакции героев.

Все элементы работают напрямую через компоненты в дереве иерархии и скрипты прикрепленные к объектам интерфейса и объектам в сцене, дополнительные требования не нужны для работы с локальным проектом.

4.1.4 Возможность интеграции новых инструментов в «СДГВМ»

Система генерации игровых предусматривает добавление новых инструментов через менеджер пакетов или ассетов в папки внутри открытого проекта или можно использовать Unity магазин ассетов, или Unity менеджер пакетов. Библиотеки созданные пользователями, загружаются через конвейеры, данные для импорта должны быть в поддерживаемом платформой Unity формате - JSON, XML, png, fbx, jpeg, gif, unity, txt. Можно увеличить количество поддерживаемых типов файлов при помощи установки пакетов для совместимости новых форматов.

4.1.5 Виды режимов работ инструмента генерации «СДГВМ»

Реализованы режимы функционирования инструмента генерации адаптивных игровых миров:

Обычная генерация игрового мира;

Тестирование генерации;

Обучающий;

В режиме «Обычная генерация игрового мира» пользователь может указать параметры для генерации игрового прототипа сложность и стиль жанра, режим выбран по умолчанию при запуске сцены «Сцена\_Динамического\_Генератора», пользователь использует инструмент «СДГВМ», через интерфейс в режими Unity play mode.

Тестирование генерации: возможность запустить любое количество генераций, для составления метрик качества логики квестов, иконок и аватарок героев, при завершении тестирования выдается отчет с показом ошибок.

«Обучающий» создан для ознакомления разработчиков с системой, показывает пример сгенерированного виртуального мира и доступ к текстовым обучающем материалам содержащим инструкцию по настройке параметров квеста, или импорта спрайтов, режим активируется через кнопку «Помощь».

4.1.5.1 Возможности Unity Console для диагностирование и логирования ошибок

Компоненты подсистем, должны предоставлять функционал для генерации квестов, модели поведения не игровых персонажей, иконок квестовых предметов и аватаров персонажей, инстумент «СДГВМ» включает элементы интерфейса для просмотра генерируемых квестов и диалогов NPC, в консоль Unity показываются дебагом диагностические события.

В программе создана система записи диагностической информации при возникновении ошибок генерации контента, сведения о процессе и параметры генерации будут записаны в .log файл в папке «error\_logs».

4.1.5.2 Дальнейшая эволюция и перспективы системы генерации «СДГВМ»

Система динамической генерации виртуальных миров создан на платформе Unity следственно может модернизироваться через Unity менеджер пакетов, для инструмента необходима возможность обновлять модели ONNX и LLM-Unity для повышения качества генерируемого контента, предусмотрена возможность реализации задачи на вырост и добавление 3D генерации миров и интеграции сетевой игры через особые пакеты и создание сервера для сетевых взаимодействий. Масштабируемость в проекте реализована через модульную архитектуру позволяющей подсоединить к инструменту генерации новый функционал через плагины устанавливаемые в Unity менеджер пакетов, для масштабирования не нужно перерабатывать ядра системы.

4.1.5.3 Обзор набора исполняющего персонала для оперирования «СДГВМ»

Нужен один пользователь для работы с запущенной системой генерации на одной рабочей машине, система не включает специфичных компонентов требующих специализированного персонала. Запуск и настройку Unity выполняет сам разработчик. Для оперирования инструментом «СДГВМ» используется UI с элементами на канвасе, появляющегося при переходе в режим тестирования сцены Unity Play Mode, дальше пользователи взаимодействуют с настройками генерации – сложность и жанр для квестов, промт для икон и аватарок, с слайдерами для настройки уровня интеллекта не игровых персонажей.

4.1.6 Необходимые умения и знания для персонала генерирующего при помощи «СДГВМ» контент

В результате анализа возможных взаимодействий пользователя с «СДГВМ» выявлено, для работы необходимы знания, основных понятий Unity – дерево иерархии, пользование режимом Unity-Play mode для тестирования сгенерированных виртуальных миров, нужно понимать обучение ONNX происходит через цифровые сигналы, а не через промты – нужно для оценки и составления метрик определяющих качество поведения не игровых персонажей. Нужно уметь настраивать объект UI – Canvas для настройки интерфейса диалогового окна, или окна выбора действия, предъявляются требования к знанию языка C#.

Для работы с «СДГВМ» требуется ознакомление с документацией Unity, и рассмотрением обучающего режима с материалами, инструмент предполагает доступность пользования и требует базовых навыков работы с платформой Unity требующей основной документации. Рекомендуется уметь работать с терминалом Git Bash, для сохранения и фиксации изменений.

4.1.7 Данные о количественных и качественных показателях «СДГВМ» для платформы Unity

Были получены средние показатели по количественных и качественных величинах связанных с генерацией контента:

Таблица 1.1.7. Статистические данные о качественных показателях СДГВМ.

|  |  |  |
| --- | --- | --- |
| Показатель | Средняя величина | Пиковая величина |
| Время обработки одного запроса на генерацию | 1 минута 45 секунд | 5 минут 51 секунда |
| Время загрузки сцены с сгенерированным миром в Unity Play Mode | 3,71 секунда | 8 секунд |
| Квесты/час | 40 | 62 |
| Икон/час | 70 | 125 |
| Качество логической связи в контенте | 84,43% | 98,2% |

«СДГВМ» при минимальных 16 гигабайт памяти, даёт результаты указанные выше, время обработки запросов может быть лучше при использовании графической карты.

4.1.8 Решение проблем с случаями сбоев на локальной машине генерации инструмента «СДГВМ»

При возникновении сбоев в «СДГВМ» при генерации контента, появляются выводы в консоль Unity, с указанием типа произошедшей ошибки, в системе предусмотрена система сохранения изменений на сцене встроенная в Unity, рекомендуется использовать Git, для фиксирования изменений в ветке соответствующей разработчику, также система предусматривает подключения RAID – массивов и других средств резервного копирования для защиты информации при уничтожении или повреждения накопителя на основной локальной машине, для сохранения изменений сделанных в проекте. Инструмент «СДГВМ» способен заменять сломавшиеся изображения на шаблон, не останавливая Unity Play Mode, вызывая оповещение об ошибки в консоль и уточнении о использовании шаблона.

4.1.9 Безопасность и защита генерируемого контента инструментом «СДГВМ»

При запуске билда с готовым сгенерированным миром пользователем купившим игру сделанную при помощи инструмента «СДГВМ» - спрайты, квесты, неигровые персонажи будут зашифрованы и скрыты от пользователя благодаря системе защиты контента с использованием шифрования алгоритма AES-256. Шифрование применяется на все ассеты и сцены, их нельзя будет использовать или забрать себе, библиотека Unity System.Security.Cryptography используется для шифрования.

4.1.10 Требования к визуальному дизайну, цветовой схеме и эстетике UI

Визуальный дизайн графического интерфейса инструмента генерации контента «СДГВМ» разработан с учетом интуитивного способностей восприятия разных тестировщиков, было выявлено: панели настроек не должны быть перегружены различными элементами, было минимизировано количество объектов UI на канвас, были проведены тесты понятности и удобства интерфейса для соответстввия заявленным стандартам для низкого порога вхождения для пользования системой генерацией «СДГВМ». Цвета для оформления подобраны в соответствии с ориентированностью на взаимосвязь логики элемента UI с исполняющей функцией, например текстовый объект возникающий при успешной генерации покрашенный в зеленый цвет, для снижения утомляемости и когнитивной нагрузки, а также для лучшей видимости элементов, подобраны успокаивающие цвета бледных оттенков и серого цвета.

4.1.11 Совместимость инструмента генерации «СДГВМ» с другими платформами и OS

Инструмент «СДГВМ» работает на следующих операционных системах: Android, Windows, Ubuntu, iOS, Linux, можно установить платформу Unity на все ранее перечисленные операционные системы, сгенерированный инструментом контент может быть импортирован в другие игровые движки: Construct 3, Godot, Unreal Engine, GDevelop.

4.1.12 «СДГВМ» – локальная работа в Unity

Система динамической генерации виртуальных миров адаптирована к локальной работе при полном отключении интернета, но требуется использовать заранее загруженные модели для генерации диалогов, квестов, икон и аватарок, минимальные требования запуска системы генерации на локальной машине наличие двух плашек памяти по восемь гигабайт, видеокарта рекомендуется к использованию для увеличению скорости генерации контента. Необходимо периодично проводить проверки файлов моделей для локальной генерации, нужно обновлять пакеты LLM-Unity, Sentis, Newtonsoft.Json , Stable Diffusion Unity, ML-Agents, 2D Sprite, ML-Agents. Хранение сгенерированного контента происходит в папке ассеты локального проекта Unity, происходит шифрование контента при создании билда, оперирование инструмнтом генерации проходит без участия специальных ролей – сетевых администраторов, безопасников, администраторов, тестировщиков и разработчиков.

4.1.13 Сохранение творческого потенциала сгенерированного контента

Необходимо защитить уникальность сгенерированного контента, путем выдачи цифрового отпечатка для защиты идей автора и предотвращения использования идей путем отслеживания источника происхождения контента.

Доступ к расшифровке сгенерированного «СДГВМ» осуществляется разработчиком создавшим контент.

Сгенерированный контент в виде ассетов будет скрыт в файлах билда для защиты авторской идеи от использования и модернизации.

4.1.14 Осуществление сохранения сгенерированного контента в случаях технических неисправностей

Благодаря реализации скрипта автосохранения на C# будет происходить фиксация изменений сгенерированного контента в сцене, сохранение происходит раз в три минуты и при отключении электричества можно вернуться к моменту откуда закончил, запись происходит в папку «AutoSaveVault» в ассетах.

4.1.15 Защита локальной машины с «СДГВМ» от воздействий с внешней стороны

Рекомендуется использовать источник питания с буфером по минимизации скачков напряжения, лучше минимизировать возможные риски связанные с физическими воздействиями – тряской, вибрациями, также нужно соблюсти минимальные требования к запуску системы «СДГВМ».

4.1.16 Патентование уникальных алгоритмов генерации при помощи авторского свидетельства

Использование «СДГВМ» пользователями должно осуществляться с соответствием лицензионного соглашения принимаемого при получении инструмента генерации контента, решения реализующие функционал «СДГВМ» защищаются авторским правом согласно четвертому разделу конституции Российской Федерации федеральному закону номер двести тридцать, сто сорок девятому и сто пятьдесят второму, а также статье тысяча двести пятьдесят девятой. Пользовательское соглашение включает в себя распространение и использование кода системы генерации в коммерческих целях, на основе данных Роспатента алгоритмы генерации «СДГВМ» оригинальны и пока не имеется аналогов.

4.1.17 Важные факторы совместимости и стандартизации «СДГВМ»

Документирование по типу шаблонов JSON и функционирующее C# скрипты записываются в Git Hub, для независимого пользования без прямого обращения к автору, при разработке использованы последние версии Unity и стабильные версии установленных пакетов, для обеспечения максимальной совместимости, шаблоны JSON создавались в соответствии с общими стандартами, для совместимости с разными игровыми движками, инструмент может быть расширен новыми модулями для генерации объемного ландшафта «Map Magic 2» и модулями с генерацией объемных моделей персонажей с анимациями и объемными предметами вместо иконок для квестов.

4.1.18 Необходимый для удобства пользования функционал системы генерации «СДГВМ»

Возможность принимать JSON шаблоны созданные пользователями для реализации новых типов квестов и диалогов, экспорт отчетов по генерации для обнаружения проблем по генерации, наличие обучающих материалов прямо в проекте для решения возникающих проблем по не знанию основ принципов и механизмов работы «СДГВМ», умная локализация интерфейса на различные языки через пакет Unity «Localisation», возможность писать обратную связь для решения недостатков инструмента генерации «СДГВМ».

4.2 Требования к функциям (задачам), выполняемым системой

Система динамической генерации игровых миров должна реализовывать генерацию образовательных квестов вместе с диалоговыми деревьями с выбором варианта взаимодействия или общения с не игровыми персонажами, также «СДГВМ» способна создавать визуалы в виде иконок, спрайтов и аватарок, для углубленного погружение в игровое пространство созданного виртуального мира. «СДГВМ» создана, для обогащения знаний о национальной культуре России через выполнение разных видов квестов, заданий сгенерированных системой динамической генерации виртуальных миров.

**4.2.1 Модульная подсистема «СДГВМ» для генерации контента**

Управляемая клиентом подсистема предназначена для генерации, редактирования, отображения игрового контента в сцене при запуске режима Unity Play Mode в интерфейсе инструмента, генерация происходит за счет межнейросетевого взаимодействия реализованного через ONNX, система работает без подключения интернета за счет установленных локально моделей.

**4.2.1.1 Блок «Сказитель историй» генерация квестов**

Обеспечивает создание квестов через шаблоны JSON, настраиваемые пользователем – разработчиком, по параметрам: жанр, сложность можно настраивать квест, созданный квест в виде текста показывается в Text Mesh Pro объекте, есть поля ввода – «input field» объект для ввода заголовка - «название квеста», поле ввода для аннотации – краткого описания, поле ввода самого квеста, объект Raw-Image для вставки визуалального описания квеста.

4.2.1.2 Блок «Контроллер поведения не игровых персонажей» генерация диалогов

За обучение и контроль поведения NPC отвечает блок «Контроллер поведения» использующий обученную локальную ONNX для превращение JSON файлов с параметрами в адаптивную реакцию не игрового персонажа на действия игрока, можно установить тип поведения - «враждебный, нейтральный, дружественный, стремящийся обмануть, желающий получить выгоду», есть объект Text Mesh Pro с выводом сгенерированного диалога для ознакомления.

4.2.1.2.1 Математический анализ контролера поведения Mistral 7B – Instruct

Выбранная модель имеет ряд преимуществ над альтернативными вариантами: формат GGUF уменьшил размер модели с четырнадцати Гигабайт, до пяти Гигабайт, также уменьшилось количество требуемой оперативной памяти до восьми гигабайт оперативной памяти, минимум требуется четыре с половиной Гигабайт, модель «Контролера поведения не игровых персонажей» является трансформером, использует преобразует биграммы из JSON в эмбеддинги – числовые вектора для численного сравнения различия смысла слов, это позволяет оптимизировать модель для специфических диалогов, кросс-энтропия используется в качестве функции потерь, а головы важности ищут итерационно нужное слово итерационно генерируя диалог. Оценка качества модели оценивается метриками BLEU по формуле:

BLEU = BP \* exp() где:

– точность биграмм и n грамм

=

– счетчик совпадений, сверяющий с эталоном количество совпадений в тексте

– допустимый максимум количества n грамм

Веса для установления порядка n грамм

= ; N – максимальный порядок

BP – штраф за короткий текст

BP = ;

c – длина сгенерированного текста

r – длина эталонного текста

Для минимизации функции потерь, использована кросс энтропия:

H(p,q) = - ) + (1 - )]

Обозначения для переменных:

– истинная метка

– предсказанная вероятность

N – количество примеров

Эмбеддинги преобразованы из биграмм по формуле:

=

– эмбеддинг слова,

T – размер последовательности

- скрытое состояние на шаге (t),

W – матрица весов модели

Формула расчета скорости генерации одного токена в секунду:

E =

– время

–токенов всего

Подтвердить полезность квантизации модели Mistral 7B – Instruct способна формула для вычисления размерности модели при сжатии:

Compressed factor = = 2,8

Качество машинного перевода измеряется метрикой «Metric of Evaluation of Translation with Explicit ORdering»

METEOR = \* (1 - ) ; Переменные формулы рассчитываются по формулам:

= , Точность – P, Полнота – Recall;

= \* (, штраф за неправильный порядок

Расчет сложности модели проводится по формуле «Perplexity»

PP(W) = exp( - |))

N – количество токенов

W – последовательность токенов

Вероятность токена P (|)

4.2.1.2.2 Обоснование выбора языковой модели для генерации квестов и диалогов

Модель Mistral 7B Instruct имеет ряд преимуществ, над аналогами: эффективное потребление ресурсов локальной машины, благодаря квантизации ускоряется скорость обработки запросов. Модель имеет большую точность, скорость и логическую связанность сгенерированного текста, по сравнению с аналогами Llama, Grok, GPT-Chat, BERT. Модель преобразовывает JSON с тегами являющимися классами в числовые вектора, а благодаря головам важности определяется следующее слово на новом шаге.

4.2.1.2.3 Внутренее строение Mistral 7B - Instruct

Структура модели сделана из тридцати двух слоев трансформеров, благодаря четырём тысячам девяносто шести токенам возможно создание сложных диалоговых и квестовых деревьев с помощью числовых векторов большего размера. Вычислительная нагрузка оптимизирована благодаря квантизации модели в формат gguf. Входная обработка происходит в преобразовании n-грамм в токены являющиеся классами, трансформерный блок - «feed-forward network FFN», отвечает за преобразование n-грамм в эмбеддинги используя слои внимания, FNN рассчитывается по формуле:

= \* x + , является числовым вектором, матрица весов, – смещение, x – входной класс, небходимо для создания логичных квестов и диалогов в системе динамической генерации виртуальных миров, правильная последовательность расстановки слов имеет ключевое значение, далее рассмотрена формула для работы слоев внимания:

Attention(Q, K, V) = softmax() \* V, в формуле – размер ключей, Q, V, K являются ключами и запросами вычисляющимися по формуле: Q = \*h, V = \* h, K = \* h, h является входным числовым вектором слоя, слои внимания нужны для введения диалога в долгосрочной перспективе не теряя логической связанности, слои внимания включают feed forward network, для нелинейного преобразования:

FFN(x) = max(0, x+) + , переменные в формуле означают: , – смещения весов, , – матрицы весов, необходимо для поиска зависимостей между словами и использования сложных шаблонов, используя флаг запуска –cpu на компьютерах без видеокарт ускоряется генерация и предотвращается переобучение. Генерация следующих классов происходит через softmax в линейном слое, по формуле: P() = , в формуле где – исходные ненормированные оценки, вычисляются по формуле: z = h + , является смещением, – выходная матрица, исходные ненормированные оценки помогают оптимизировать диалоги для повышения реализма.

Обучение происходит по кросс – энтропийному лоссу :

H(p,q) = - ) + (1 - ))], в формуле переменная истинная метка , предсказание. Необходимо для дообучения модели на инструкциях, улучшая качество понимания промтов.

4.2.1.3 Блок «Генератор икон» генерация визуальных активов

Использование локальной модели Stable-Diffusion пакетом Unity Sprite Generator, для генерации иконок предметов и аватаров персонажей, на вход модели приходит JSON с параметрами: стиль, окружение, размер. Пользователь-разработчик взаимодействует с интерфейсом при запуске Unity Play Mode, с объектами input field для названия, стиля, разрешения, загрузки текстур.

4.2.1.3.1 Реализация генерации иконок и спрайтов через Comfy UI

Локальный сервер запускается через скрипт main, при запуске появляется хост адрес [http://127.0.0.1](http://127.0.0.1" \o "http://127.0.0.1):8188, при переходе на адрес появляется вкладка ComfyUI в браузере, рабочее пространство ComfyUI позволяет создавать шаблоны для генерации разного типа визуального контента - 3D модели, видео, картинки, спрайты, можно использовать готовый шаблон с настроенными параметрами для нужного типа генерации, ComfyUI использует установленные в проекте модели для генерации 2D изображений sd-turbo и sd3-medium, для генерации ComfyUI использует токенизатор для генерации скрытых факторов декодирующихся в изображение, при завершении генерации можно сохранить изображение имеющее png формат, в папку «Exports» подпапку «Generated\_Visuals».

4.2.1.3.2 Обоснование выбора моделей sd turbo и sd3 medium, для генерации контента

Модели имеют низкое потребление вычислительных ресурсов локальной машины, возможности гибкой настройки позволяют генерировать изображение высокого качества, Stable Diffusion Turbo используется для быстрой генерации визуалов, адаптирована к простым и низкокачественным иконкам, спрайтам, благодаря минимизации артефактов, для генерации не нужно подключение к интернету, модели загружены локально в проект, для работы с ними нужно запустить main.py локальный сервер, без запуска локального сервера генерация не возможна. Генерация тяжелых спрайтов и иконок должна происходить через sd3-medium, нужно или в шаблоне при запуске ComfyUI назначить в первом блоке выбора модели, по умолчанию стоит sd-turbo, отличие от sd-turbo в количестве шагов пятьдесят штук, у turbo десять шагов, высокая детализация необходима для ситуаций где требуется глубокое погружение в созданный виртуальный мир, для этого необходимо высокое качество визуализируемых персонажей, на предметы можно использовать sd-turbo, модели легко интегрируются в Unity и способны работать синхронно, благодаря ONNX и маршрутизации JSON между моделями, через взаимные запросы.

4.2.1.4 Блок «Самоучитель с обратной связью» обучающие материалы, фидбэк

Обучающие материалы расположены в ассетах папка «Tutorial», необходимо получать фидбэк от разработчиков позволяя написать описание проблем и загрузить в виде лог файла в git, обращение происходит поле ввода в графическом интерфейсе, сохранение лога происходит в папку «User\_feedback». Оставлено упоминание о необходимости обновлять установленные пакеты в проекте для минимизации ошибок, также есть ссылка на Git с документацией, во время запуска Unity Play Mode в интерфейсе есть кнопка «Помощь» при нажатии которой появляется PDF с краткой инструкцией по основным шагам генерации.

4.2.1.5 Блок «Инструмент для автоматического создания интерфейса «СДГВМ»»

Через Editor, был добавлен скрипт добавляющий новый инструмент, в раздел Tools, нажатием кнопки «Generate» в инструментах открывается выпадающий список доступных инструментов автоматического создания объектов в сцене, нужно выбрать «Storytelling UI Layout» – при нажатии выведется сообщение «UI добавлен в сцену», рекомендация перед любым изменением скрипта генерации интерфейса, нужно старый сохранять в Prefab, и удалять полностью со сцены старый интерфейс, по причине наложения друг на друга, и возникновений ошибок с отображением. Инструмент создающий интерфейс создаёт три основных кнопки для открытия интерфейсов подсистем генерации «Сказителя историй», «Контролера поведения не игровых персонажей», « Генератора икон». В интерфейсе подсистемы «Контролер поведения не игровых персонажей» создаются следующие компоненты в иерархии: поле ввода «Имя персонажа»; выпадающий список «Отношение к игроку»; поле ввода «Реакция»; выпадающий список «Эмоция»; поле ввода «Окружение»; кнопка «Сгенерировать диалог». Компоненты интерфейса подсистемы «Сказитель историй»: Input\_field «Тема истории»; выпадающий список «Жанр повествования» – например: постапокалипсис, фентези, реализм, приключение, киберпанк, стимпанк, сказочный, драма; Input\_field «Длина истории», кнопка «Сгенерировать историю», текстовое поле «Текст истории». Интерфейс подсистемы «Генератор икон»: «Описание иконки» - Input\_field, «Стиль иконки» – выпадающий список, Input\_field «размер иконки», «Сгенерировать иконку - кнопка», TMP текст «Статус генерации».

**4.2.1.5 Блок «Автоматизированное тестирование» проведение нескольких генераций с выводом графиков и рекомендаций**

«СДГВМ» использует Test Framework, для проверки логики в диалогах и оценки качества визуалов, блок при работе выдает отчет с указанием ошибок и недочетов, рекомендации в отчете говорят что необходимо исправить для увеличения качества генерируемого контента.

4.2.2 Разработка Модульной подсистемы «СДГВМ» для генерации игрового контента

Оперируемая одним человеком «СДВГМ» в режиме Unity Play Mode, модульная подсистема с разделением на генерацию разных типов контента на текст адаптивных диалогов, квестов и визуалов в виде иконок и аватарок, во время генерации происходит общение между нейросетями и благодаря межнейросетевому взаимодействию между всеми типами контента существует логическая связь, модели установлены локально не требуется подключение к интернету для генерации по API, нужно запускать сцену «Сцена\_Динамического\_Генератора», сгенерированный контент можно увидеть в реальном времени.

**4.2.3 Разработка подсистемы «СДГВМ» для хранения классификации и поиска контента**

При работе с сгенерированными ассетами, возникли проблемы с поиском и классификацией контента, для решения проблемы было использовано разделение контента на отдельные папки в ассетах по типу, папка «Generated\_Quests» для квестов, «Dialogues» – для диалогов, «Item Icons» – для иконок предметов, «Characters\_Avatars» – аватарки героя игрока и не игровых персонажей, поиск необходимого ассета происходит через поисковик встроенный в Unity.

**4.2.4 Разработка подсистемы «СДГВМ» для оценки производительности**

В результате углубленного анализа, были выявлены критические недостатки связанные с потреблением памяти при синхронной генерации контента моделями Mistral 7B-Instruct, sd-turbo, sd3-medium, случается не хватка операционной памяти, приводящая к вылету Unity Editor и появлению окна Unity Crash Report, для снижения вероятности вылета при синхронной генерации контента были применены метрики «Использования оперативной памяти», «Нагрузки на процессор», «Скорость записи и чтения с диска». Формулы представлены в следующем виде: «Использование оперативной памяти RAM»

= , в формуле выделенная операционная память системой под Юнити в битах -Profiler.GetTotalReservedMemoryLong(), выполняющиеся процессы в данный момент, расход на оперативную память в битах. Решение проблемы в возможности отдельной генерации каждого компонента виртуального игрового пространства отдельно с задействованием одной модели за раз, очередь делает процесс генерации медленнее, но без вылетов работы программы.

= \* 100, Enviroment.TickCount – время работы “СДГВМ” начиная момента запуска в “мс”

n – количество ядер в процессоре,

Время работы отдельного взятого ядра в “мс” – TotalProcessorTime[i].

Формула «Скорости чтения с диска» оформлена в формате:   
 = , Размер файла в битах – FileSize[i], TimeToWrite[i] – время записи в секундах, количество файлов - k. Решение проблемы с медленной загрузкой файлов интегрировано методами SaveStory(), SaveIcon(), SaveNPCDialogue(), буфер можно увеличить по формуле: = min(16384,), доступная память вычисляется = 8 гигабайт «8192 Мегабайта» - .

**4.2.5 Разработка подсистемы «СДГВМ» контроля модулей генерации**

Редактирование и дополнительная настройка модулей генерации контента «СДГВМ» происходит за счёт подсистемы администрирования модулями и пакетами Unity, настройка моделей для генерации контента происходит в интерфейсе при входе в Unity Play Mode для модулей существуют уникальные индивидуальные настройки.

Настройки «Сказителя историй» состоят из:

поля ввода с заголовком – название квеста;

поля ввода с аннотацией;

Text Mesh Pro – содержание квеста;

выпадающий список с жанрами;

выпадающий список c сложностью;

Raw Image для визуального описания квеста.

Настройки «Контроллера поведения неигровых персонажей» состоят из:

Поля ввода наименования персонажа;

Выпадающий список отношения NPC к игроку «дружелюбный, нейтральный, враждебный»;

Text Mesh Pro – для отображения текста диалога;

Выпадающий список – выбор эмоциональной окраски диалога;

Поле ввода для назначения уровня соответствия от нуля до ста, определяет реакцию на действия игрового персонажа.

Настройки «Генератора икон» включают в себя:

Поле ввода – название иконки, аватарки;

Поле ввода – выбор окружения;

Выпадающий список – разрешение изображения;

Выпадающий список – выбор фильтра отображения;

Поле ввода - загрузка собственного изображения.

**4.2.6 Разработка подсистемы «СДГВМ» тестирования генерации**

Необходимые функции – вывод производительности в виде графиков и диаграмм, создание файлов отчетности по возникшим ошибкам с рекомендациями о возможных решениях, возможность запустить цепочку из генераций для сбора данных необходимых для ведения аналитики критических проблем, оперирование подсистемой происходит через графический интерфейс при запуске Unity Play Mode состоящий из:

Канвас с объектом рисования линии – отображение графиков;

Text Mesh Pro с отображением частоты процессора и количеством используемых ядер;

Кнопка «Создать отчет» – генерация отчета, с рекомендациями по решению проблем, в отчете описываются возникшие во время генерации ошибки;

Кнопка «Экспортировать отчет» – создание отчета в PDF или JSON, для дальнейшей обработки;

Text Mesh Pro с выводом состояния проведения теста.

Подсистема продумана для дальнейшего расширения и можно добавить тесты новых типов контента, 3D моделей, ландшафта, музыки, звуков, анимаций, эффектов.

4.3 Требования к видам обеспечения

4.3.1 Требования к математическому обеспечению системы

Математические алгоритмы реализуются в ONNX встроенные в Unity LLM-Unity, Stable Diffusion, не требует дополнительного математического обеспечения.

4.3.2 Требования к информационному обеспечению системы

Структура данных системы динамической генерации виртуальных миров:

Формат хранения: JSON, HTML, log, txt, png, jpeg, fbx форматы совместимые с системой хранения платформы Unity, дополнительно можно добавить пакеты, для увеличения возможных форматов для использования.

Метаданные: тэги в JSON файле в виде параметров. В JSON подающийся на входе ONNX модели есть тэги – параметры для настройки генерации пример входных данных в JSON: { «ДействиеИгрока»: тэг - «отказаться», «Статус не игрового персонажа»: тэг - «нейтральный», «Контекст»: {«Местоположение»: тэг - «Таверна», «Отношение»: тэг - «странник»}, пример выходного файла JSON: {«Диалог»: тэг - «Ты пожалеешь!», «Действие»: тэг - «Нападение», «Эмоция»: тэг - «Злость», «Визуал»: тэг - «Угрожающая поза с перекошенным лицом»}, нейросети используют тэги для генерации текста подходящего по смыслу.

Сгенерированный игровой контент: квесты, диалоги, иконки, аватарки.

«СДГВМ» использует технологию файловой системы Unity для хранения ассетов, таким образом обеспечивать самостоятельность в хранении файлов системы генерации «СДГВМ» не нужно подключать дополнительные системы хранения, что уменьшает порог входа пользования системой для инди-студий и соло разработчиков.

В системе генерации есть логирование в виде отчетов, есть экспорт отчетов в JSON или PDF, система включает статус проведения тестирования, отчет об ошибках, рекомендации по решению.

Доступ к системе предоставлен одному человеку, ролей и авторизации не требуется.

4.3.3 Требования к лингвистическому обеспечению системы

Необходима локализация для расширения целевой аудитории способной использовать «СДГВМ», необходимо соблюсти языковые особенности склонений и времен для различных языков, копки именуются глаголами в прошедшем времени. Работа с «СДГВМ» требует знание C#, для редактирования кода и внесения нового функционала. Необходима унификация терминов для увеличение восприятия интерфейса для пользователя. Русский язык в «СДГВМ» очень «Требовательный» к склонениям, перевод с модели изначально будет ломать склонения и коверкать множественное число. Интуитивность восприятия пользователями функционала кнопок повышается путём использования глаголов. Перевод кнопок осуществляется через пакет Unity «Localisation», используется умная локализация подставляющая текст в кнопки специальным выпадающим списком добавляющимся установленным пакетом в сцене при запуске в правом углу экрана.

4.3.4 Требования к программному обеспечению системы

«СДГВМ» может быть установлена на множество операционных систем: Linux, Ubuntu 24.04, macOS X 11+, Windows 10 - 11, iOS, Android.

Платформа Unity должна быть версии 6000.1.1f1 для повышения совместимости, необходимос установить в проекте пакеты «LLM-Unity», «Unity Sprite Generator», также нужны установленные локально в проект модели: mistral-7b-instruct-v0.2.Q4\_K\_M.gguf, для работы с моделями в Unity нужно использовать библиотеку llama.cpp, запуск генерации требует создания локального сервера в проекте для запуска и работы модели, для настройки сервера в инспекторе нужно назначить скрипт LLM объекту сервера и в поле LLM-model переносим модель из локального хранилища проекта «из StreamingAssets – models». Редактирование кода происходит в среде разработки Visual Studio Code, для управления и контроля версий рекомендуется использовать Git Bash в режиме редактирования файлов notepad, для отправки больших файлов в коммиты использовать Git-lfs.

4.3.5 Требования к техническому обеспечению

Unity для работы с моделями ONNX требует шестнадцать гигабайт памяти минимум, нужен процессор пятого поколения рекомендуется использовать Ryzen 7000 series или Intel Core 13 «Raptor Lake», 4 ядра минимум 2,5 ГГц, необходимо иметь 50 Гб свободного места на жестком диске, 10 Гб весит платформа Unity, модели вместе с пакетами около 30 Гб, ещё 10 Гб на проект и хранение сохраненного контента. Видеокарта не обязательна но рекомендуется использовать для ускорения скорости генерации контента. Также необходимо развернуть локальный сервер обрабатывающий запросы генераций с развернутой моделью требующей около 10 Гб оперативной памяти.

4.3.6 Требования к метрологическому обеспечению

Для улучшения качества генерации диалогов необходимо уменьшить параметр температура для LLM модели и количество токенов должно быть около двухсот, параметр «Повторное исправление» должен быть установлен на 1,15. Для предотвращения возникновения ошибок при генерации нужно не допускать полного использования оперативной памяти, во время длительного сеанса игры могут возникнуть проблемы с количеством доступных ресурсов.

4.3.7 Требования к организационному обеспечению

Персонал для обслуживания «СДГВМ» не требует специальных ролей, для минимизации входного порога пользования соло разработчикам и состоит из одного рабочего места, работник оперирующий «СДГВМ» должен обеспечивать администрирование моделей, управлять локальным сервером для обработки генераций, также работник осуществляет модернизацию и разработку скриптов C# для добавления нового функционала, работник обязан обновлять пакеты Unity до последней версии для исключения проблем совместимости. После проведения тестирования на подопытных, выявлен перечень умений для работника с «СДГВМ»: необходимо иметь основные знания о архитектуре проекта юнити, разбираться с встроенной системой поиска файлов, уметь работать с инспектором и настраивать через него параметры модели для генерации контента, для разработки нового функционала, необходимо знать необходимые пакеты юнити необходимые для интеграции LLM и генераторов 2D спрайтов и уметь работать с пакетным менеджером юнити, разработка и поддержка скриптов функционала «СДГВМ» осуществляется через C# следовательно необходимо уметь работать с встроенными библиотеками юнити, и разбираться в индивидуальных особенностях языка C#, также операторы «СДГВМ» должны ознакомиться с обучающими материалами, находящимися локально в проекте в папке «Обучающие материалы» в Assets.

4.3.8 Требования к методическому обеспечению

Для работников «СДГВМ» разработаны обучающие материалы в Assets «Обучающие материалы», в обучающих материалах лежат PDF с инструкцией по эксплуатации «СДГВМ» и инструкцией по настройке локального сервера, настройки параметров модели в инспекторе, также сделана PDF с объяснением математики и описания взаимодействий ONNX моделей, в обучающих материалах показаны примеры входных и выходных данных, для объяснения принципов работы и создания понимания принципов работы «СДГВМ» у работников. На основе опытов, были созданы рекомендации для генерации с готовыми шаблонами и советами по настройке моделей, также созданы рекомендации по устранению ошибок генерации, приведены примеры шаблонных ошибок и пути решения, например очищение памяти при длительной сессии генерации.

# **5. СОСТАВ И СОДЕРЖАНИЕ РАБОТ ПО СОЗДАНИЮ СИСТЕМЫ**

Таблица 2. Состав и содержание работ по созданию системы.

|  |  |  |  |
| --- | --- | --- | --- |
| № наименование этапа | Наименование работ, входящих в состав этапа | Сроки  выполнения  этапа работ (в календарных днях) | Отчетная  документация  исполнителя |
| 1 этап. Разработка ТЗ для «СДГВМ», написание статей по модулям «СДВГМ», создание первой главы ВКР | Разработка технического задания на разработку системы динамического генератора виртуальных миров, исследование инструментов генерации контента – нахождение лучших вариантов для «СДГВМ» написание статей по исследованиям, на основе выбранных инструментов составляются диаграммы первой главы ВКР и пишется первая глава ВКР. | С 17.09.25 по  28.10.25 | Техническое задание «СДГВМ» ГОСТ 34.602-89  и ГОСТ 19.201-78,  Отчет по исследованиям в области инструментов генерации ГОСТ 7.1 - 2003,  Первая глава ВКР «Динамическая генерация виртуальных миров при помощи нейросетей» в соответствии ГОСТ Р 7.0.11 - 2011 |
| 2 этап.  Создание архитектуры «СДГВМ» в проекте юнити, тестирование генерации квестов, диалогов и икон, создание локализации для интерфейса | Создание объектов интерфейса, объектов для визуализации сгенерированного контента, нахождение оптимальный параметров модели для генерации и повышение качества контента, подключение пакета Unity Localization, для перевода текста на кнопках и перевода унифицированных элементов интерфейса | С 29.10.25 по  25.11.25 | Создание инфраструктуры в проекте Unity ГОСТ 19.401-78,  Улучшение сгенерированного контента «икон, диалогов, квестов» ГОСТ 19.301-79,  Установление пакетов локализации и перевод элементов интерфейса ГОСТ 7.75-97 и ГОСТ 19.505-79 |
| 3 этап. Сборка, интеграция, тестирование | Создание межнейросетевого общения через ONNX, проведение тестов по генерации и исправление возникающих ошибок | с 26.11.25 по 15.12.25 | Тестирование и исправление ошибок  ГОСТ 7.0.97 и ГОСТ Р ИСО/МЭК 12207-2010  Работа с ONNX LLM-Unity ГОСТ 19.301-79 |
| 4 этап. Оптимизация работы «СДГВМ», перед внедрением на локальные машины МУИВ | Оптимизация производительности моделей ONNX, написание второй главы ВКР «Динамическая генерация игровых миров при помощи нейросетей» | с 16.12.25 по 27.01.26 | Написание второй главы ГОСТ 7.0.11-2011  Оптимизация производительности моделей ONNX генерирующих контент ГОСТ 7.32-2017 |
| 5 этап. Внедрение и эксплуатация конструктора квестов “СДГВМ” | Внедрение «СДГВМ» на компьютеры МУИВ Тестирование системы динамической генерации виртуальных миров, на локальных машинах вуза | С 28.01.26 по 15.03.26 | Внедрение на компьютеры ГОСТ 34.603-92 Тестирование производительности на компьютерах МУИВ ГОСТ 7.0.97-2016 |
| 6 этап. Эксплуатационная практика. | Добавление нового функционала в «СДГВМ», дообучение модели на датасетах с квестами, общее улучшение качества и оптимизация | С 16.03.26 по 17.05.26 | Реализация нового функционала ГОСТ 19.201-78 Дообучение модели Mistral-7B-Instruct на датасетах с квестах ГОСТ 34.602-89 Улучшение качества контента и оптимизация системы ГОСТ 7.0.97-2016 |
| 7 этап. Защита ВКР «Динамическая генерация игровых миров при помощи нейросетей» | Разработка списка источников и литературы, полное оформление текста в ВКР, подготовка к отчету и защите ВКР, введение жизненного цикла и поддержка «СДГВМ» | с 02.06.26 по 8.06.26 | Редактирование текста ВКР ГОСТ 7.0.11-2011,  Защита отчета ГОСТ 7.32-2017,  Введение жизненного цикла и поддержка «СДГВМ» ГОСТ Р ИСО/МЭК 12207-2010 |

# 6 РАЗРАБОТКА ПРОГРАММЫ

Для создания сцены с генератором квестов - модуль «сказитель историй», требуется создать в сцене UI объект «Полотно» параллельно с ним создается EventSystem внутри Canvas нужно создать панель «StoryTellerPanel» и панель «LoadingPanel», необходимо создать «LLM\_UI\_Binder» с компонентом LLMUI Binder скриптом, для заполнения LLMUI Binder нужно реализовать на сцене поле ввода для описания истории, поле ввода для длины истории, выпадающий список с стилем генерируемого квеста, выпадающий список типа квеста, выпадающий список с сложностью, кнопка «генерировать», кнопка «сохранить», игровой объект контроллера прототипа генератора квестов с скриптом «LLM Prototype Controller», «LLM» скрипт устанавливается на пустой объект в сцене в настройки модели добавляется хранящаяся локально в проекте модель Mistral-7B-Instruct, находящаяся в папке StreamingAssets/Models. Для работы с моделью нужен объект с компонентом «LLM Character» скрипт, нужно настроить промт для контекста работы с квестами по тематике национальной культуры России.

Объяснение работы сцены с квестовом генератором «Сказитель историй»: нажатие кнопки «Сгенерировать» запускает функцию «generateButton.onClick.AddListener(OnGenerateClicked)» в скрипте LLMUIBinder.cs метод «Start()», вызывается метод «async void OnGenerateClicked», происходит сбор пользователя из UI, формируется JSON для Mistral-7B-Instruct, в случае отдельной сцены с «сказителем историй» происходит асинхронная генерация, следующим этапом работы «LLMQuestGenerator.cs» посылающий дебаг в консоль «Генерация квеста с promt», происходит передача текста промта в LLMPrototypeController, полученный ответ от модели возвращает в LLMUIBinder, LLMPrototypeController используя метод «public async Task<string> ProcessQuestRequest(string promt)» – благодаря асинхронному вызову пользовательский интерфейс не замораживается можно двигать мышью и видеть изменения в объекте TMPro, возвращает готовый квест сгенерированный моделью, в конце происходит «outputText.text = result;» вывод результата в объект TMPro в графическом интерфейсе. Созданы три основных объекта – LLMPrototypeController.cs для управления языковой моделью Mistral-7B-Instruct – отправка промта модели, возврат текста в консоль и объект TMPro на сцене. Скрипт LLMQuestGenerator для формирования JSON использующегося в качестве промта в виде токенов содержащим классы, для языковой модели трансформера Mistral-7B-Instruct. LLM\_UI\_Binder управляющий логикой и связывающий кнопки с функциями. выпадающими списками полями ввода, с LLMQuestGenerator и LLMPrototypeController.   
 Для реализации сцены с генератором аватарок, спрайтов, иконок – модуль «Генератор икон», необходимо создать объекты с скриптами LLMPrototypeController, LLMCharacter, полотно Canvas с панелью генератора икон «UI panel», поле ввода под описание иконки, выпадающий список с стилем иконок, поле ввода под указание размера иконок, кнопки для генерации икон, и сохранения в Exports/Visuals, текстовое поле с статусом генерации иконки, на уровне панели генератора икон создать панель загрузки c статусом генерации. Необходимо запускать скрипт main.py для использования ComfyUI при генерации аватарок, спрайтов, иконок используются локально установленные в проекте диффузионные модели sd-turbo и sd3-medium, генерация происходит за счет отправки промта с собранными параметрами генерации из полей ввода и выпадающих списках в виде JSON преобразующийся в эмбеддинги, при завершении генерации при помощи LLMUIBinder сгенерированное изображение предается на RawImage находящийся на сцене, для реализации представления пользователю результата, на сцене используются два новых скрипта ComfyUIManager.cs для передачи промта сервису ComfyUI, LLMIconGenerator.cs вызывает метод «GenerateIcon» создающий промт для дифузионок и посылает его ComfyUIManager.  
 Для сборки сцены с генератором диалоговых деревьев, необходимо реализовать в сцене полотно – Canvas с дочерними компонентами: Текстовое поле для имени говорящего NPC, поле ввода для описания окружения NPC, выпадающий список с эмоциями не игрового персонажа, поле ввода для игрока задающего вопрос NPC, кнопка завершающая цепочку диалога, копка отправить для отправки вопроса к неигровому персонажу, кнопка сохранения сгенерированного диалога, кнопка сгенерировать диалог на основе введенных данных о контексте окружения, характере поведения NPC, поле ввода реакции не игрового персонажа, для выполнения асинхронного метода generateButton.onClick.AddListener(OnGenerateClick), для этого в сцене должен быть объект с скриптом LLMQuestGenerator.cs, нужны объекты с скриптами LLMCharacter.cs – с ссылкой на LLM объект с загруженной моделью, необходимо ввести уточняющий работу промт «Ты чат-бот работающий с диалоговыми деревьями в юнити для обучения студентов Московского института имени Витте национальной культуре России, опор на историчность и национальность и национальную культуру России», пустой объект с LLMPrototypeController.cs – для связывания LLM.cs LLMCharacter.cs отправки запроса Mistral-7B-Instruct, LLMUIBinder.cs связывающий кнопки с функциями асинхронной генерации для возможности видеть изменения в текстовых полях TMPro на сцене выдающих результат генерации, LLM.cs – нужный для интеграции модели в генерацию диалогового дерева, а также настройки потоков и количества используемых ядер процессора и графической памяти, LLM позволяет настроить количество контекста в модели, для увеличения производительности или точности выдачи подходящего по контексту слова на следующем шаге генерации. Для работы сцены с модулем «контроллера поведения не игровых персонажей» необходим пустой объект с скриптом NPC Dialogue Controller для управления объектами кнопок, полей ввода, текстовых полей, связки с LLMPrototypeController.cs для отправки промтов модели Mistral-7B-Instruct, а также для вызова асинхронной генерации имитации диалога между NPC.

7 ПОРЯДОК РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ.

7.1 Виды, состав, объем и методы испытаний системы

«СДГВМ» тестировался и проходил испытания на функционал «способность ONNX моделей обмениваться между собой информацией и создавать виртуальный контент», стабильность результатов благодаря режиму тестирования и запуску тестов с сбором аналитической отчетности, измерял качество и производительность при помощи Unity Profiler.

7.2 Общие требования к приемке работ по стадиям

Приемка сделанной работы по реализации системы динамической генерации виртуальных миров, реализуетсясогласно этапам предоставленным в пятой главе технического задания. Функционал «СДГВМ» сверяется на соответствие с утвержденным Андреем Алексеевичем Блощуком техническим заданием, включающим испытания качества, производительности, стабильности.   
 Приемка работ начинается с сдачи готового технического задания по «СДВГМ», написанной первой главы диплома, до декабря утверждается тема и формируется оценка актуальности «СДГВМ» с дорожной картой реализации, перед утверждением техническое задание на разработку «СДГВМ» получило замечания от основного научного руководителя и преподавателей МУИВ, включая Максима Владимировича, Николая Александровича, Андрея Сергеевича, Федора Владимировича, Ольги Муратовной, на основе предоставленных советов и замечаний были внесены правки в настоящее техническое задание по разработке «СДГВМ».

Вторым этапом приёмки выполненной работы по реализации «СДГВМ», проверка руководителем наличия интерфейса для генерации контента, функций генерации квестов, диалогов, иконок, руководитель смотрит на наличие правильного перевода в локализации сделанной на основе пакета Unity «Localisation».

Третьим этапом приёмки будет демонстрация руководителю логов об проведении тестов с указанием качества контента, руководитель обозревает логи и просит объяснить возникающие в результате генерации ошибки, создаются рекомендации размещаемые в локальные файлы «СДГВМ» в папку «Обучающие материалы», Андрей Алексеевич просматривает папку с обучающими материалами по оперированию «СДГВМ» и даёт оценку качеству объяснений с замечаниями, использующимися для улучшения качества объяснения.

Четвертым этапом проверяется качество оптимизации моделей ONNX, для потребления ресурсов локальной машины и проверяются ситуации с длительной сессией генерации, проходит общая доработка и исправление ошибок.

Пятым этапом приемки выполненной работы является защита отчета о выполненной работе с презентацией перед проверяющей комиссией, проведение поддержки проекта в будущем после защиты «СДГВМ».

7.3 Статус приёмочной комиссии

Приемочная комиссия по проверке отчета по разработке системы динамической генерации виртуальных миров состоит из работников МУИВ: Сергея Александровича Зайцева, Андрея Алексеевича Блощука – научного руководителя ВКР «Динамическая генерация игровых миров при помощи нейросетей», Андрея Алексеевича Простомолотова, Сергея Сергеевича Постнова, Дмитрия Павловича Короткова.

**8 ТРЕБОВАНИЯ К СОСТАВУ И СОДЕРЖАНИЮ РАБОТ ПО** **ПОДГОТОВКЕ ОБЪЕКТА АВТОМАТИЗАЦИИ К ВВОДУ СИСТЕМЫ В ДЕЙСТВИЕ**

Разрабатывая «СДГВМ» были выявлены требования по подготовке и представлению к защите ВКР, руководитель Андрей Алексеевич Блощук предъявил следующие требования для реализации «СДГВМ» и защите ВКР:

Построение матрицы ответственности по ролям, демонстрирующую системный подход к реализации, распределению сфер ответственности;

Выполнение требований технического консультанта – Максима Владимировича Преображенского, установление контрольных точек в процессе реализации проекта, также постановка приоритетных задач, для получения MVP версии «СДГВМ»;   
 Тестирование и оптимизация итогового инструмента «СДГВМ» для ускорения скорости генерации, и улучшения качества генерируемого контента, при помощи использования метрик и юнит тестов.

Таблица 3. Матрица ответсвенности – роли участников разработки «СДГВМ»

|  |  |  |
| --- | --- | --- |
| Задача | Роль | ФИО участника |
| Проверка технической документации | Научный руководитель «СДГВМ» | Блощук Андрей Алексеевич |
| Объединение общей архитектуры проекта, разработка функционала | Технический консультант | Максим Владимирович Преображенский |
| Тестирование интерфейса на понятность | Тестировщик интерфейса | Мудров Никита Андреевич |
| Помощь в разработке скриптов функционала | Программист | Исаев Николай Александрович, Шебанов Вячеслав Викторович |
| Тестирование функционала и соотнесение с ГОСТ 7.0.97-2016 | Знающий стандарты платформы Unity специалист | Преображенский Максим Владимирович, Блощук Андрей Алексеевич |
| Обзор ошибок в архитектуре текстового содержания диплома и раздача замечаний по улучшений | Научный руководитель проекта «СДГВМ» | Блощук Андрей Алексеевич |
| Оптимизация технической составляющей | Технический консультант | Максим Владимирович Преображенский |
| Подготовка к защите ВКР | Научный руководитель проекта «СДГВМ» | Блощук Андрей Алексеевич |

9 ТРЕБОВАНИЯ К ДОКУМЕНТИРОВАНИЮ

Документы для разработки инструмента динамической генерации виртуальных миров, были согласованы научным руководителем и распределены в соответствии ГОСТ 34.201-89 и РД 50-34.698-90.

Таблица 4. Требования к документированию.

|  |  |  |
| --- | --- | --- |
| Стадия создания | Наименование документа | Дополнительные указания |
| ТЗ | Техническое задание системы динамической генерации игровых миров «СДГВМ» | ГОСТ 34.602-89 – требования к интерфейсу пользователей, утверждает метрики качества BLEU для Mistral 7B – Instruct «Контролер поведения не игровых персонажей» |
| ТП | Записка с описанием принципов преобразования n-грамм в эмбеддинги трансформерами модели Mistral 7B - Instruct | Описание принципов работы трансформеров в РД 50-34.698-90, а также РД 50-34.698-90 содержит сведения о архитектуре модели, применяющейся кросс энтропии использующейся для функции минимизации потерь, размере модели, описание взаимодействий при меж нейросетевом общении через LLM-Unity |
| Методики проведения испытаний функционала «СДГВМ» | Перечень необходимых проверок функционала в соответствии с ГОСТ 34.603-3, включает в себя метрики для проверки качества по ГОСТ 7.0.97-2016 |
| Организационная структура и распределение ролей | Были созданы роли на основе поставленных научным руководителем в ходе проектирования «СДГВМ» задач, информация включена в РД 50-34.698-90 |
| Объяснение функций для автоматизации реализуемых в «СДГВМ» | Генерация контента в виде икон, диалогов, историй и сюжета описана в РД 50-34.698-90 |
| РД | “Минимальные системные требования к локальной машине” | Для запуска и эксплуатации системы определены минимальные требования к ресурсам локальной машины, |
| “Учебные материалы” с инструкциями в формате PDF | ГОСТ 19.503-79 помог создать инструкции по запуску, настройке и эксплуатации |

10 ИСТОЧНИКИ РАЗРАБОТКИ

Идея создания системы динамической генерации игровых миров основана на глубоких исследованиях российского рынка геймдева, в ходе исследования выявлена востребованность доступных инструментов для генерации игрового контента, ниша генерации контента для игровых прототипов актуальна для создания систем автоматизации функционала сборки игровых виртуальных миров, целью разработки «СДГВМ» было создание доступного инструмента генерации игровых прототипов для инди-студий и соло разработчиков, существующие аналоги доступные для платформы Unity, платные или являются отдельным инструментом содержащим функционал генерации одного модуля системы «СДВГМ». Документация по разработке, обучающие материалы, требования к качеству сгенерированного контента разрабатывались согласно ГОСТ 34.603-92, ГОСТ 34.201-89, ГОСТ 34.602-89 и приказом министерства образования и науки РФ №816 Об утверждении порядка реализации образовательных программ высшего образования.

# Приложение 2. Исходный код “СДГВМ”

<https://github.com/Slavik993/Diplom_Shebanov_2026.git>

ComfyUIManager.cs - нужен для настройки подключения к локальному серверу генерации ComfyUI, в инспекторе назначается IP адрес, наименование JSON рабочего потока для генерации - без этого не будет генерации, ComfyUI нужно объянить как нужно использовать модель Awesome RPG Icon 2000 в формате шаблона JSON, также нужен для автозапуска сервера, при отсутствии установленных библиотек в виртуальной среде или недостаточной мощности компьютера, нужно запускать сервер вручную, в инспекторе назначается прогресс бар генерации иконки и текстовый объект TextMeshPro для отображения процентов состояния готовности картинки.

using UnityEngine;

using UnityEngine.UI;               // для Slider

using UnityEngine.Networking;

using TMPro;                        // для TMP\_Text

using System;                       // ← ОБЯЗАТЕЛЬНО! Action<>, Func<>, etc.

using System.Collections;

using System.IO;

using System.Text;

using System.Text.RegularExpressions;

using System.Diagnostics;            // для Process

using Debug = UnityEngine.Debug;    // чтобы не было конфликта Debug   // ← чтобы не было конфликта Debug

public class ComfyUIManager : MonoBehaviour

{

    [Header("ComfyUI Settings")]

    public string workflowFile = "awesome\_rpg\_icon\_workflow.json";

    public string comfyURL = "http://127.0.0.1:8188";

    public float pollInterval = 1f;

    //public Slider progressBar;        // перетащи Slider из UI

    //public TMP\_Text progressText;     // опционально — текст "45%"

    public Slider iconProgressBar;        // Перетащи сюда Slider из UI

    public TMP\_Text iconProgressText;     // Перетащи сюда TextMeshPro для процентов (можно оставить пустым)

    private string currentPromptId = "";

    [Header("Server Auto-Start")]

    public bool autoStartServer = true;

    public bool useCPU = true;

    [Header("Timeout Settings")]

    [Tooltip("Таймаут запуска сервера в секундах (рекомендуется 180+ для CPU режима)")]

    public int serverStartTimeout = 300; // Увеличен до 3 минут

    [Header("Path Settings")]

    [Tooltip("Оставьте пустым для автоматического поиска")]

    public string customComfyUIPath = "";

    private string ComfyUIPath

    {

        get

        {

            if (!string.IsNullOrEmpty(customComfyUIPath) && Directory.Exists(customComfyUIPath))

            {

                return customComfyUIPath;

            }

            #if UNITY\_EDITOR

            string editorPath = Path.Combine(Application.dataPath, "ComfyUI");

            if (Directory.Exists(editorPath))

            {

                return editorPath;

            }

            string projectPath = Path.Combine(Application.dataPath, "..", "ComfyUI");

            if (Directory.Exists(projectPath))

            {

                return projectPath;

            }

            string portablePath = Path.Combine(Application.dataPath, "..", "..", "ComfyUI\_windows\_portable");

            if (Directory.Exists(portablePath))

            {

                return portablePath;

            }

            return editorPath;

            #else

            return Path.Combine(Application.dataPath, "..", "ComfyUI\_Portable");

            #endif

        }

    }

    private string availableModel = null;

    private Process comfyProcess = null;

    private static bool serverAlreadyRunning = false;

    void Start()

    {

        UnityEngine.Random.InitState((int)System.DateTime.Now.Ticks);

        StartCoroutine(InitializeComfyUI());

    }

    void OnApplicationQuit()

    {

        if (comfyProcess != null && !comfyProcess.HasExited)

        {

            UnityEngine.Debug.Log("🔵 Leaving ComfyUI server running...");

        }

    }

    IEnumerator InitializeComfyUI()

    {

        UnityWebRequest testReq = UnityWebRequest.Get($"{comfyURL}/system\_stats");

        yield return testReq.SendWebRequest();

        if (testReq.result == UnityWebRequest.Result.Success)

        {

            serverAlreadyRunning = true;

            UnityEngine.Debug.Log("✅ ComfyUI server already running!");

        }

        else

        {

            UnityEngine.Debug.Log("⚠️ ComfyUI server not detected");

            if (autoStartServer && !serverAlreadyRunning)

            {

                UnityEngine.Debug.Log("🚀 Starting ComfyUI server...");

                yield return StartComfyUIServer();

                serverAlreadyRunning = true;

            }

            else

            {

                UnityEngine.Debug.LogError("❌ Please start ComfyUI manually or enable autoStartServer!");

                yield break;

            }

        }

        yield return LoadAvailableModels();

    }

    IEnumerator StartComfyUIServer()

    {

        string comfyPath = ComfyUIPath;

        string pythonExe = Path.Combine(comfyPath, "python\_embeded", "python.exe");

        string mainScript = Path.Combine(comfyPath, "ComfyUI", "main.py");

        UnityEngine.Debug.Log($"🔍 Looking for ComfyUI at: {comfyPath}");

        if (!Directory.Exists(comfyPath))

        {

            UnityEngine.Debug.LogError($"❌ ComfyUI folder not found: {comfyPath}");

            UnityEngine.Debug.LogError("💡 Make sure ComfyUI folder exists or set customComfyUIPath!");

            yield break;

        }

        if (!File.Exists(pythonExe))

        {

            UnityEngine.Debug.LogError($"❌ Python not found: {pythonExe}");

            UnityEngine.Debug.LogError("💡 Check python\_embeded folder in ComfyUI!");

            yield break;

        }

        if (!File.Exists(mainScript))

        {

            UnityEngine.Debug.LogError($"❌ main.py not found: {mainScript}");

            yield break;

        }

        bool processStarted = StartComfyProcess(pythonExe, mainScript, comfyPath);

        if (!processStarted)

        {

            UnityEngine.Debug.LogError("❌ Failed to start ComfyUI process");

            yield break;

        }

        UnityEngine.Debug.Log($"⏳ Waiting for ComfyUI to start (timeout: {serverStartTimeout}s)...");

        if (useCPU)

        {

            UnityEngine.Debug.Log("⚠️ CPU mode enabled - startup may take 2-3 minutes");

        }

        float elapsed = 0f;

        bool started = false;

        int checkInterval = 3; // Проверяем каждые 3 секунды вместо 2

        while (elapsed < serverStartTimeout)

        {

            yield return new WaitForSeconds(checkInterval);

            elapsed += checkInterval;

            UnityWebRequest checkReq = UnityWebRequest.Get($"{comfyURL}/system\_stats");

            yield return checkReq.SendWebRequest();

            if (checkReq.result == UnityWebRequest.Result.Success)

            {

                started = true;

                UnityEngine.Debug.Log($"✅ ComfyUI server started successfully! (took {elapsed:F1}s)");

                break;

            }

            // Показываем прогресс каждые 15 секунд

            if ((int)elapsed % 15 == 0 || elapsed >= serverStartTimeout - checkInterval)

            {

                float progress = (elapsed / serverStartTimeout) \* 100f;

                UnityEngine.Debug.Log($"⏳ Still starting... {elapsed:F0}s / {serverStartTimeout}s ({progress:F0}%)");

            }

        }

        if (!started)

        {

            UnityEngine.Debug.LogError($"❌ Server failed to start within {serverStartTimeout}s!");

            UnityEngine.Debug.LogError("💡 Solutions:");

            UnityEngine.Debug.LogError("   1. Increase 'Server Start Timeout' in Inspector (try 300s)");

            UnityEngine.Debug.LogError("   2. Start ComfyUI manually first, then run Unity");

            UnityEngine.Debug.LogError("   3. Check if ComfyUI console shows any errors");

            UnityEngine.Debug.LogError("   4. Disable 'Use CPU' if you have NVIDIA GPU");

        }

    }

    private bool StartComfyProcess(string pythonExe, string mainScript, string comfyPath)

    {

        try

        {

            string arguments = $"\"{mainScript}\" --listen 127.0.0.1 --port 8188";

            if (useCPU)

            {

                arguments += " --cpu";

                UnityEngine.Debug.Log("🖥️ Starting in CPU mode (slower but works without GPU)");

            }

            ProcessStartInfo startInfo = new ProcessStartInfo

            {

                FileName = pythonExe,

                Arguments = arguments,

                WorkingDirectory = Path.Combine(comfyPath, "ComfyUI"),

                UseShellExecute = false,

                CreateNoWindow = false,

                RedirectStandardOutput = false,

                RedirectStandardError = false

            };

            UnityEngine.Debug.Log($"▶️ Launching: {pythonExe}");

            UnityEngine.Debug.Log($"📝 Arguments: {arguments}");

            UnityEngine.Debug.Log($"📁 Working dir: {startInfo.WorkingDirectory}");

            comfyProcess = Process.Start(startInfo);

            if (comfyProcess != null)

            {

                // Принудительно убиваем старый процесс, если он остался

                try

                {

                    foreach (var process in Process.GetProcessesByName("python"))

                    {

                        if (process.MainModule.FileName.Contains("ComfyUI"))

                        {

                            process.Kill();

                            process.WaitForExit(3000);

                        }

                    }

                }

                catch { }

            }

            return comfyProcess != null;

        }

        catch (Exception e)

        {

            UnityEngine.Debug.LogError($"❌ Failed to start ComfyUI: {e.Message}");

            return false;

        }

    }

    IEnumerator LoadAvailableModels()

    {

        UnityEngine.Debug.Log("🔍 Checking available models...");

        UnityWebRequest req = UnityWebRequest.Get($"{comfyURL}/object\_info/CheckpointLoaderSimple");

        yield return req.SendWebRequest();

        if (req.result == UnityWebRequest.Result.Success)

        {

            string response = req.downloadHandler.text;

            availableModel = ExtractFirstModel(response);

            if (!string.IsNullOrEmpty(availableModel))

            {

                UnityEngine.Debug.Log($"✅ Found model: {availableModel}");

            }

            else

            {

                UnityEngine.Debug.LogError("❌ No models found! Add models to ComfyUI/models/checkpoints/");

            }

        }

        else

        {

            UnityEngine.Debug.LogError($"❌ Failed to connect to ComfyUI: {req.error}");

        }

    }

    public void Generate(string prompt)

    {

        if (string.IsNullOrEmpty(availableModel))

        {

            UnityEngine.Debug.LogError("❌ No model available!");

            return;

        }

        StartCoroutine(GenerateTexture(prompt, (tex) => {

            if (tex != null)

            {

                GetComponent<Renderer>().material.mainTexture = tex;

            }

        }));

    }

    public IEnumerator GenerateTexture(string prompt, Action<Texture2D> callback)

    {

        // Сброс прогресс-бара в начало

        ResetProgressBar();

        if (string.IsNullOrEmpty(availableModel))

        {

            UnityEngine.Debug.LogError("Модель не найдена в checkpoints!");

            callback?.Invoke(null);

            yield break;

        }

        string path = Path.Combine(Application.streamingAssetsPath, workflowFile);

        if (!File.Exists(path))

        {

            UnityEngine.Debug.LogError("Workflow не найден: " + path);

            callback?.Invoke(null);

            yield break;

        }

        // Подготовка JSON-шаблона

        string template = File.ReadAllText(path);

        int newSeed = UnityEngine.Random.Range(1, int.MaxValue);

        template = template.Replace("<PROMPT>", EscapeJson(prompt));

        template = Regex.Replace(template, @"""seed""\s\*:\s\*-?\d+", $"\"seed\": {newSeed}");

        template = template.Replace("УКАЖИТЕ\_ИМЯ\_ВАШЕЙ\_МОДЕЛИ.safetensors", availableModel);

        string payload = $"{{\"prompt\": {template}, \"client\_id\": \"unity\_{UnityEngine.Random.Range(100000,999999)}\"}}";

        // Ждём готовности сервера

        bool serverReady = false;

        float waitTime = 0f;

        while (!serverReady && waitTime < 30f)

        {

            var testReq = UnityWebRequest.Get($"{comfyURL}/prompt");

            yield return testReq.SendWebRequest();

            if (testReq.result == UnityWebRequest.Result.Success)

                serverReady = true;

            else

            {

                yield return new WaitForSeconds(1f);

                waitTime += 1f;

            }

        }

        if (!serverReady)

        {

            UnityEngine.Debug.LogError("ComfyUI не отвечает! Запустите сервер вручную.");

            callback?.Invoke(null);

            yield break;

        }

        // Отправляем промт

        byte[] body = Encoding.UTF8.GetBytes(payload);

        using (var req = new UnityWebRequest($"{comfyURL}/prompt", "POST"))

        {

            req.uploadHandler = new UploadHandlerRaw(body);

            req.downloadHandler = new DownloadHandlerBuffer();

            req.SetRequestHeader("Content-Type", "application/json");

            yield return req.SendWebRequest();

            if (req.result != UnityWebRequest.Result.Success)

            {

                UnityEngine.Debug.LogError($"Ошибка отправки промта: {req.error}");

                callback?.Invoke(null);

                yield break;

            }

            currentPromptId = ExtractPromptId(req.downloadHandler.text);

            if (string.IsNullOrEmpty(currentPromptId))

            {

                UnityEngine.Debug.LogError("Не удалось получить prompt\_id");

                callback?.Invoke(null);

                yield break;

            }

            UnityEngine.Debug.Log($"Генерация иконки началась (prompt\_id: {currentPromptId})");

        }

        // ОЖИДАНИЕ + ПРОГРЕСС-БАР (15 МИНУТ!)

        string imageFilename = null;

        float elapsed = 0f;

        float timeout = 900f; // 15 минут — спокойно ждёт даже CPU-режим

        while (elapsed < timeout && string.IsNullOrEmpty(imageFilename))

        {

            yield return new WaitForSeconds(1f);

            elapsed += 1f;

            // Прогресс-бар

            if (iconProgressBar != null)

                iconProgressBar.value = Mathf.Clamp01(elapsed / timeout);

            if (iconProgressText != null)

                iconProgressText.text = $"{(int)(elapsed / timeout \* 100)}%";

            var historyReq = UnityWebRequest.Get($"{comfyURL}/history/{currentPromptId}");

            yield return historyReq.SendWebRequest();

            if (historyReq.result == UnityWebRequest.Result.Success)

            {

                imageFilename = ExtractImageFilename(historyReq.downloadHandler.text);

            }

        }

        // Финал прогресс-бара

        if (iconProgressBar != null) iconProgressBar.value = 1f;

        if (iconProgressText != null) iconProgressText.text = "Готово!";

        if (string.IsNullOrEmpty(imageFilename))

        {

            UnityEngine.Debug.LogError($"Таймаут генерации иконки (15 минут)");

            callback?.Invoke(null);

            yield break;

        }

        // Скачиваем готовую иконку

        string imageUrl = $"{comfyURL}/view?filename={imageFilename}&type=output&subfolder=";

        var texReq = UnityWebRequestTexture.GetTexture(imageUrl);

        yield return texReq.SendWebRequest();

        if (texReq.result == UnityWebRequest.Result.Success)

        {

            Texture2D tex = DownloadHandlerTexture.GetContent(texReq);

            UnityEngine.Debug.Log($"Иконка загружена: {tex.width}x{tex.height}");

            callback?.Invoke(tex);

        }

        else

        {

            UnityEngine.Debug.LogError("Ошибка загрузки иконки: " + texReq.error);

            callback?.Invoke(null);

        }

    }

    // Сброс прогресс-бара

    private void ResetProgressBar()

    {

        if (iconProgressBar != null) iconProgressBar.value = 0f;

        if (iconProgressText != null) iconProgressText.text = "";

    }

    private IEnumerator CheckQueueStatus(string promptId)

    {

        UnityWebRequest queueReq = UnityWebRequest.Get($"{comfyURL}/queue");

        yield return queueReq.SendWebRequest();

        if (queueReq.result == UnityWebRequest.Result.Success)

        {

            string queueJson = queueReq.downloadHandler.text;

            int runningCount = Regex.Matches(queueJson, @"""queue\_running""").Count;

            int pendingCount = Regex.Matches(queueJson, @"""queue\_pending""").Count;

            if (runningCount > 0 || pendingCount > 0)

            {

                UnityEngine.Debug.Log($"📊 Queue status - Running: {runningCount}, Pending: {pendingCount}");

            }

        }

    }

    private string ExtractFirstModel(string json)

    {

        try

        {

            Match match = Regex.Match(json, @"""ckpt\_name"":\s\*\[\s\*\[(.\*?)\]");

            if (match.Success)

            {

                string modelsStr = match.Groups[1].Value;

                Match modelMatch = Regex.Match(modelsStr, @"""([^""]+)""");

                if (modelMatch.Success)

                {

                    return modelMatch.Groups[1].Value;

                }

            }

            return null;

        }

        catch (Exception e)

        {

            UnityEngine.Debug.LogError("Parse error: " + e.Message);

            return null;

        }

    }

    private string EscapeJson(string text)

    {

        return text.Replace("\\", "\\\\")

                   .Replace("\"", "\\\"")

                   .Replace("\n", "\\n")

                   .Replace("\r", "\\r")

                   .Replace("\t", "\\t");

    }

    private string ExtractPromptId(string json)

    {

        try

        {

            Match match = Regex.Match(json, @"""prompt\_id""\s\*:\s\*""([^""]+)""");

            return match.Success ? match.Groups[1].Value : null;

        }

        catch

        {

            return null;

        }

    }

    private string ExtractImageFilename(string json)

    {

        try

        {

            Match match = Regex.Match(json, @"""filename""\s\*:\s\*""([^""]+)""");

            return match.Success ? match.Groups[1].Value : null;

        }

        catch

        {

            return null;

        }

    }

}

ComfyUIUpdater.cs

Нужен для диагностирования и проверки работоспособности запущенного сервера ComfyUI, для выведения предупреждающих сообщений в консоле Unity, не нужно вешать скрипт на объект на сцене, работает пассивно из папки Scripts.

using UnityEngine;

using UnityEngine.Networking;

using System.Collections;

using System.Diagnostics;

using System.IO;

using Debug = UnityEngine.Debug;

public class ComfyUIUpdater : MonoBehaviour

{

    [Header("Настройки")]

    [Tooltip("Путь к папке ComfyUI (например: C:/ComfyUI\_windows\_portable)")]

    public string comfyUIPath = "C:/ComfyUI\_windows\_portable";

    [Tooltip("Проверять обновления при старте")]

    public bool checkOnStart = true;

    [Tooltip("Автоматически устанавливать обновления")]

    public bool autoInstall = false;

    private string pythonPath;

    private string comfyUIMainPath;

    void Start()

    {

        if (checkOnStart)

        {

            CheckAndUpdate();

        }

    }

    [ContextMenu("Проверить обновления ComfyUI")]

    public void CheckAndUpdate()

    {

        if (!Directory.Exists(comfyUIPath))

        {

            Debug.LogError($"ComfyUI не найден по пути: {comfyUIPath}");

            return;

        }

        // Определяем пути

        pythonPath = Path.Combine(comfyUIPath, "python\_embeded", "python.exe");

        comfyUIMainPath = Path.Combine(comfyUIPath, "ComfyUI");

        if (!File.Exists(pythonPath))

        {

            Debug.LogError($"Python не найден: {pythonPath}");

            return;

        }

        StartCoroutine(UpdateProcess());

    }

    IEnumerator UpdateProcess()

    {

        Debug.Log("🔍 Проверка обновлений ComfyUI...");

        // 1. Обновляем сам ComfyUI

        yield return StartCoroutine(RunCommand(

            "git",

            $"pull",

            comfyUIMainPath,

            "ComfyUI"

        ));

        // 2. Обновляем custom nodes

        string customNodesPath = Path.Combine(comfyUIMainPath, "custom\_nodes");

        if (Directory.Exists(customNodesPath))

        {

            foreach (string nodeDir in Directory.GetDirectories(customNodesPath))

            {

                if (Directory.Exists(Path.Combine(nodeDir, ".git")))

                {

                    string nodeName = Path.GetFileName(nodeDir);

                    yield return StartCoroutine(RunCommand(

                        "git",

                        "pull",

                        nodeDir,

                        $"Node: {nodeName}"

                    ));

                }

            }

        }

        // 3. Обновляем зависимости Python

        yield return StartCoroutine(RunCommand(

            pythonPath,

            "-m pip install --upgrade pip",

            comfyUIMainPath,

            "pip"

        ));

        yield return StartCoroutine(RunCommand(

            pythonPath,

            "-m pip install -r requirements.txt --upgrade",

            comfyUIMainPath,

            "dependencies"

        ));

        Debug.Log("✅ Обновление завершено!");

    }

    IEnumerator RunCommand(string program, string arguments, string workingDir, string taskName)

    {

        ProcessStartInfo startInfo = new ProcessStartInfo

        {

            FileName = program,

            Arguments = arguments,

            WorkingDirectory = workingDir,

            UseShellExecute = false,

            RedirectStandardOutput = true,

            RedirectStandardError = true,

            CreateNoWindow = true

        };

        Process process = new Process { StartInfo = startInfo };

        try

        {

            process.Start();

            string output = process.StandardOutput.ReadToEnd();

            string error = process.StandardError.ReadToEnd();

            process.WaitForExit();

            if (process.ExitCode == 0)

            {

                Debug.Log($"✅ {taskName}: успешно\n{output}");

            }

            else

            {

                Debug.LogWarning($"⚠️ {taskName}: {error}");

            }

        }

        catch (System.Exception e)

        {

            Debug.LogError($"❌ Ошибка {taskName}: {e.Message}");

        }

        yield return null;

    }

    [ContextMenu("Установить популярные ноды")]

    public void InstallPopularNodes()

    {

        StartCoroutine(InstallNodesProcess());

    }

    IEnumerator InstallNodesProcess()

    {

        string customNodesPath = Path.Combine(comfyUIMainPath, "custom\_nodes");

        string[] popularNodes = new string[]

        {

            "https://github.com/ltdrdata/ComfyUI-Manager.git",

            "https://github.com/Kosinkadink/ComfyUI-VideoHelperSuite.git",

            "https://github.com/pythongosssss/ComfyUI-Custom-Scripts.git"

        };

        foreach (string repo in popularNodes)

        {

            string repoName = Path.GetFileNameWithoutExtension(repo);

            yield return StartCoroutine(RunCommand(

                "git",

                $"clone {repo}",

                customNodesPath,

                $"Installing {repoName}"

            ));

        }

        Debug.Log("✅ Ноды установлены! Перезапустите ComfyUI.");

    }

    [ContextMenu("Проверить статус сервера")]

    public void CheckServerStatus()

    {

        StartCoroutine(PingServer());

    }

    IEnumerator PingServer()

    {

        using (UnityWebRequest www = UnityWebRequest.Get("http://127.0.0.1:8188/system\_stats"))

        {

            yield return www.SendWebRequest();

            if (www.result == UnityWebRequest.Result.Success)

            {

                Debug.Log("✅ ComfyUI сервер работает!\n" + www.downloadHandler.text);

            }

            else

            {

                Debug.LogError("❌ ComfyUI не отвечает. Запустите его!");

            }

        }

    }

}

CheckComfyModels.cs

Скрипт нужен для проверки подгруженных моделей ComfyUI, определяет сколько моделей подключено и говорит название, модели хранятся в папке ComfyUI под папке models, дочерней папке checkpoints, при наличии установленной модели скажет название в консоль Unity, скрипт работает пассивно без приерепления к объекту на сцене.

using UnityEngine;

using UnityEngine.Networking;

using System.Collections;

public class CheckComfyModels : MonoBehaviour

{

    void Start()

    {

        StartCoroutine(CheckAvailableModels());

    }

    IEnumerator CheckAvailableModels()

    {

        // Получаем список доступных чекпоинтов

        string url = "http://127.0.0.1:8188/object\_info/CheckpointLoaderSimple";

        UnityWebRequest req = UnityWebRequest.Get(url);

        yield return req.SendWebRequest();

        if (req.result == UnityWebRequest.Result.Success)

        {

            Debug.Log("📋 CheckpointLoaderSimple info:\n" + req.downloadHandler.text);

        }

        else

        {

            Debug.LogError("❌ Failed: " + req.error);

        }

        // Также проверим общую информацию

        url = "http://127.0.0.1:8188/object\_info";

        req = UnityWebRequest.Get(url);

        yield return req.SendWebRequest();

        if (req.result == UnityWebRequest.Result.Success)

        {

            // Ищем CheckpointLoaderSimple в общем списке

            string text = req.downloadHandler.text;

            int idx = text.IndexOf("CheckpointLoaderSimple");

            if (idx > 0)

            {

                string excerpt = text.Substring(idx, Mathf.Min(500, text.Length - idx));

                Debug.Log("📋 CheckpointLoaderSimple excerpt:\n" + excerpt);

            }

        }

    }

}

ComfyUIIntegration.cs

Нужен для отправки промта на сервер и выгрузки сгенерированной иконки через pipeline в Unity сцену объект Raw Image в формате png.

using UnityEngine;

using UnityEngine.Networking;

using UnityEngine.UI;

using System.Collections;

using TMPro;

public class ComfyUIIntegration : MonoBehaviour

{

    [Header("UI")]

    public TMP\_InputField promptInput;

    public RawImage outputImage;

    public Button generateButton;

    private readonly string serverURL = "http://127.0.0.1:8188";

    void Start()

    {

        generateButton.onClick.AddListener(GenerateImage);

    }

    public void GenerateImage()

    {

        if (string.IsNullOrEmpty(promptInput.text)) return;

        StartCoroutine(SendPrompt(promptInput.text));

    }

    IEnumerator SendPrompt(string prompt)

    {

        // Простой JSON для дефолтного workflow (адаптируй под свой)

        string json = "{\"prompt\": {\"1\": {\"inputs\": {\"text\": \"" + prompt + "\", \"clip\": [\"4\", \"IMAGE\"]}}, \"2\": {\"inputs\": {\"ckpt\_name\": \"sd\_xl\_base\_1.0.safetensors\"}}, \"3\": {\"inputs\": {\"seed\": 123, \"steps\": 20, \"cfg\": 8, \"sampler\_name\": \"euler\", \"scheduler\": \"normal\", \"denoise\": 1, \"model\": [\"2\", 0], \"positive\": [\"6\", 0], \"negative\": [\"7\", 0], \"latent\_image\": [\"5\", 0]}}, \"4\": {\"inputs\": {\"width\": 512, \"height\": 512, \"batch\_size\": 1}}, \"5\": {\"inputs\": {\"samples\": [\"3\", 0]}}, \"6\": {\"inputs\": {\"text\": \"positive prompt\", \"clip\": [\"4\", 0]}}, \"7\": {\"inputs\": {\"text\": \"negative prompt\", \"clip\": [\"4\", 0]}}, \"8\": {\"inputs\": {\"filename\_prefix\": \"UnityGen\", \"images\": [\"5\", 0]}}}";

        using (UnityWebRequest www = UnityWebRequest.PostWwwForm(serverURL + "/prompt", json))

        {

            www.SetRequestHeader("Content-Type", "application/json");

            yield return www.SendWebRequest();

            if (www.result != UnityWebRequest.Result.Success)

            {

                Debug.LogError("ComfyUI error: " + www.error);

                yield break;

            }

            // Ждём генерацию (проверь /history для статуса)

            yield return new WaitForSeconds(10f);  // Адаптируй под сложность

            yield return StartCoroutine(LoadGeneratedImage());

        }

    }

    IEnumerator LoadGeneratedImage()

    {

        // Загружаем последний результат (адаптируй URL по /history)

        using (UnityWebRequest www = UnityWebRequestTexture.GetTexture(serverURL + "/view?filename=UnityGen\_00001\_.png&subfolder=output&type=output"))

        {

            yield return www.SendWebRequest();

            if (www.result == UnityWebRequest.Result.Success)

            {

                Texture2D tex = DownloadHandlerTexture.GetContent(www);

                outputImage.texture = tex;

                Debug.Log("Изображение загружено!");

            }

            else

            {

                Debug.LogError("Ошибка загрузки: " + www.error);

            }

        }

    }

}

DiagnoseComfyUI.cs

Скрипт выводящий в консоль диагностику о правильно загруженных в чекпоинты диффузионных моделях.

using UnityEngine;

using UnityEngine.Networking;

using System.Collections;

public class DiagnoseComfyUI : MonoBehaviour

{

    void Start()

    {

        StartCoroutine(RunDiagnostics());

    }

    IEnumerator RunDiagnostics()

    {

        Debug.Log("=== 🔍 COMFYUI DIAGNOSTICS ===");

        // Тест 1: Проверка подключения

        Debug.Log("\n[TEST 1] Checking connection...");

        UnityWebRequest req = UnityWebRequest.Get("http://127.0.0.1:8188/system\_stats");

        yield return req.SendWebRequest();

        if (req.result != UnityWebRequest.Result.Success)

        {

            Debug.LogError("❌ ComfyUI not running! Start ComfyUI server first.");

            yield break;

        }

        Debug.Log("✅ ComfyUI is running");

        // Тест 2: Получаем информацию о CheckpointLoaderSimple

        Debug.Log("\n[TEST 2] Checking CheckpointLoaderSimple...");

        req = UnityWebRequest.Get("http://127.0.0.1:8188/object\_info/CheckpointLoaderSimple");

        yield return req.SendWebRequest();

        if (req.result == UnityWebRequest.Result.Success)

        {

            string response = req.downloadHandler.text;

            Debug.Log("📋 CheckpointLoaderSimple Response:\n" + response);

            // Проверяем, есть ли модели в списке

            if (response.Contains("\"ckpt\_name\""))

            {

                int start = response.IndexOf("\"ckpt\_name\"");

                int end = response.IndexOf("]", start) + 1;

                string ckptSection = response.Substring(start, end - start);

                Debug.Log("📦 ckpt\_name section: " + ckptSection);

                if (ckptSection.Contains("[[]]") || ckptSection.Contains("[[], "))

                {

                    Debug.LogError("❌ MODELS LIST IS EMPTY!");

                    Debug.LogError("⚠️ ComfyUI cannot find any checkpoint models.");

                }

            }

        }

        // Тест 3: Проверяем все доступные загрузчики

        Debug.Log("\n[TEST 3] Checking all loaders...");

        req = UnityWebRequest.Get("http://127.0.0.1:8188/object\_info");

        yield return req.SendWebRequest();

        if (req.result == UnityWebRequest.Result.Success)

        {

            string response = req.downloadHandler.text;

            // Ищем все типы загрузчиков моделей

            string[] loaders = {

                "CheckpointLoaderSimple",

                "CheckpointLoader",

                "UNETLoader",

                "DiffusersLoader"

            };

            foreach (string loader in loaders)

            {

                if (response.Contains(loader))

                {

                    Debug.Log($"✅ Found loader: {loader}");

                }

            }

        }

        // Тест 4: Проверяем embeddings (должны быть доступны)

        Debug.Log("\n[TEST 4] Checking if any resources are available...");

        req = UnityWebRequest.Get("http://127.0.0.1:8188/embeddings");

        yield return req.SendWebRequest();

        if (req.result == UnityWebRequest.Result.Success)

        {

            Debug.Log("📁 Embeddings response: " + req.downloadHandler.text);

        }

        Debug.Log("\n=== 🏁 DIAGNOSTICS COMPLETE ===");

        Debug.LogWarning("\n⚠️ NEXT STEPS:\n" +

            "1. Check ComfyUI console output for errors\n" +

            "2. Verify models are in: ComfyUI/models/checkpoints/\n" +

            "3. Restart ComfyUI server\n" +

            "4. Try loading a model manually in ComfyUI web interface");

    }

}

GameAI.cs

Важный скрипт для связывания элементов интерфейса на сцене с вызовом методов и функций, содержит в себе инициализирующие промты для работы разных моделей, автосохранение сгенерированного контента после генерации.

using UnityEngine;

using UnityEngine.UI;

using TMPro;

using LLMUnity;

using System.Collections;

using System.IO;

using System.Text.RegularExpressions;

using System;

public class GameAI : MonoBehaviour

{

    [Header("LLM NPC / Текст")]

    public LLMCharacter llmCharacter;

    [Header("==== INPUT LEFT PANEL ====")]

    public TMP\_InputField inputPrompt;

    public TMP\_InputField inputLength;

    public TMP\_Dropdown dropdownStyle;

    public TMP\_Dropdown dropdownType;

    public TMP\_Dropdown dropdownDifficulty;

    public TMP\_InputField inputIconStyle;

    public TMP\_InputField inputIconSize;

    public TMP\_Dropdown dropdownNPCEmotion;

    public TMP\_Dropdown dropdownNPCRelation;

    [Header("==== NPC CHAT ====")]

    public Button playerSendButton;

    public ScrollRect chatScrollRect;

    public TMP\_Text chatHistoryText;

    public TMP\_InputField playerInput;

    [Header("==== TEXT OUTPUT CENTER ====")]

    public TMP\_Text textStoryOutput;

    public ScrollRect storyScrollRect;

    [Header("==== IMAGE OUTPUT ====")]

    public RawImage iconPreview;

    [Header("==== BUTTONS ====")]

    public Button btnGenerate;

    public Button btnSaveAll;

    [Header("==== IMAGE GENERATION ====")]

    public ComfyUIManager comfy;

    [Header("==== AUTO SAVE SETTINGS ====")]

    public string saveFolderRoot = "QuestSessions";

    public bool autoSaveAfterGeneration = true;

    private string currentSessionFolder;

    private int generationCounter = 0;

    void Start()

    {

        CreateSessionFolder();

        btnGenerate.onClick.AddListener(GenerateAll);

        btnSaveAll.onClick.AddListener(SaveAll);

        if (playerSendButton != null)

            playerSendButton.onClick.AddListener(SendPlayerMessage);

        if (playerInput != null)

        {

            playerInput.onSubmit.AddListener(\_ => SendPlayerMessage());

            playerInput.onEndEdit.AddListener(text =>

            {

                if (Input.GetKeyDown(KeyCode.Return) || Input.GetKeyDown(KeyCode.KeypadEnter))

                    SendPlayerMessage();

            });

        }

        // Настройка центрирования текста квеста

        if (textStoryOutput != null)

        {

            textStoryOutput.alignment = TextAlignmentOptions.Center;

        }

        StartCoroutine(GenerateNPCResponse(""));

    }

    void CreateSessionFolder()

    {

        string sessionName = DateTime.Now.ToString("yyyy-MM-dd\_HH-mm-ss");

        currentSessionFolder = Path.Combine(Application.dataPath, saveFolderRoot, sessionName);

        Directory.CreateDirectory(currentSessionFolder);

        Debug.Log($"Session folder: {currentSessionFolder}");

    }

    public void GenerateAll()

    {

        generationCounter++;

        StartCoroutine(GenerateAllSequence());

    }

    public void SendPlayerMessage()

    {

        if (string.IsNullOrWhiteSpace(playerInput.text)) return;

        string playerMessage = playerInput.text.Trim();

        AddChatMessage("Игрок", playerMessage);

        playerInput.text = "";

        playerInput.ActivateInputField();

        StartCoroutine(GenerateNPCResponse(playerMessage));

    }

    IEnumerator GenerateNPCResponse(string playerMessage)

    {

        if (!llmCharacter)

        {

            AddChatMessage("NPC", "Ошибка: нет связи с ИИ");

            yield break;

        }

        AddChatMessage("NPC", "…");

        StartCoroutine(ScrollDelayed());

        // КЛЮЧЕВОЕ ИЗМЕНЕНИЕ: используем прямую инструкцию без ролевых меток

        string systemPrompt;

        if (string.IsNullOrEmpty(playerMessage))

        {

            systemPrompt = "Напиши короткое приветствие от дружелюбного персонажа русской сказки. 2-3 предложения. Говори от первого лица.";

        }

        else

        {

            string emotion = dropdownNPCEmotion.captionText.text;

            systemPrompt = $@"Игрок сказал тебе: ""{playerMessage}""

    Ты - персонаж русской сказки. Твоё настроение: {emotion}

    Ответь игроку естественно, от первого лица, 2-3 предложения на русском языке.

    Говори просто и по-человечески, без меток вроде ""NPC:"" или ""Ответ:"".

    Твой ответ:";

        }

        bool done = false;

        string fullResponse = "";

        int tokenCount = 0;

        const int maxTokens = 150; // Ограничение на количество токенов

        llmCharacter.Chat(systemPrompt, r =>

        {

            fullResponse = r;

            tokenCount++;

            // Завершаем когда есть 2-3 предложения или достигнут лимит токенов

            int sentenceCount = Regex.Matches(r, @"[.!?]").Count;

            if (sentenceCount >= 2 || tokenCount > maxTokens)

            {

                done = true;

            }

        });

        // Ждём завершения с таймаутом

        float timeout = 30f;

        float elapsed = 0f;

        while (!done && elapsed < timeout)

        {

            elapsed += Time.deltaTime;

            yield return null;

        }

        yield return new WaitForSeconds(0.3f);

        string reply = fullResponse.Trim();

        Debug.Log($"[DEBUG] Сырой ответ модели (длина {reply.Length}): '{reply}'");

        if (!string.IsNullOrWhiteSpace(reply))

        {

            // Очистка от markdown

            reply = Regex.Replace(reply, @"\\*\\*(.\*?)\\*\\*", "$1");

            reply = Regex.Replace(reply, @"\\*(.\*?)\\*", "$1");

            // ВАЖНО: убираем все возможные префиксы ролей (включая в начале строки)

            reply = Regex.Replace(reply, @"^(NPC|Ответ|Реплика|Персонаж|Твой ответ):\s\*", "", RegexOptions.IgnoreCase);

            reply = Regex.Replace(reply, @"^\([^)]+\)\s\*[-–—]?\s\*", ""); // убираем (Пираты против Петра Первого) -

            // Убираем повторяющиеся префиксы внутри текста

            reply = Regex.Replace(reply, @"\bNPC[:\s]\*", "", RegexOptions.IgnoreCase);

            // Убираем кавычки

            reply = reply.Trim('"', '«', '»', ' ', '\n', '\r');

            // Ограничиваем 2-3 предложениями

            var sentenceMatches = Regex.Matches(reply, @"[^.!?]+[.!?]+");

            if (sentenceMatches.Count > 0 && sentenceMatches.Count > 3)

            {

                string limited = "";

                for (int i = 0; i < Math.Min(3, sentenceMatches.Count); i++)

                {

                    limited += sentenceMatches[i].Value;

                }

                reply = limited.Trim();

            }

            // Убираем мусор типа "...", если это единственное содержимое

            if (reply == "..." || reply == "…") reply = "";

        }

        // Fallback если ответ неадекватный

        if (string.IsNullOrWhiteSpace(reply) || reply.Length < 5 || reply.ToLower().StartsWith("npc"))

        {

            string[] fallback = {

                "Ох, милок, расскажи поподробнее...",

                "Ну ты даёшь! А дальше-то что?",

                "Слушаю тебя, странник.",

                "Хм... интересно. Продолжай.",

                "Да ты что! Не может быть!",

                "Ох, батюшки... ну и дела.",

                "Ишь ты какой! Это надо же!",

                "Ай да молодец! Рассказывай дальше."

            };

            reply = fallback[UnityEngine.Random.Range(0, fallback.Length)];

            Debug.LogWarning($"[DEBUG] Неадекватный ответ: '{fullResponse}', используем fallback");

        }

        // Убираем "…" и вставляем настоящий ответ

        if (chatHistoryText != null)

        {

            string text = chatHistoryText.text;

            int index = text.LastIndexOf("<color=#FFAA00>NPC:</color> …");

            if (index >= 0)

                chatHistoryText.text = text.Substring(0, index);

        }

        AddChatMessage("NPC", reply);

        StartCoroutine(ScrollDelayed());

        Debug.Log($"[DEBUG] Итоговый ответ NPC: '{reply}'");

    }

    private void AddChatMessage(string sender, string message)

    {

        if (chatHistoryText == null) return;

        string color = sender == "Игрок" ? "#00FF00" : "#FFAA00";

        chatHistoryText.text += $"\n<color={color}>{sender}:</color> {message}";

        ScrollToBottom();

    }

    private void ScrollToBottom()

    {

        if (chatScrollRect != null)

        {

            Canvas.ForceUpdateCanvases();

            chatScrollRect.verticalNormalizedPosition = 0f;

            Canvas.ForceUpdateCanvases();

        }

    }

    private IEnumerator ScrollDelayed()

    {

        yield return null;

        yield return null;

        ScrollToBottom();

    }

    private string GetShortChatHistory()

    {

        if (string.IsNullOrEmpty(chatHistoryText.text))

            return "Диалог только начинается.";

        string fullHistory = chatHistoryText.text

            .Replace("<color=#00FF00>Игрок:</color>", "Игрок:")

            .Replace("<color=#FFAA00>NPC:</color>", "NPC:")

            .Replace("\n", " | ");

        if (fullHistory.Length > 3000)

        {

            fullHistory = "..." + fullHistory.Substring(fullHistory.Length - 3000);

        }

        return fullHistory;

    }

   IEnumerator GenerateAllSequence()

    {

        yield return StartCoroutine(GenerateStoryCoroutine());

        yield return new WaitForSeconds(0.5f);

        yield return StartCoroutine(GenerateIconCoroutine());

        if (autoSaveAfterGeneration)

            SaveCurrentGeneration();

        // ЭТА СТРОЧКА — ГАРАНТИРОВАНТИРОВАННАЯ ПРОКРУТКА ВНИЗ ПОСЛЕ ВСЕГО!

        yield return StartCoroutine(ScrollStoryToBottom());

    }

    private IEnumerator ScrollStoryToBottom()

    {

        yield return null; // ждём один кадр

        yield return null; // ещё один — чтобы текст точно отрисовался

        Canvas.ForceUpdateCanvases();

        if (storyScrollRect != null)

        {

            storyScrollRect.verticalNormalizedPosition = 0f; // 0 = самый низ

            Canvas.ForceUpdateCanvases();

        }

    }

    IEnumerator GenerateStoryCoroutine()

    {

        if (!llmCharacter) yield break;

        string prompt = $@"Создай квест на русском языке.

    Тема: {inputPrompt.text}

    Длина: {inputLength.text} слов

    Стиль: {dropdownStyle.captionText.text}

    Тип: {dropdownType.captionText.text}

    Сложность: {dropdownDifficulty.captionText.text}

    Напиши только текст квеста, без пояснений.";

        textStoryOutput.text = "Генерация текста квеста...";

        bool done = false;

        llmCharacter.Chat(prompt, r =>

        {

            textStoryOutput.text = r;

            done = true;

        });

        yield return new WaitUntil(() => done);

        // КЛЮЧЕВАЯ ЧАСТЬ — ПРОКРУТКА ВНИЗ ПОСЛЕ ГЕНЕРАЦИИ

        yield return null; // ждём один кадр

        Canvas.ForceUpdateCanvases();

        if (storyScrollRect != null)

        {

            storyScrollRect.verticalNormalizedPosition = 0f; // 0 = низ

            Canvas.ForceUpdateCanvases();

        }

    }

    IEnumerator GenerateIconCoroutine()

    {

        string prompt = $"Awesome RPG icon of {inputPrompt.text}, russian folk style, sharp, centered, transparent background";

        bool done = false;

        yield return comfy.GenerateTexture(prompt, tex =>

        {

            if (tex != null) iconPreview.texture = tex;

            done = true;

        });

        yield return new WaitUntil(() => done);

    }

    void SaveCurrentGeneration()

    {

        try

        {

            string folder = Path.Combine(currentSessionFolder, $"gen\_{generationCounter:D3}");

            Directory.CreateDirectory(folder);

            File.WriteAllText(Path.Combine(folder, "quest.txt"), textStoryOutput.text);

            File.WriteAllText(Path.Combine(folder, "chat.txt"), chatHistoryText.text);

            if (iconPreview.texture is Texture2D tex)

                File.WriteAllBytes(Path.Combine(folder, "icon.png"), tex.EncodeToPNG());

            Debug.Log($"Сохранено: {folder}");

        }

        catch (System.Exception e) { Debug.LogError(e.Message); }

    }

    public void SaveAll() => SaveCurrentGeneration();

    public void OpenSessionFolder() => Application.OpenURL("file://" + currentSessionFolder);

}

IconGenerator.cs

Содержит основные функции для генерации и скачивания иконки с локального сервера.

using UnityEngine;

using UnityEngine.UI;

using UnityEngine.Networking;  // ← ОБЯЗАТЕЛЬНО!

using System.Collections;

public class IconGenerator : MonoBehaviour

{

    public InputField promptField;

    public RawImage iconDisplay;

    public Button generateButton;

    void Start()

    {

        if (generateButton != null)

            generateButton.onClick.AddListener(GenerateIcon);

    }

    public void GenerateIcon()

    {

        if (string.IsNullOrEmpty(promptField.text))

        {

            Debug.LogError("Введите промпт!");

            return;

        }

        StartCoroutine(SendToComfyUI(promptField.text));

    }

    IEnumerator SendToComfyUI(string prompt)

    {

        string json = "{\"prompt\": \"" + prompt + "\", \"steps\": 20}";

        byte[] bodyRaw = System.Text.Encoding.UTF8.GetBytes(json);

        using (UnityWebRequest www = new UnityWebRequest("http://127.0.0.1:8188/prompt", "POST"))

        {

            www.uploadHandler = new UploadHandlerRaw(bodyRaw);

            www.downloadHandler = new DownloadHandlerBuffer();

            www.SetRequestHeader("Content-Type", "application/json");

            yield return www.SendWebRequest();

            if (www.result != UnityWebRequest.Result.Success)

            {

                Debug.LogError("ComfyUI ошибка: " + www.error);

            }

            else

            {

                Debug.Log("Запрос отправлен в ComfyUI");

                yield return new WaitForSeconds(8f);

                StartCoroutine(LoadImage());

            }

        }

    }

    IEnumerator LoadImage()

    {

        using (UnityWebRequest www = UnityWebRequestTexture.GetTexture("http://127.0.0.1:8188/history/last"))

        {

            yield return www.SendWebRequest();

            if (www.result == UnityWebRequest.Result.Success)

            {

                Texture2D tex = DownloadHandlerTexture.GetContent(www);

                iconDisplay.texture = tex;

                Debug.Log("Иконка получена!");

            }

            else

            {

                Debug.LogError("Ошибка загрузки: " + www.error);

            }

        }

    }

}

# **Приложение 3. Руководство администратора**

# **Введение**

## **Область применения**

Инструментарий система динамической генерации виртуальных миров, необходим для автоматизации генерации игрового контента образовательных квестов по национальной культуре Российской Федерации при помощи межнейросетевого общения языковых и диффузионных моделей, сгенерированный контент в виде текста квеста, диалога и иконки, дорабатывается специализированными знатоками исторического и национального контекста, для увеличения образовательной и научной ценности квестов, контент вставляется в проекты в виде визуальных новелл. СДГВМ включает три модуля: “контроллер поведения не игровых персонажей”, “генератор икон”, “сказитель историй”, используются Awesome RPG Icon 2000 основанная на модели SD 1.5 обученная двумя тысячами эпох на одном стиле генерации Warcraft и Dungeons and Dragons, Mistral-7B-Instruct большая квантованная языковая модель с семью миллиардами параметров, использующаяся для генерации текста квестов и текста выдающегося LLMCharacter моделью поведения не игрового персонажа.

## **1.2 Описание возможностей**

Система динамической генерации виртуальных миров использует LLM-Unity для языковой модели Mistral-7B-Instruct и портативную версию ComfyUI amd версии способную запустить локальный сервер генерации без видеокарты в компьютере, возможности могут быть легко расширены установлением новых моделей и созданием новых заводских построек в ComfyUI интерфесе сервера - новые производственные линии для генерации 3D моделей, анимаций, видео, ландшафтной генерации, на данный момент СДГВМ способен быть инструментарием, для создания образовательных квестов и визуальных новелл, для увеличения уровня толерантности иностранных студентов к национальной культуре Российской Федерации.

## **1.3 Уровень подготовки администратора**

Для пользования системой динамической генерации виртуальных миров необходимы базовые знания работы с платформой Unity, СДГВМ нужна для создателя квестового контента работает роль разработчика для генерации, конечный пользователь иностранный студент не должен уметь пользоваться платформой Unity, для прохождения квеста или визуальной новеллы, разработчик генерирующий контент работает не посредственно с модулями системы динамической генерации виртуальных миров, также важно уметь работать с виртуальной средой Python, необходимо уметь запускать скрипты через терминал с специальными флагами, для корректной работы.

## **1.4 Перечень эксплуатационной документации**

Разработчик являющийся пользователем обязан прочитать руководство администратора, для корректного использования системы динамической генерации миров а для понимания в полной мере автоматизированного производственного процесса по созданию квестов, где участвуют две пользовательские роли - знаток исторического и культурного контекста и иностранный студент проходящий в конечном итоге квесты с сбором статистики для подтверждения динамических улучшений толерантности к национальной культуре Российской Федерации. Также разработчику работающему с СДГВМ следует почитать руководства к пакету LLM-Unity, руководство к ComfyUI, для возможности корректировки скриптов методом патчирования официальных библиотек, для выполнения специфических задач.

# **Назначение и условия применения**

СДГВМ разрабатывался для обеспечения разработчиков инструментарием, для автоматизированного создания квестов для повышения национально-культурной адаптации иностранных студентов. При помощи модуля “сказитель историй” с интегрированной моделью Mistral-7B-Instruct происходит автоматизация создания сюжета квеста, модуль “генератор икон” содержит интегрированную в производственный поток модель Awesome RPG Icon 2000, для генерации иконок для вставки в квесты и визуальные новеллы, модуль “контроллер поведения неигровых персонажей” использует скрипты LLM-Unity включающие внутрение модели на Tensorflow и Barracuda, для обучения ONNX моделей поведения не игровых персонажей. Инструментарий содержит научную значимость благодаря применению нейронных сетей в образовании и сбор статичных данных по психологическому состоянию студентов, для дообучения моделей. СДГВМ помогает развивать направление Game Dev, для поддержки стартапа, можно реализовывать практики для внедрения новых модулей с нейросетями позволяющих генерировать новые типы контента: 3D модели, видео, анимации, возможно дальнейшее расширение и коммерциализация стартапа за счет продажи пакета в Unity Assets Store. СДГВМ не требует подключения к интернету и будет работать при полном отключении интернета в стране.

Таблица 2.1. Требования к аппаратному обеспечению.

|  |  |  |
| --- | --- | --- |
| Параметр | Минимальные требования | Рекомендуемые требования |
| Операционная система | Linux Debian | Linux Debian или Windows 11 |
| Процессор | Intel Core i3 выше восьмого поколения | Intel Core i7 выше девятого поколения |
| ОЗУ | 8 Гигабайт | Выше 16 Гигабайт |
| Пространство на диске | 15,7 ГБ (16 884 219 904 байт) | 25 ГБ |
| Видеокарта | Не нужна | Любая видеокарта с 4 Гб памяти и больше “очень сильно ускоряет процесс генерации текста и иконок” |
| Разрешение экрана | 1366 на 768 | 1920 на 1080 лучший выбор |

Для работы программы требуются зависимости нужной версии:

Таблица 2.2.

|  |  |  |
| --- | --- | --- |
| Компонент | Версия | Примечание |
| Unity | 6000.0.30f1 | Основная платформа с интегрированными моделями |
| Python | 3.11.9 | Требуеьт запуск локального сервера для генерации икон |
| .NET Framework | 4.8 | Для отправки запросов на сервер |
| Visual C++ Redistributable | 2019-2022 | Нужен для работы с скриптами Unity |

Для работы СДГВМ айтишникам нужно добавить в белый список запуск сетевого скрипта main.py для запуска локального сервера генерации иконок, нужно убедиться что не один exe файл не блокируется политикой безопасности.

Для старта генерации в интерфейсе необходимо ввести описание истории лучше не писать слишком длинное описание, может привести к потере смысла, по умолчанию нейронка берет 8192 контекстуальных символа, лучше делать описание одним или двумя предложениями, нельзя кидать бессмысленные словосочетания или набор слов, описание должно содержать смысловую нагрузку, длина истории обозначается в количестве слов, при недостатке смыслового содержания промта получается меньше слов чем указано в поле ввода количество слов, нужно выбрать из выпадающих списков желаемый контекст, для иконок можно ничего не заполнять, стиль RPG и размер 512 на 512 настроен по умолчанию. После заполнения нужно нажать кнопку “генерировать”, после будет происходить процесс генерации в консоли будет отображаться время, а слайдер на экране будет показывать процент готовности генерации иконы, при завершении процесса генерации происходит сохранение контента.

# **Подготовка к работе**

Для работы СДГВМ нужно загрузить портативный ComfyUI версии amd, платформу Unity, для запуска, установить пакеты Newtonsoft и LLM-Unity, обязательно нужно установить локально в проект в ComfyUI папке, папка models с папкой checkpoints положить туда Awesome RPG Icon 2000, а в streaming assets нужно положить JSON с workflow, чтобы модель могла генерировать используя запущенный локальный сервер workflow имеет название: “awesome\_rpg\_icon\_workflow”. Mistral-7B-Instruct должна лежать в Streaming Assets также, но её можно хранить не в проекте и подгрузить в скрипт LLM самостоятельно через инспектор один раз, при запуске сцены с настроенной конфигурацией системы динамической генерации виртуальных миров будут выведены предупреждающие об отсутствии установленных элементов предупреждающие сообщения, например о необходимости запустить сервер генерации вручную через терминал Visual Studio Code, или отсутствии установленной модели для ComfyUI или проблема с запуском LLM-сервиса. Сначала рекомендуется разобраться с языковыми моделями и LLM, для получения текста истории или диалога с NPC, далее нужно разбираться с ComfyUI - следует заранее перед запуском запускать локальный сервер генерации через терминал с флагом cpu командой python main.py --cpu, нужно открывать Visual Studio Code в папке с main.py, иначе команда не найдёт скрипта, для исполнения. Далее после фикса проблем найденных при помощи дебаговых сообщений скриптов в консоли, можно осуществлять полноценную генерацию иконки, текста квеста и диалога. После автоматического сохранения сгенерированного контента, нужно зайти в папку QuestSessions и выбрать папку с текущим датой и временем это будет папка с актуальным новым сгенерированным после сессии контентом, и отправить контент знатоку исторического и культурного контекста знаток вносит правки и повышает уровень образовательного материала, потом исправленный контент отправляется разработчику с готовым шаблоном визуальной новеллы, для создания конечного квеста, для пользования студентами. Главные ошибки вынесены в дебаговые сообщения в консоли, главный ориентир при настройке и установке недостающих компонентов, для полноценной работы системы.

# **Подготовка к работе**

Система динамической генерации игровых миров выполняет важные задачи для автоматизации создания образовательного контента, для квестов повышающих адаптацию к национальной культуре Российской Федерации, в функционал входит генерация текста квеста, текста диалога, генерация иконки, сохранение сгенерированного контента для последующей работы, для выполнения функций, необходимо перед работой СДГВМ запустить локальный сервер генерации через Python, проветсти настройку в инспекторе объекта LLM находящегося в сцене, нужно загрузить локальную модель Mistral-7B-Instruct, необходимо проанализировать дебаговые сообщения при запуске, для понимания чего именно не хватает для корректной работы системы. Процесс генерации включает следующие условия: проверка работоспособности включающая просмотр дебаговых сообщений и устранение проблем, ввод начальных данных для формирования промта: нужно заполнить поля ввода описанием истории и выбрать в выпадающих списках подходящий под контекст задуманной генерации параметров - “поведения не игровых персонажей, отношение к игроку, тип квеста, стиль истории, сложность и т.д.”, дальше идёт нажатие кнопки генерации и ожидание пока создастся контент. Сохранение происходит само без вмешательства пользователя, но можно нажать кнопку сохранить, для сохранения при вылете или отключении электропитании, СДГВМ создаст бэкап и контент сохраниться в любом случае. В итоге работы системы получается контент отправляемый знатокам историческо культурного контекста для повышения образовательной ценности сгенерированных материалов, после проверки и преобразования материалы отправляют разработчикам с шаблонном визуальной новеллы и создаётся полноценный квест для иностранных студентов повышающий уровень толерантности к Росссийской национальной культуре и особенностям социума, для повышения успеваемости студентов и реализации в социуме.   
 Правильная последовательность действий: запуск локального сервера ComfyUI через скрипт main.py используя терминал и команду “python main.py --cpu”, вторым шагом нужно заполнить поле ввода описания истории и выбрать в выпадающих списках желаемые параметры генерации, третьим шагом нужно нажать кнопку “генерировать”, четвёртым шагом происходит автосохранение, пятым шагом сохраненный контент отправляется знатокам истории и культуры, для редактирования и внесения ценности - можно сделать практику для студентов педагогического факультета по редактированию материалов сгенерированных СДГВМ, для придания большей ценности.

По требуемым ресурсам компьютера необходимо минимум 8 Гигабайт ОЗУ, процессор Intel Core I3 восьмого поколения, 15,7 ГБ (16 883 593 216 байт) на жёстком диске минимум.

Для поиска необходимых файлов в проекте нужно использовать следующие пути: JSON workflow находится на пути: SDGVM\Assets\StreamingAssets; модель Awesome RPG Icon 2000 находится на пути ComfyUI\models\checkpoints. Пути указанные ранее обязательны для работоспособности проекта нельзя менять.

# **Аварийные ситуации**

Аварийные ситуации возможны при длительном использовании системы динамической генерации виртуальных миров: основные проблемы с нехваткой памяти при длительной сессии генерации решены, нужно быть готовым к появлению новых проблем, с учетом эволюции и добавления нового функционала СДГВМ, появляются новые проблемы, включая проблемы при длительной сессии эксплуатации СДГВМ, за счёт технических особенностей компьютеров с 8 Гб оперативной памяти, может происходить Unity Crash Log, необходимо перезапустить проект и продолжить генерировать контент, при отсутствии модели Mistral-7B-Instruct, нужно убедиться что загружена локально в компонент LLM, если в инспекторе модель не выбрана, нужно поставить галочку возле Mistral-7B-Instruct v 0.2, при эксплуатации возможны проблемы при отсутствии необходимого количества пространства на жёстком диске, возможны проблемы с работой инструментария СДГВМ, или вылет в виде Unity Crash Log.

При обнаружении несанкционированного вмешательства в данные, стоит проверить целостность сохранённого контента на предмет изменения или удаления части сгенерированных материалов, для предотвращения случаев удаления и утраты сгенерированного контента всегда перед уходом с рабочего места, нужно обязательно сделать копию контента на внешний накопитель или облачное хранилище. При попытках испортить конфигурацию СДГВМ злоумышленниками использовать сохраненную копию системы, для быстрого востановления и продолжения генерации контента.

# **Рекомендации по освоению**

Для освоения инструментария системы динамической генерации игровых миров, для генерации контента, нужно прочитать о основном функционале в руководстве администора, для понимания основных проблем возникающих при генерации необходимо читать консольные дебаговые сообщения, о недостающих компонентов в модулях для работы, о ошибках связанных с ComfyUI - включать сервер вручную через терминал нужно всегда, несмотря на автозапуск - может не сработать на слабых компьютерах, нужно убежиться о наличии установленных локально и подключенных в проект моделях нейронных сетей: Mistral-7B-Instruct и Awesome RPG Icon 2000, необходимо достаточное количества места на жестком диске, для записи необходимых для генерации весов модели и сгенерированного в итоге контента.

**Приложение 4. Руководство пользователя**

1. **Введение**

Конечными пользователями являются студенты, для прохождения созданных квестов при помощи инструментария СДГВМ, область применения системы динамической генерации виртуальных миров - создание игрового контента для исторических квестов или визуальных новелл по культурно-национальной тематик, текущая версия СДГВМ может генерировать текст сюжета для квестов, диалог, рисовать иконку, конечный пользователь студент не должен уметь пользоваться инструментом генерации контента, платформой Unity, студентам не требуется особых навыков для выбора ответа да или нет в визуальной новелле или других особых вариантов ответа, пользователю нужно почитать руководство по прохождению культурно-национального квеста.

1. **Назначение и условия применения**

Данное средство автоматизации создания культурно -национальных квестов и новелл, применяется в роли инструментария генерации контента. Условия - в первую очередь необходимость адаптации иностранных студентов к национальной культуре Российской Федерации, или необходимость внедрения в программу нового типа образовательных материалов в виде квестов и визуальных новелл содержащих историческую или культурную ценность, для повышения уровня знаний о культуре РФ и сближению молодёжи с историческими корнями. Необходимые требования вычислительной техники к готовому квесту или новелле намного ниже, не требуется много памяти на диске - около гигабайта требования к готовому квесту, не требуется мощный процессор или увеличенное количество оперативной памяти. Для прохождения квестов студентами нужно ознакомиться с руководством по прохождению - объясняются банально базовые вещи, как выйти, как зайти и как выбрать да или нет.

1. **Подготовка к работе**

Конечным пользователям студентам, не нужно вести никакие подготовки, для них разработан уже готовый квест в формате Build Unity или exe, студентам необходимо нажать два раз на ярлык запуска. При входе нужно заполнить данные о номере группы, ФИО для идентификации собранных статических данных о студенте, для проведения психологического исследования изменения уровня адаптации к национальной культуре Российской Федерации.

1. **Описание операций**

Для успешного анализа результатов, существует функционал, для отслеживания быстрых переходов между сценами для поиска бездумных ответов, механизмы ловли моментов не честного прохождения квестов студентами существуют, для подверждения достоверности и точности полученных результатов психологического исследовательского анализа проведенного психологами. Квест или визуальная новелла несет задачу сбора данных для дальнейшего исследования влияния СДГВМ, на иностранных студентов, на основе результатов полученных с исследований система динамической генерации виртуальных миров получает улучшения, для наибольшего воздействия затрагивающего толерантность к культуре.

Перечисление операций обработки: шаг первый запуск студентами квеста и прохождения формы регистрации для идентификации, шаг второй автоматический сбор статистических данных с тестирования студентов при помощи встроенных методов, сбор данных происходит с учётом механизма анализирующего честность прохождения квеста, шаг третий автоматическое сохранение результатов, для дальнейшего анализа психологами экспертами, третий шаг не нужно исполнять пользователю происходит автоматически, четвёртый шаг анализ собранных данных без участия пользователя, пятый шаг получение результатов анализа для дальнейшего улучшения СДГВМ инструментария генератора контента, в пятом шаге пользователь не участвует.

1. **Аварийные ситуации**

Возможен Unity Crash Log, с низкой долей вероятности, конечному пользователю студенту требуется запустить квест заново, также возможен сбой системы автосохранения или недостаточное количество места на диске для сохранения статических данных, нужно освободить место на диске и пройти квест заново, для сохранения данных нужных для исследования. Возможны технические проблемы с пропажей кнопок интерфейса или не возможности кликнуть на кнопку - нужно обратится за помощью к разработчику создающего квесты, не к разработчику работающему с СДГВМ, для генерации контента. При вмешательстве в данные или изменении изначально выданного экспертами культуры материалами, для создания квестов, нужно обратится к знатокам культуры о проблеме, для ещё одного цикла очистки данных, используемых в обучении и адаптации студентов.

1. **Рекомендации по освоению**

Для конечного пользователя - студента рекомендации просты, запустить exe файл или Unity Build, не торопясь вдумчиво выбрать нужные варианты ответов, дождаться автосохранения и закрыть квест. При возникновении технических проблем с сохранением или не функционирующим интерфейсом, следует обратится за помощью к разработчику, преподавателю, или техническую поддержку - заявить о проблеме расписать подробное описание и попросить выдать другой вариант квеста, для прохождения и дальнейшей оценки.