Федеральное государственное автономное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий  
Кафедра «Информатика и информационные технологии»

Направление подготовки/ специальность: Автоматизированные системы обработки информации и управления

ОТЧЕТ

по проектной практике

Студент: Румянцев Влад. Группа: 241-331

Место прохождения практики: Московский Политех, кафедра «Информатика и информационные технологии»

Отчет принят с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель практики: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2025

**ОГЛАВЛЕНИЕ**

**Оглавление**

[**Общая информация о проекте** 1](#_Toc199266989)

[**Общая характеристика деятельности организации *(заказчика проекта)*** 3](#_Toc199266990)

[**Описание задания по проектной деятельности** 4](#_Toc199266991)

[**Описание достигнутых результатов по проектной деятельности** 5](#_Toc199266992)

[**Базовая часть задания** 6](#_Toc199266993)

[**Вариативная часть задания** 8](#_Toc199266994)

[**Вклад в открытый проект на GitHub или GitVerse** 11](#_Toc199266995)

[**Заключение** 12](#_Toc199266996)

[**Литература и интернет-ресурсы** 13](#_Toc199266997)

[**Информация для сведения** 14](#_Toc199266998)

# **Общая информация о проекте**

Название проекта: Разработка компьютерной игры с процедурной генерацией и элементами выживания

Цели проекта: Создание компьютерной игры с процедурно генерируемыми биомами, динамической экосистемой и механиками выживания, которая будет обеспечивать игроку уникальный опыт и высокую реиграбельность.ей и элементами выживания

Задачи проекта:

* Разработать основные механики процедурной генерации миров и их контента (фауны, флоры, ландшафта).
* Определить принципы адаптации и эволюции игрока и существ.
* Реализовать алгоритмы генерации локаций, базовую логику выживания (поиск ресурсов, укрытия, защита от хищников).
* Добавить первичные элементы прогрессии (эволюция, прокачка навыков, примитивные технологии).
* Ввести положительные и отрицательные модификаторы, влияющие на геймплей (например, повышенная сила, но замедленная скорость).
* Сбалансировать риски и выгоды различных эволюционных путей.
* Реализовать динамические связи между видами (хищники, добыча, симбиоз).
* Учесть факторы вымирания, размножения и климатических изменений.  
  Добавить особые зоны (аномалии, нестандартная гравитация, токсичные участки).
* Реализовать систему случайных событий (катаклизмы, вторжения, перемены погоды).
* Провести альфа- и бета-тестирование с реальными игроками.
* Настроить сложность и вознаграждение за исследования, выживание и крафт.
* Подготовить пользовательский интерфейс, обучающие материалы, реализовать стабильный мультиплеер (при необходимости).
* Опубликовать игру на выбранных платформах (ПК, мобильные устройства и т. д.).

# **Общая характеристика деятельности организации *(заказчика проекта)***

Наименование заказчика: Московский Политехнический университет

Описание деятельности: Московский Политехнический университет является образовательной структурой, дающей возможность студентам разрабатывать инновационные проектов разный областях информационных технологий. Университет предлагает использование новейших передовых технологий для разработки проектов студентов.

# **Описание задания по проектной деятельности**

Необходимо разработать игровую концепцию и создать функциональный прототип компьютерной игры в жанре survival с элементами процедурной генерации. В рамках проекта требуется провести анализ целевой аудитории, изучив предпочтения игроков в жанрах survival и sandbox, а также проанализировать конкурентные решения на игровом рынке. Следующим этапом является разработка игровой бизнес-модели, которая должна включать выбор оптимальных платформ распространения (Steam, Epic Games Store, консоли) и проработку моделей монетизации (premium-модель, ранний доступ, DLC).

Параллельно необходимо спроектировать и протестировать минимально жизнеспособный продукт (MVP), обеспечив базовый функционал: процедурную генерацию мира, основные механики выживания и систему эволюции персонажа. Особое внимание следует уделить геймдизайну и юзабилити, чтобы обеспечить увлекательный и комфортный игровой процесс.

Финансовая часть проекта включает прогнозирование рентабельности и сроков окупаемости. На основе этих данных требуется подготовить инвест-презентацию с четкой структурой, включающей описание игрового продукта, анализ игрового рынка, конкурентные преимущества, финансовые показатели и маркетинговую стратегию. Дополнительно необходимо изучить возможности привлечения финансирования через гранты для игровых проектов, конкурсы инди-разработчиков и программы поддержки креативных индустрий.

Важной составляющей задания является развитие профессиональных компетенций в области игровой разработки, в частности, применение современных технологий и инструментов искусственного интеллекта для решения различных задач проекта - от генерации процедурного контента до балансировки игровых механик. Критериями успешного выполнения проекта станут: работоспособность игрового прототипа, готовность материалов для поиска инвестиций, а также наличие положительных отзывов от тестовой аудитории и выраженного интереса со стороны потенциальных издателей. Проект выполняется командой разработчиков с активным использованием современных инструментов разработки и AI-ассистентов для оптимизации рабочих процессов.

# **Описание достигнутых результатов по проектной деятельности**

Ключевыми факторами успеха оказались:

Эффективное распределение ролей в команде разработчиков и постоянное взаимодействие с научным руководителем

Использование современных игровых технологий и алгоритмов процедурной генерации

Гибкий итерационный подход к разработке, позволяющий оперативно вносить изменения на основе тестирования

В ходе работы над проектом:

1. Сформирована уникальная концепция survival-игры с динамической экосистемой и механиками эволюции
2. Разработаны и реализованы ключевые системы:
   * Процедурная генерация миров (биомы, ландшафт, ресурсы)
   * Механики выживания (потребности, укрытие, защита)
   * Система эволюции с балансом модификаций
3. Созданы:
   * Функциональный прототип игры
   * Официальный сайт проекта
   * Базовый набор 3D-моделей существ и объектов
4. Проведено:
   * Тестирование с фокус-группой (50+ участников)
   * Анализ игрового рынка и конкурентов
   * Предварительные расчеты бизнес-модели

Дальнейшие планы разработки:

1. Расширение функционала MVP:
   * Добавление системы сезонных изменений
   * Реализация кооперативного режима
   * Введение новых биомов и существ
2. Тестирование ключевых гипотез:
   * Достаточна ли вариативность процедурной генерации
   * Сбалансирована ли система эволюции
   * Интересен ли игрокам концепт динамической экосистемы
3. Подготовка к раннему доступу:
   * Доработка UI/UX
   * Создание обучающих материалов
   * Формирование маркетинговой стратегии

# **Базовая часть задания**

1. **Настройка Git и репозитория:**

* Создайте личный или групповой репозиторийна GitHub или GitVerse на основе предоставленного шаблона.
* Освойте базовые команды Git: клонирование, коммит, пуш и создание веток.
* Регулярно фиксируйте изменения с осмысленными сообщениями к коммитам.
* **Ожидаемое время:** 5 часов.

1. **Написание документов в Markdown:**

* Все материалы проекта (описание, журнал прогресса и др.) должны быть оформлены в формате Markdown.
* Изучите синтаксис Markdown и подготовьте необходимые документы.
* **Ожидаемое время:** 5 часов.

1. **Создание статического веб-сайта:**

* Вы можете использовать **только HTML и CSS** для создания сайта, если освоение более сложных инструментов представляется трудным. Это делает задание доступным для студентов с базовым уровнем подготовки.
* **Желательно** применять генераторы статических сайтов, такие как Hugo (рекомендуется), для упрощения процесса и получения дополнительных навыков. В случае выбора Hugo можно воспользоваться инструкциями из [Hugo Quick Start Guide](https://gohugo.io/getting-started/quick-start/).
* Создайте новый сайт об основном проекте по дисциплине «Проектная деятельность», выберите тему и добавьте контент. Оформление и наполнение сайта должны быть уникальными (не совпадать с работами других студентов) более, чем на 50%.
* Сайт должен включать:
* **Домашнюю страницу** с аннотацией проекта.
* **Страницу «О проекте»** с описанием проекта.
* **Страницу или раздел «Участники»** с описанием личного вклада каждого участника группы в проект по «Проектной деятельности».
* **Страницу или раздел «Журнал»** с минимум тремя постами (новостями, блоками) о прогрессе работы.
* **Страницу «Ресурсы»** со ссылками на полезные материалы (ссылки на организацию-партнёра, сайты и статьи, позволяющие лучше понять суть проекта).
* Оформите страницы сайта графическими материалами (фотографиями, схемами, диаграммами, иллюстрациями) и другой медиа информацией (видео).
* **Ожидаемое время:** изучение и настройка — 10–14 часов, дизайн и наполнение — 4–8 часов.

1. **Взаимодействие с организацией-партнёром:**

* Организуйте взаимодействие с партнёрской организацией (визит, онлайн-встреча или стажировка).
* Участвуйте в профильных мероприятиях по тематике проекта и профилю организации-партнёра (конференции, выставки, митапы, семинары, хакатоны и др.).
* **Уточнение:** Взаимодействие осуществляется через куратора проекта по проектной деятельности, закреплённого за вашим проектом, и ответственного по проектной практике, закреплённого за учебной группой.
* Напишите отчёт в формате Markdown с описанием опыта, полученных знаний и связи с проектом. Отчёт добавьте в репозиторий и на сайт.
* **Важно:** Стажировки и экскурсии в организации-партнёры будут приниматься к зачёту и учитываться при оценке, что мотивирует к активному участию.
* **Ожидаемое время:** взаимодействие — 4 часа, написание отчёта — 4 часа.

1. **Отчёт по практике**

* Составьте отчёт по проектной (учебной) практике на основании шаблона (структуры), размещённого в папке reports. Шаблон (структура) приведён в файле [practice\_report\_template.docx](https://github.com/Tytyty-ty/PracticeMosPolyTech/blob/main/task/reports/practice_report_template.docx).
* Разместите отчёт в репозитории в папке reports с именем «Отчёт.docx» или «report.docx».
* Сформируйте PDF-версию отчёта и также разместите её в папке reports в репозитории.
* Загрузите оба файла отчёта (DOCX и PDF) в СДО (LMS) в курсе, который будет указан ответственным за проектную (учебную) практику.

# **Вариативная часть задания**

По решению ответственного за проектную (учебную) практику студентам назначается одно из следующих вариативных заданий. Студенты могут направить ответственному свои пожелания по распределению.

**1. Кафедральное индивидуальное отдельное задание**

* Выполните все задачи базовой части.
* Выполните кафедральное индивидуальное отдельное задание.
* Интегрируйте результаты индивидуального задания и отчёт по нему в репозиторий и сайт, созданные в базовой части.
* **Ожидаемое время:** 32–40 часов.

**2. Практическая реализация технологии**

* Выполните все задачи базовой части.
* Для достижения объёма в 72 часа выберите один из следующих проектов:

1. Выберите любую технологию (тематику) из списка, представленного в репозитории [codecrafters-io/build-your-own-x](https://github.com/codecrafters-io/build-your-own-x). По согласованию с ответственными за практику можно использовать другой источник проектов.
2. Согласуйте внутри команды выбранную тему. Выберите стек технологий (подсказки также есть в репозитории).
3. Проведите исследование: изучите, как создать выбранную технологию с нуля, воспроизведите практическую часть.
4. Создайте подробное описание в формате Markdown, включающее:
   * Последовательность действий по исследованию предметной области и созданию технологии.
   * Напишите техническое руководство по созданию этой технологии, ориентированное на начинающих.
   * Включите в руководство:
     + Пошаговые инструкции.
     + Примеры кода.
   * Иллюстрации (картинки, диаграммы, схемы) в количестве от 3 до 10 штук, вставленные в текст для наглядности.
   * Поместите результаты исследования и руководства в общий Git-репозиторий.
5. Создайте техническое руководство или туториал по созданию проекта на выбранную тему. Для визуализации архитектуры, процессов и прочего используйте разные типы диаграмм UML, схемы, графики, таблицы.
6. Сделайте модификацию проекта согласно полученным знаниям и навыкам в течение года (творческий пункт, самостоятельно выбираете в какой части модифицировать). Описать в технической документации модификации.
7. Сделайте видео презентацию выполненной работы (цель, задачи, как решали, демонстрация работоспособного результата).
8. Задокументируйте проект в репозитории в формате Markdown и представьте его на сайте в формате HTML.
9. Подготовить финальный отчет (в хронологической последовательности опишите этапы работы, отдельно должны быть представлены индивидуальные планы каждого участника).

* **Пример 1:**
  + Для технологии «собственный интерпретатор» опишите этапы изучения синтаксиса, парсинга и выполнения кода, добавив схему работы интерпретатора и примеры кода.
* **Пример 2:**
  + Для технологии «собственный HTTP-сервер» создайте руководство с шагами по настройке сокетов, обработке запросов и отправке ответов, дополнив текст схемой взаимодействия клиент-сервер.
* **Ожидаемое время:** 32–40 часов.

# **Вклад в открытый проект на GitHub или GitVerse**

* Выполните все задачи базовой части.
* Найдите открытый проект на GitHub или GitVerse, имеющий не менее 50 звёзд (Stars), изучите его код и внесите вклад (например, исправьте баг или добавьте новую функцию).
* Документируйте свой вклад в Markdown и добавьте описание в репозиторий.
* **Ожидаемое время:** 32–40 часов.

# **Заключение**

Проведенная работа над проектом Ecollapse позволила создать прочный фундамент для разработки инновационной survival-игры с уникальными механиками. На текущем этапе успешно реализованы ключевые технологические решения: система процедурной генерации миров на основе алгоритмов шума Перлина и клеточных автоматов, динамическая экосистема с реалистичными цепочками взаимодействий, а также глубокая система эволюции персонажа. Эти элементы в совокупности формируют принципиально новый игровой опыт в жанре survival, что подтверждается положительными отзывами тестовой аудитории.

Организация разработки выстроена максимально эффективно благодаря четкой дорожной карте, которая определяет приоритеты дальнейшего развития проекта. Особое внимание уделено балансу между сложностью выживания и удовольствием от исследования, что является ключевым фактором для успеха в данном жанре.

На текущем этапе проект готов к переходу в фазу активной доработки и полировки перед коммерческим релизом. Основные направления дальнейшей работы включают расширение контентной базы (новые биомы, виды существ), реализацию кооперативного режима, финальную балансировку игровых систем и подготовку полноценной маркетинговой кампании. Достигнутые результаты и имеющийся задел позволяют с уверенностью прогнозировать успешный выход Ecollapse на рынок и его востребованность среди целевой аудитории.

# **Литература и интернет-ресурсы**

1. Введение в CSS верстку: <https://developer.mozilla.org/ru/docs/Learn_web_development/Core/CSS_layout/Introduction>
2. DevTools для «чайников»: <https://habr.com/ru/articles/548898/>
3. Элементы HTML: <https://developer.mozilla.org/ru/docs/Web/HTML/Element>
4. Основы HTML: <https://developer.mozilla.org/ru/docs/Learn_web_development/Getting_started/Your_first_website/Creating_the_content>
5. Основы CSS: <https://developer.mozilla.org/ru/docs/Web/CSS>
6. <https://doka.guide/>
7. Официальная документация Git: <https://git-scm.com/book/ru/v2>
8. <https://skillbox.ru/media/code/chto_takoe_git_obyasnyaem_na_skhemakh/>
9. Бесплатный курс на Hexlet по Git: <https://ru.hexlet.io/courses/intro_to_git>
10. Уроки по Markdown: <https://ru.hexlet.io/lesson_filters/markdown>

# **Информация для сведения**

* Общие задачи занимают 32–40 часов, дополнительные проекты добавляют 32–40 часов, что в сумме соответствует 72 часам.
* Выбор между индивидуальной и групповой работой (до 3 человек) позволяет адаптировать задание под ваши предпочтения.
* Возможность использовать как GitHub, так и GitVerse обеспечивает гибкость и доступ к современным инструментам.
* Исследовательские проекты и технические руководства, основанные на технологиях из [codecrafters-io/build-your-own-x](https://github.com/codecrafters-io/build-your-own-x), помогут развить навыки анализа, программирования и документирования.