

Федеральное государственное автономное образовательное учреждение высшего  
образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет «Факультет информационных технологий»

Кафедра «Информатика и вычислительная техника»

Направление подготовки/ специальность: 09.03.02 «Информационные системы и  
технологии»

## ОТЧЕТ

по проектной практике

Студент: Патаруева Анастасия Группа: 241-336

Место прохождения практики: Московский Политех, кафедра информатики и  
информационных технологий

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

Руководитель практики: \_\_\_\_\_

Москва 2025

# ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ .....	2
Введение .....	3
1. Общая информация о проекте.....	3
Название проекта .....	3
Цели и задачи проекта .....	3
2. Общая характеристика деятельности организации (заказчика проекта) .....	3
Наименование заказчика .....	3
Организационная структура.....	4
Описание деятельности .....	4
3. Описание задания по проектной практике .....	5
Введение .....	5
Базовая часть задания .....	5
Вариативная часть задания.....	8
4. Описание достигнутых результатов по проектной практике .....	12
Сайт .....	12
Телеграмм-бот на Python.....	13
Встреча с партнерами .....	16
Заключение .....	17
Источники.....	18

# **Введение**

## **1. Общая информация о проекте**

### **Название проекта**

Проект группы проектов игровой индустрии «Семилучье».

### **Цели и задачи проекта**

Цель проекта: выпуск в релиз первого уровня игры, демонстрация возможностей.

Задачи:

- разработка концепции игры (определить жанр, целевую аудиторию и основные механики игры);
- планирование (составить план проекта с указанием сроков, этапов);
- гейм-дизайн (создание игрового мира и персонажей);
- левел-дизайн (карты уровней, последовательность, препятствия, мини-игры и квесты, сложность);
- разработка и интеграция графики (выбрать игровой движок, реализовать основные механики игры (движение, взаимодействие, физика), интегрировать спрайты и анимации в игру);
- UI/UX-дизайн (пользовательский интерфейс);
- написание сценария и диалогов;
- тестирование и оптимизация.

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

### **Наименование заказчика**

Московский Политех (Федеральное государственное автономное образовательное учреждение высшего образования «Московский Политехнический Университет»)

### **Организационная структура**

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования «Московский Политехнический Университет» (Московский Политех)

### **Описание деятельности**

Заказчиком проекта выступает образовательное учреждение, в рамках которого реализуется проектная практика. Основной вид деятельности — образовательная, с акцентом на развитие прикладных компетенций студентов в рамках проектной работы.

### **3. Описание задания по проектной практике**

#### **Введение**

Задание на проектную (учебную) практику разработано для студентов первого курса, обучающихся по направлениям подготовки, связанным с информационными технологиями и информационной безопасностью. Трудоёмкость практики составляет 72 академических часа. Задание может выполняться индивидуально или в составе группы до 3 человек. Для управления версиями будет использоваться Git, для написания документации — Markdown, а для создания статического веб-сайта — языки разметки HTML и CSS, но опционально допускается использовать генераторы статических сайтов, такие, как Hugo. В качестве платформы для размещения репозитория допустимо использовать как GitHub, так и GitVerse, что обеспечивает гибкость в выборе инструментов. Также предусмотрено взаимодействие с организациями-партнёрами, включая стажировки, которые будут приниматься к зачёту при оценке.

Задание состоит из двух частей. Первая часть является общей и обязательной для всех студентов. Вторая часть вариативная. Задание на вторую часть может быть получено от:

- ответственного за проектную (учебную) практику на выпускающей кафедре;
- куратора проекта по «Проектной деятельности», но должно быть согласовано с ответственным за проектную (учебную) практику на выпускающей кафедре.

#### **Базовая часть задания**

1. Настройка Git и репозитория:

- Создайте личный или групповой репозиторий на 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.

## 2. 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.

## 3. 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.

- 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.

#### 4. 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.

## 5. 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.

- 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 tutorial 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. ЗадOCUMENTИРУЙТЕ 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.

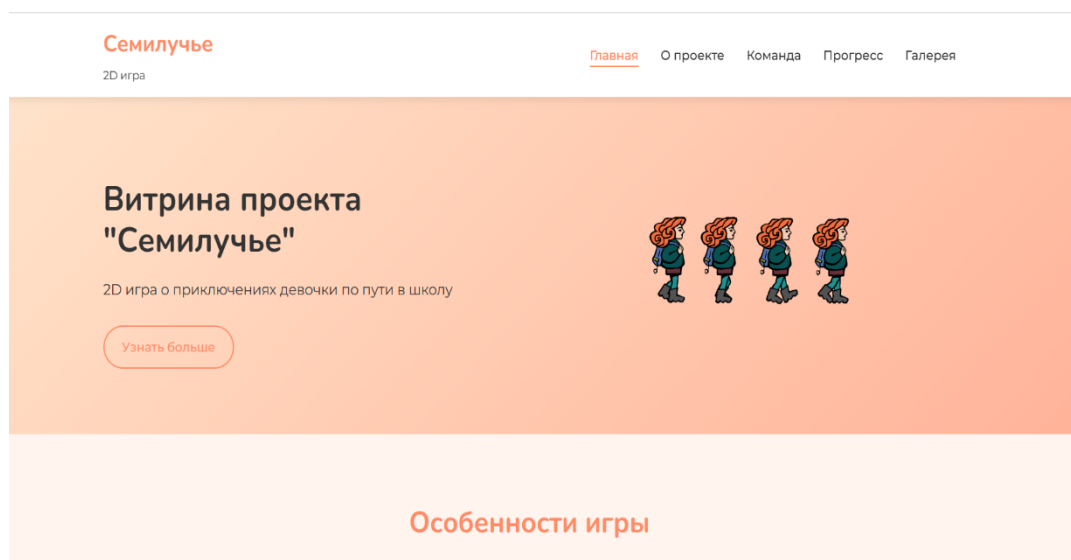
- Ожидаемое время: 32–40 часов.

## 4. Описание достигнутых результатов по проектной практике

### Сайт

Создан сайт-визитка для игрового проекта «Семилучье». Реализованы такие этапы как:

- Проектирование общего дизайна сайта;
- Проектирование дизайна главной страницы сайта;
- Проектирование дизайна страницы «Галерея»;
- Проектирование дизайна страницы «О проекте»;
- Проектирование дизайна страницы «Прогресс»;
- Проектирование дизайна страницы «О команде»;
- Реализация стилей для спроектированного дизайна в формате CSS;
- Проектирование связи между страницами сайта;
- Реализация главной страницы сайта;
- Реализация страницы «Галерея»;
- Реализация страницы «О проекте»;
- Реализация страницы «Прогресс»;
- Реализация страницы «О команде»;



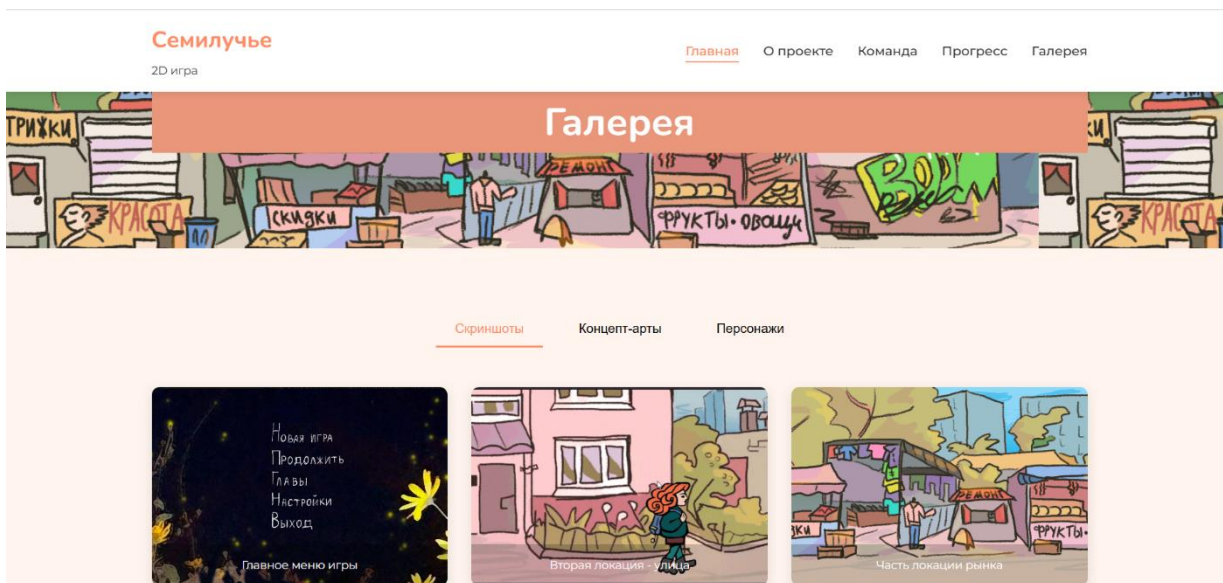


Рисунок 1. Главная страница

Рисунок 2. Галерея

Сайт расположен по ссылке: <https://windfrom.github.io/gallery.html>

## Телеграмм-бот на Python

Создан модифицированный телеграм-бот с мини-игрой «Быки и коровы», на основе предложенного для изучения материала;

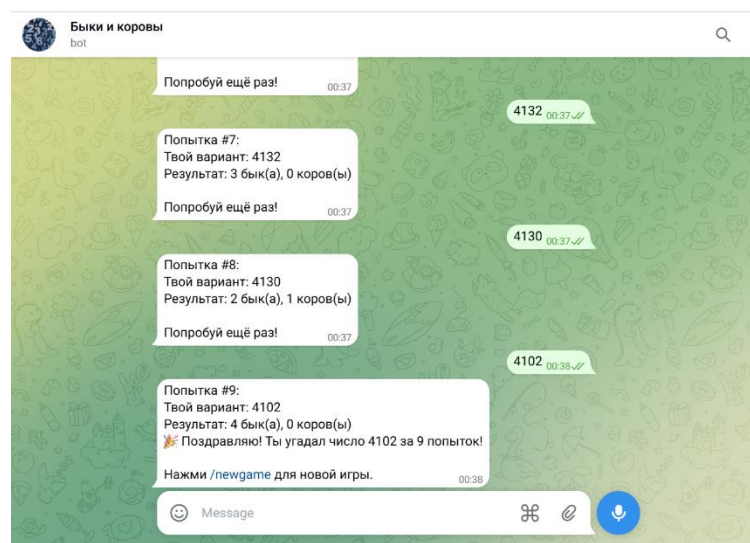


Рисунок 3. Демонстрация бота

## Пример кода:

```
import random
import telebot
from telebot import types

TOKEN = "xxx"
bot = telebot.TeleBot(TOKEN)

games = {}

def generate_secret_number():
    """Генерирует 4-значное число с уникальными цифрами"""
    digits = list('0123456789')
    random.shuffle(digits)

    while True:
        secret = ''.join(digits[:4])
        if secret[0] != '0':
            return secret
        random.shuffle(digits)

@bot.message_handler(commands=['start', 'help'])
def send_welcome(message):
    """Приветственное сообщение и правила игры"""
    markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
    markup.add(types.KeyboardButton('/newgame'))

    bot.send_message(
        message.chat.id,
        "🐮🐮 Игра 'Быки и коровы'!\n\n"
        "Я загадал 4-значное число с неповторяющимися цифрами.\n"
        "Твоя задача - угадать его.\n\n"
        "• Бык - правильная цифра на правильном месте\n"
        "• Корова - правильная цифра на неправильном месте\n\n"
        "Нажми /newgame чтобы начать!",
        reply_markup=markup
    )

@bot.message_handler(commands=['newgame'])
def new_game(message):
    """Начинает новую игру"""
    chat_id = message.chat.id
    games[chat_id] = {
        'secret': generate_secret_number(),
        'attempts': 0
    }
    print(f"New game for {chat_id}, secret: {games[chat_id]['secret']}") # Для
отладки
```

```

    bot.send_message(chat_id, "🔢 Я загадал новое 4-значное число! Попробуй угадать:")

@bot.message_handler(func=lambda message: True)
def handle_guess(message):
    """Обрабатывает попытки угадать число"""
    chat_id = message.chat.id
    user_input = message.text.strip()

    if chat_id not in games:
        bot.send_message(chat_id, "Нажми /newgame чтобы начать игру!")
        return

    if not (user_input.isdigit() and len(user_input) == 4 and len(set(user_input)) == 4):
        bot.send_message(chat_id, "❌ Нужно ввести 4-значное число с разными цифрами!")
        return

    if user_input[0] == '0':
        bot.send_message(chat_id, "❌ Число не может начинаться с 0!")
        return

    secret = games[chat_id]['secret']
    games[chat_id]['attempts'] += 1

    bulls = sum(1 for i in range(4) if user_input[i] == secret[i])
    cows = sum(1 for d in user_input if d in secret) - bulls

    attempt_text = f"Попытка #{games[chat_id]['attempts']}:\n"
    result_text = f"Твой вариант: {user_input}\nРезультат: {bulls} бык(а), {cows} коров(ы)\n"

    if bulls == 4:
        win_text = (f"🎉 Поздравляю! Ты угадал число {secret} "
                    f"за {games[chat_id]['attempts']} попыток!\n\n"
                    "Нажми /newgame для новой игры.")
        del games[chat_id]
        bot.send_message(chat_id, attempt_text + result_text + win_text)
    else:
        bot.send_message(chat_id, attempt_text + result_text + "\nПопробуй ещё раз!")

if __name__ == '__main__':
    print("Бот запущен...")
    bot.infinity_polling()

```

Также созданы два руководства по созданию бота-астролога и модифицированного бота быки-коровы. Руководства загружены на github.

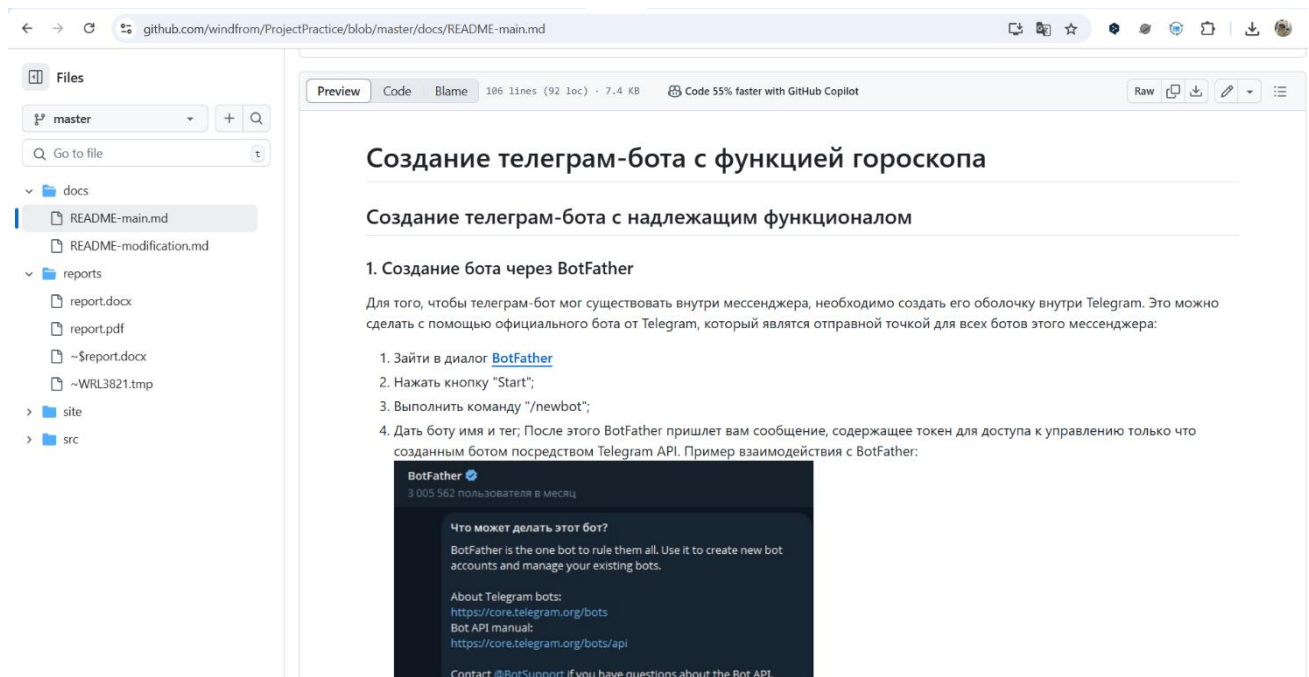


Рисунок 4. Демонстрация github

## Встреча с партнерами

Посещение форума Арт-Мастерс, где дополнительно были получены знания по дизайну, которые помогли при проектировании сайта. Так же посещение мероприятий от Московского Политеха, который является заказчиком проекта: Выставка проектов, ДОД и прочее.



## **Заключение**

Проектная практика помогла реализовать полученные за этот семестр знания, а также изучить новые вещи, которые пригодятся в будущем.

Результат проектной практики расположен по адресу:  
<https://github.com/windfrom/ProjectPractice/tree/master> .

## Источники

Разработка игр: [Электронный ресурс]. URL: <https://gamedev.ru/> (Дата обращения: 05.05.2025).

DTF.Дизайн: [Электронный ресурс]. URL: <https://dtf.ru/tag/%D0%B4%D0%B8%D0%B7%D0%B0%D0%B9%D0%BD> (Дата обращения: 05.05.2025).

Сид Мейр Жизнь в мире компьютерных игр: [Электронный ресурс]. URL: <https://flibusta.su/book/19090-zhizn-v-mire-kompyuternyih-igr/> (Дата обращения: 05.05.2025).

Гибсон Бонд Джереми Unity и C#. Геймдев от идеи до реализации[Электронный ресурс]. URL: <https://www.codelibrary.info/books/c-sharp/unity-i-c-sharp-gejmdev-ot-idei-do-realizatsii> (Дата обращения: 05.05.2025).