Федеральное государственное автономное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий  
Кафедра «Информационные системы и технологии»

Направление подготовки/специальность: 09.03.02 Информационные системы и технологии

ОТЧЕТ

по проектной практике

Студент: Кравцов Иван Васильевич\_\_\_\_\_\_\_\_\_Группа: 241-331

Место прохождения практики: Московский Политех, Кафедра информационных технологий

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

Руководитель практики: \_\_\_\_Привалов Вячеслав Анатольевич\_\_\_\_\_\_

Москва 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

1. Общая информация о проекте:

* Название проекта
* Цели и задачи проекта

1. Общая характеристика деятельности организации *(заказчика проекта)*

* Наименование заказчика
* Организационная структура
* Описание деятельности

1. Описание задания по проектной практике
2. Описание достигнутых результатов по проектной практике

ЗАКЛЮЧЕНИЕ *(выводы о проделанной работе и оценка ценности выполненных задач для заказчика)*

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

ПРИЛОЖЕНИЯ *(при необходимости)*

Введение

В рамках данной практики была поставлена задача: реализовать как базовую, так и вариативную часть проектной деятельности. Базовая часть включала в себя оформление репозитория, разработку сайта с описанием проекта, участников, этапов реализации, а также ведение прогресса. Вариативная часть заключалась в самостоятельной реализации технологии — HTTP-сервера — и создании обучающего веб-приложения с интерактивными элементами.

В данном отчёте приведено полное описание выполненной работы: от постановки цели и задач до реализации серверной логики, фронтенд-части и итогов практики.

1. Цель и задачи практики

**Цель практики.**  
Получение практического опыта в проектировании, разработке и документировании IT-систем с использованием веб-технологий и собственной серверной реализации.

**Задачи:**

1. Изучить и применить Git и GitHub для ведения репозитория проекта.
2. Разработать сайт, описывающий цели, задачи, участников и этапы реализации проекта.
3. Оформить документацию в формате Markdown и DOCX.
4. Реализовать собственный HTTP-сервер на языке Python.
5. Разработать обучающее мини-приложение: словарь и викторину по IT-терминам.
6. Подготовить отчёт о проделанной работе.

2. Общая характеристика деятельности организации (заказчика проекта)

* **Наименование заказчика:** Московский Политехнический Университет
* **Организационная структура:**
  + Ректор: Миклушевский Владимир Владимирович
  + Декан факультета информационных технологий: Демидов Дмитрий Григорьевич
  + Заведующий кафедрой «Информатика и информационные технологии»: Булатников Евгений Владиславович
* **Описание деятельности:** Московский Политехнический Университет осуществляет образовательную деятельность по программам высшего образования, научные исследования в области технических наук, а также разработку и внедрение передовых технологий в образовательный процесс. Университет активно взаимодействует с промышленными предприятиями, государственными учреждениями и научными организациями.

3. Организация работы и инструменты

Вся работа была выполнена индивидуально. Для управления кодом использовалась система контроля версий Git, репозиторий размещался на платформе GitHub.

Проектная практика была разделена на два больших этапа:

* выполнение **базовой части**, включающей оформление репозитория и создание статического сайта;
* реализация **вариативной части** — веб-приложения с серверной частью и обучающим контентом.

**Используемые инструменты и технологии:**

* Язык программирования: Python (для сервера), HTML, CSS, JavaScript (для фронтенда)
* Редактор кода: Visual Studio Code
* Локальный запуск: встроенный HTTP-сервер Python
* Документация: Markdown, Word
* Система контроля версий: Git

4. Базовая часть практики

4.1 Настройка репозитория

В начале практики был создан репозиторий на GitHub по шаблону practice-2025-1, в котором были предусмотрены каталоги для кода (src/), сайта (site/), документации (docs/), отчётов (reports/) и заданий (task/). Все изменения в ходе практики фиксировались с помощью Git и сопровождались осмысленными комментариями.

4.2 Статический сайт

В папке site/ были разработаны следующие HTML-страницы:

* **index.html** — главная страница с навигацией;
* **about.html** — описание проекта;
* **members.html** — раздел с указанием участников (в данной реализации — один студент);
* **journal.html** — журнал выполнения работы;
* **resources.html** — ссылки на использованные источники.

Сайт был оформлен с использованием CSS (assets/style.css), включая навигацию, стили заголовков, списков и таблиц.

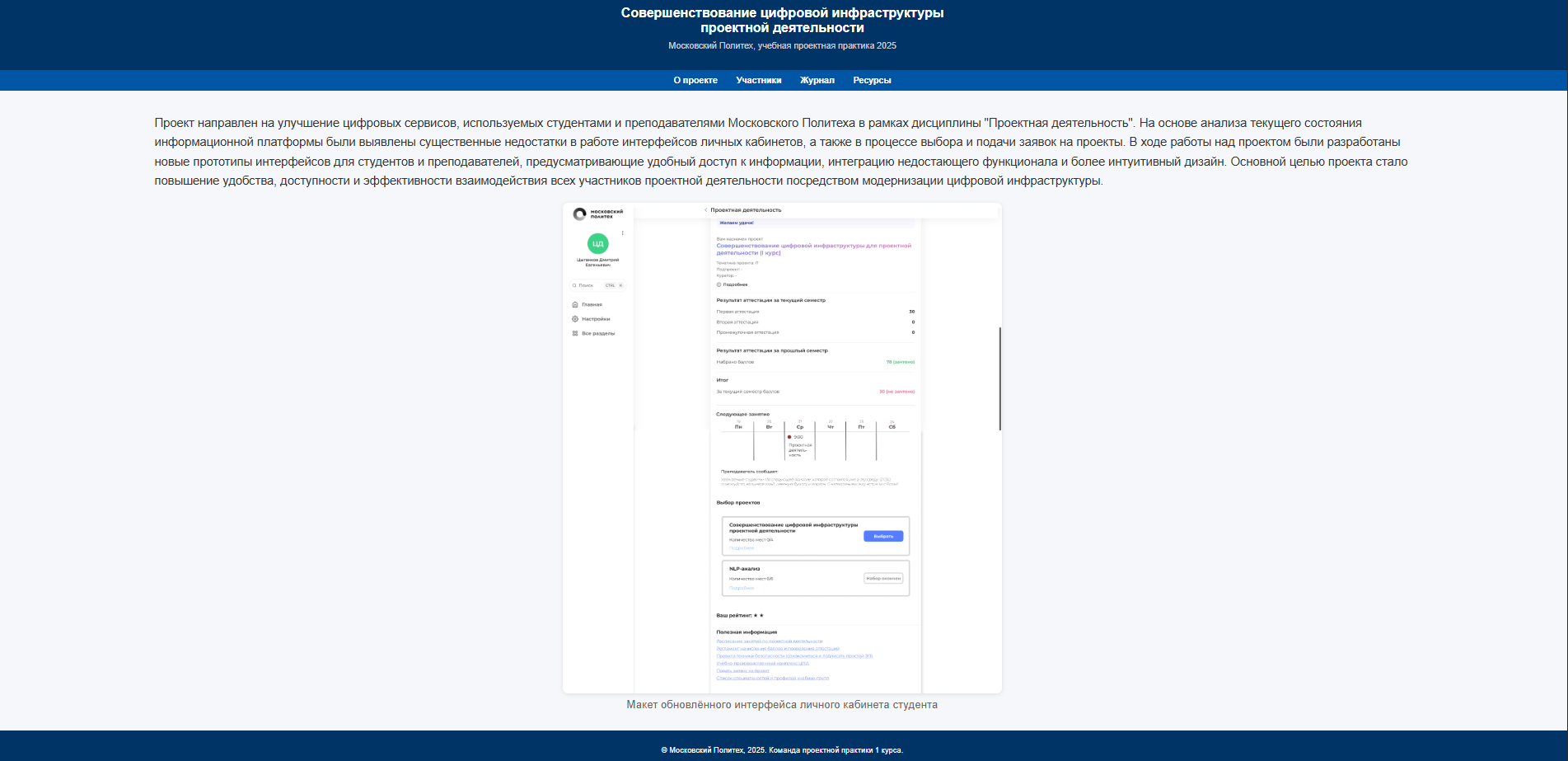


Рисунок 1 Файл Index.html

5. Вариативная часть — реализация HTTP-сервера

5.1 Постановка задачи

Для вариативной части было выбрано задание: **реализовать собственную технологию**. В качестве технологии выбран **HTTP-сервер на языке Python**, обслуживающий простой обучающий сайт.

Такой подход позволяет:

* продемонстрировать базовые навыки работы с сокетами и протоколом HTTP;
* реализовать простейший сервер без сторонних библиотек;
* обслуживать HTML, CSS и JS-файлы локально.

5.2 Реализация сервера

Сервер был реализован в файле server.py на основе стандартной библиотеки Python http.server. Он обслуживает статические файлы из каталога src/.

Пример запуска сервера:

***python server.py***

После запуска сервер становится доступен по адресу:

***http://localhost:8080***

5.3 Структура приложения

Приложение состоит из следующих компонентов:

* **index.html** — навигационная страница;
* **dictionary.html** — словарь IT-терминов;
* **quiz.html** — интерактивная викторина;
* **about.html** — описание проекта;
* **style.css** — общий стиль;
* **quiz.js** — логика проверки ответов и генерации вопросов.

6. Функциональность обучающего сайта

6.1 Словарь терминов

На странице dictionary.html представлен список базовых IT-терминов на английском языке с переводами. В таблице содержатся такие понятия, как API, Bug, Git, Loop, Backend и другие. Каждая строка таблицы содержит термин и его краткое пояснение.

Это позволяет студенту быстро ознакомиться с основными понятиями, применяемыми в других разделах сайта.



Рисунок 2Страница со словарем

6.2 Викторина

Страница quiz.html представляет собой интерактивную викторину, состоящую из 10 заранее подготовленных вопросов. Каждый вопрос имеет 3 варианта ответа, из которых пользователь должен выбрать один. При каждом открытии страницы случайным образом отбираются 5 уникальных вопросов, тем самым обеспечивается разнообразие и реиграбельность.

Если пользователь отвечает правильно на все 5 вопросов — автоматически загружается новая порция вопросов. Если допущена хотя бы одна ошибка — ответы остаются, и пользователю показывается количество правильных ответов.

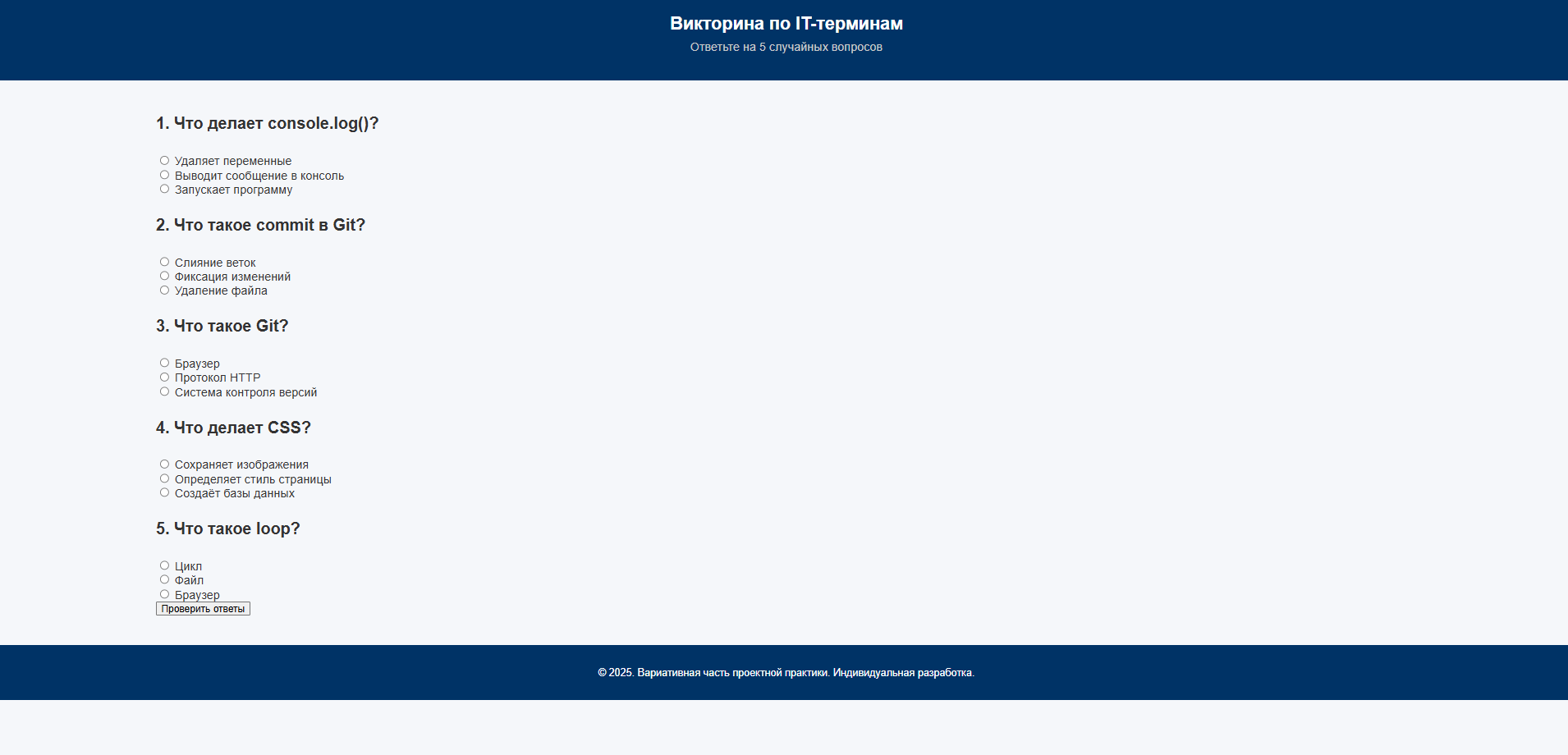


Рисунок 3 Интерактивная викторина со словами

6.3 Навигация и структура

Все страницы объединены общей навигацией, выполненной с использованием HTML и CSS. Страницы имеют единый стиль, что создаёт визуальную целостность и облегчает восприятие информации.

Дополнительно реализованы страницы:

* index.html — навигационная точка входа;
* about.html — объясняет цель проекта, технологии и способы запуска.

7. Особенности реализации и сложности

7.1 Отказ от PHP и готовых серверов

В отличие от типичного подхода с использованием Apache + PHP (например, через Open Server), реализация в рамках практики была выполнена **без готовых серверных решений**. Это позволило глубже понять принципы HTTP, внутреннюю структуру обработки запросов, работу с MIME-типами и статическими ресурсами.

7.2 Работа с JavaScript

Логика викторины была реализована с нуля. Сложности вызвали:

* рандомизация вопросов без повторов;
* генерация DOM-элементов через JS;
* проверка правильных ответов;
* управление повторной генерацией при 100% правильных ответах.

Тем не менее, все задачи были успешно решены, а код структурирован и сохранён в assets/quiz.js.

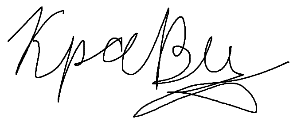
7.3 Поддержка кроссбраузерности

Проект тестировался в браузерах Google Chrome и Microsoft Edge. Особых несовместимостей выявлено не было, что говорит о корректной базовой реализации HTML/CSS/JS.

**Список литературы**

1. Бэнкс, А. HTML и CSS. Разработка и дизайн веб-сайтов / Джон Дакетт ; пер. с англ. — М. : Вильямс, 2014. — 512 с.
2. Бэнкс, А. JavaScript и jQuery. Интерактивная веб-разработка / Джон Дакетт ; пер. с англ. — М. : Вильямс, 2015. — 640 с.
3. Лутц, М. Изучаем Python / Марк Лутц ; пер. с англ. — СПб. : Символ-Плюс, 2021. — 928 с.
4. Барбаро, А. Git. Полное руководство / Скотт Шакон ; пер. с англ. — СПб. : Питер, 2019. — 384 с.
5. Хекслет. Основы HTML и CSS: онлайн-курс. [Электронный ресурс] — URL: https://ru.hexlet.io/programs/web-development-free.
6. Хекслет. Введение в JavaScript. [Электронный ресурс] — URL: <https://ru.hexlet.io/courses/js-basics>

*Подтверждаю, что отчёт выполнен лично и соответствует требованиям практики.*

**Кравцов Иван**  
**Дата:** \_\_23.05.2025\_\_\_\_\_\_\_\_\_ **Подпись:** \_\_\_\_\_\_\_\_\_\_\_\_