

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ. ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫШЕГО ОБРАЗОВАНИЯ

**«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)**

Факультет информационных
технологий Кафедра: «СМАРТ
технологий»

Направление подготовки/ специальность: 09.03.01 Информатика и
вычислительная техника

ОТЧЕТ

по проектной практике

Студент: Данихно Велес Иванович

Группа: 241–324

Место прохождения практики: Московский Политех, Центр проектной
деятельности

Отчет принят с оценкой _____ Дата _____

Руководитель практики: Сивцев А. О.

Москва 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. НАСТРОЙКА GIT И РЕПОЗИТОРИЯ (настройка Git и репозитория)	4
1.1. Изучение клонирования репозитория	4
1.2. Внесение изменений в файлы репозитория	7
1.3. Создание новой ветки	12
2. НАПИСАНИЕ ДОКУМЕНТОВ В MARKDOWN	14
2.1. Синтаксис Markdown.	14
2.1.1. Заголовки	14
2.1.2. Форматирование текста	14
2.1.3. Списки	15
2.1.4. Ссылки	15
2.1.5. Изображения	15
2.1.6. Цитаты	15
2.1.7. Код	16
2.2. Пример оформления файла в синтаксисе Markdown	16
3. СОЗДАНИЕ СТАТИЧЕСКОГО ВЕБ-САЙТА	19
3.1. Планирование и структура	19
3.2. HTML-верстка	23
3.3. Стилизация с помощью CSS	23
4. ВАРИАТИВНАЯ ЧАСТЬ	25
4.1. Генерация идей и разработка концепции макета	25
4.2. Поиск и анализ моделей зданий	26
4.3. Подбор материалов для основания макета	27
ЗАКЛЮЧЕНИЕ	29
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	30
ПРИЛОЖЕНИЕ А. Листинг файла Index.html	31
ПРИЛОЖЕНИЕ Б. Листинг файла style.css	33

ВВЕДЕНИЕ

Современная экосистема информационных технологий требует от специалистов не только глубоких теоретических знаний, но и уверенного владения практическими инструментами, которые являются стандартом в индустрии. Учебная (проектная) практика по теме «Проектная практика в информационных технологиях и информационной безопасности» была направлена именно на формирование таких ключевых навыков.

Главной целью работы стало получение практического опыта в жизненном цикле небольшого IT-проекта. Для этого были поставлены и выполнены следующие задачи:

1. **Настройка окружения:** Освоение системы контроля версий Git как основы для совместной работы и управления кодом.
2. **Документирование:** Применение языка разметки Markdown для создания понятной и структурированной технической документации.
3. **Разработка:** Создание функционального продукта — статического веб-сайта, что позволило интегрировать полученные навыки.

1. НАСТРОЙКА GIT И РЕПОЗИТОРИЯ

Система контроля версий Git и веб-платформа GitHub являются индустриальным стандартом для разработки программного обеспечения. Git позволяет отслеживать все изменения в файлах проекта, возвращаться к предыдущим версиям, создавать параллельные ветки для разработки нового функционала и эффективно работать в команде. GitHub, в свою очередь, предоставляет облачное хранилище для Git-репозитория, а также инструменты для совместной работы, такие как pull-запросы и обсуждение кода. В данном разделе отчета пошагово демонстрируется освоение базовых операций, необходимых для начала работы с любым IT-проектом.

1.1. Изучение клонирования репозитория.

Первым шагом в работе с любым существующим проектом является создание его локальной копии. В данном случае, поскольку мы начинали работу с предоставленного шаблона, был использован механизм импорта репозитория. Этот процесс позволяет создать в своем аккаунте полную копию другого репозитория, включая всю его структуру и историю.

Для оформления своей работы клонируем шаблон репозитория Московского политеха.

Для этого сначала скопируем имя шаблона.

Перейдём на репозиторий Московского политеха по ссылке: <https://github.com/mospol/practice-2025-1>.

Нажмём кнопку «<> Code», а затем на значок «копировать в буфер» (Рис. 1.1).

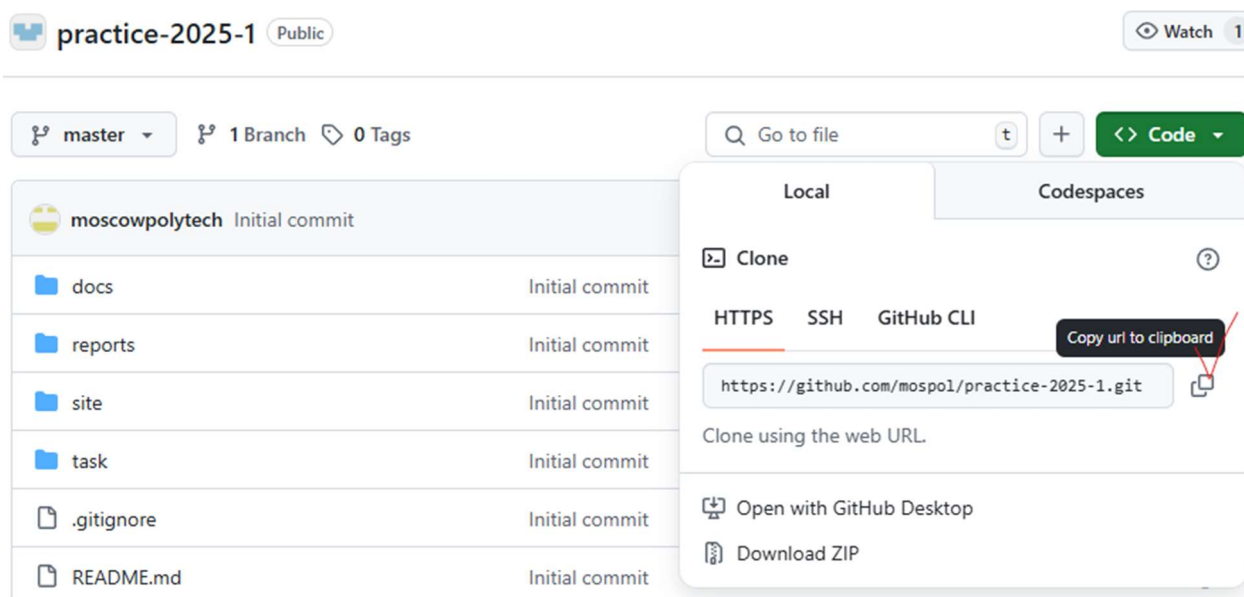


Рисунок 1.1 – Окно шаблона репозитория Московского политеха.

Далее заходим на свой GITHUB и нажимаем кнопку «+ » и выбираем «Import repository» (Рис. 1.2)

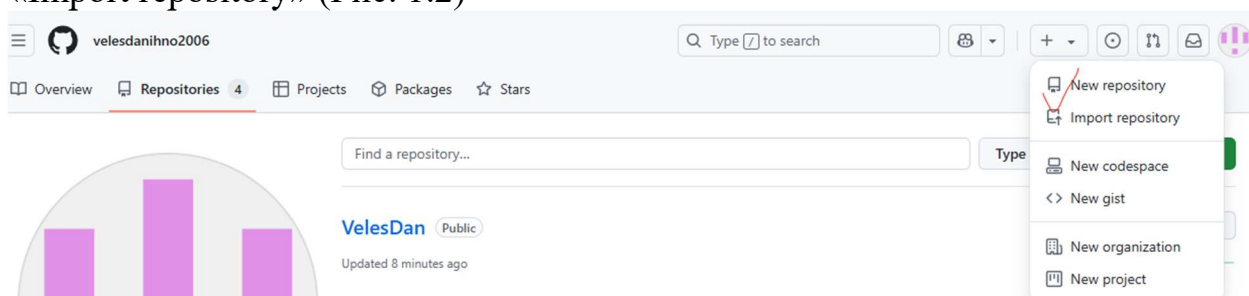


Рисунок 1.2 – GITHUB и «Import repository»

В появившемся окне прописываем путь из шаблонному репозиторию Московского политеха. Прописываем своё имя репозитория – в моём случае VelesDan. Устанавливаем доступность «Public». И нажимаем кнопку «Begin Import» (Рис. 1.3).

Import your project to GitHub

Import all the files, including revision history, from another version control system.

Required fields are marked with an asterisk (*).

Support for importing Mercurial, Subversion and Team Foundation Version Control (TFVC) repositories ended on April 12, 2024. For more details, see the [changelog](#).

Your source repository details

The URL for your source repository *

Learn more about [importing git repositories](#).

Please enter your credentials if required for cloning your remote repository.

Your username for your source repository

Your access token or password for your source repository

Your new repository details

Owner *


 velesdanihno2006 ▾

Repository name *

/

✔ VelesDan is available.

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

 You are creating a public repository in your personal account.

Cancel

Begin import

Рисунок 1.3 – Окно импорта репозитория.

Если клонирование прошло успешно, (Рис.1.4) то мы получим сообщение «Import complete!». И увидим, что репозиторий клонирован полностью из шаблона Московского политеха (Рис. 1.5).

Preparing your new repository

There is no need to keep this window open. We'll email you when the import is done.



Importing complete! Your new repository [velesdanihno2006/VelesDan](#) is ready.

Рисунок 1.4 – Успешный импорт

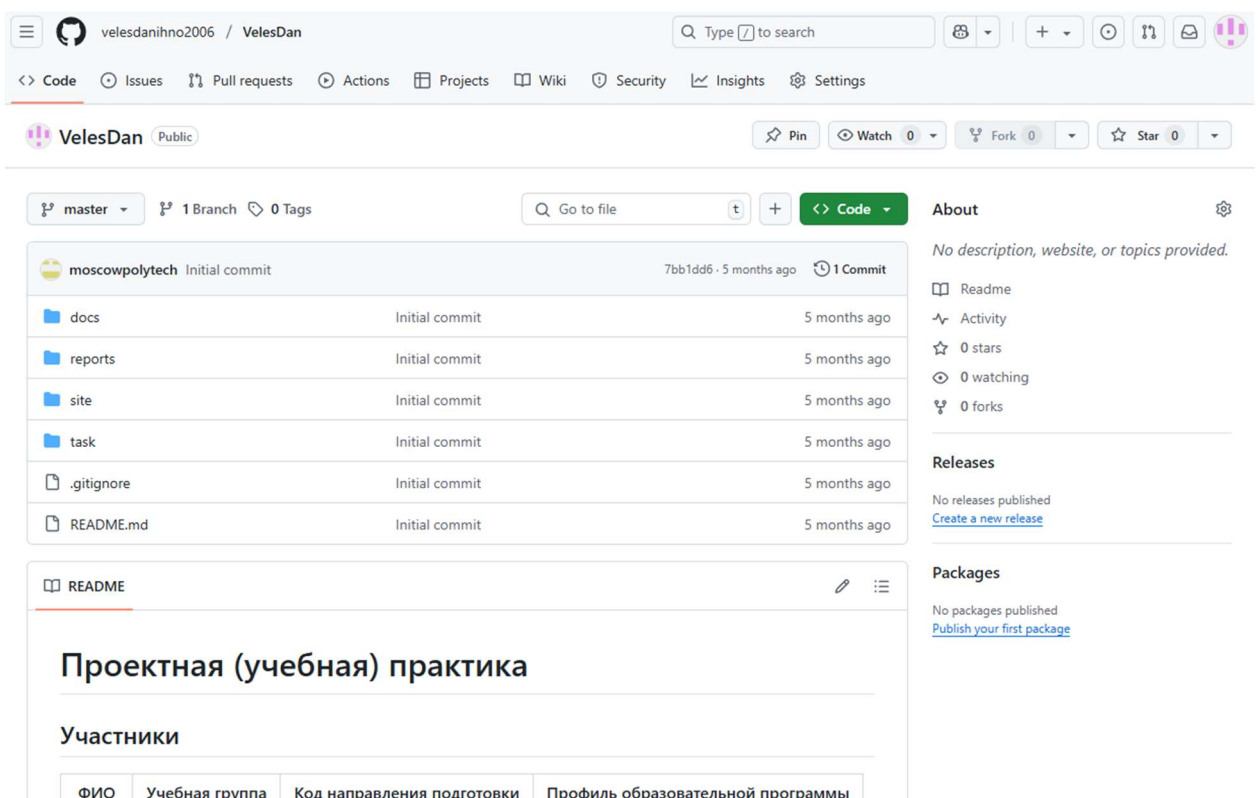


Рисунок 1.5 – Результат клонирования

1.2. Внесение изменений в файлы репозитория.

Основной цикл работы в Git включает в себя внесение изменений в файлы, их фиксацию (создание "коммита") и отправку в удаленный репозиторий. Для обеспечения контроля качества и совместного обсуждения изменений в командных проектах используется механизм Pull Request (запрос на слияние). Он позволяет предложить свои изменения для включения в основную ветку, предварительно получив одобрение от других участников или руководителя.

Для примера изменим файл README.md.

Для начала нажмём редактировать файл (карандашик напротив имени

файла)

(Рис.

1.6).

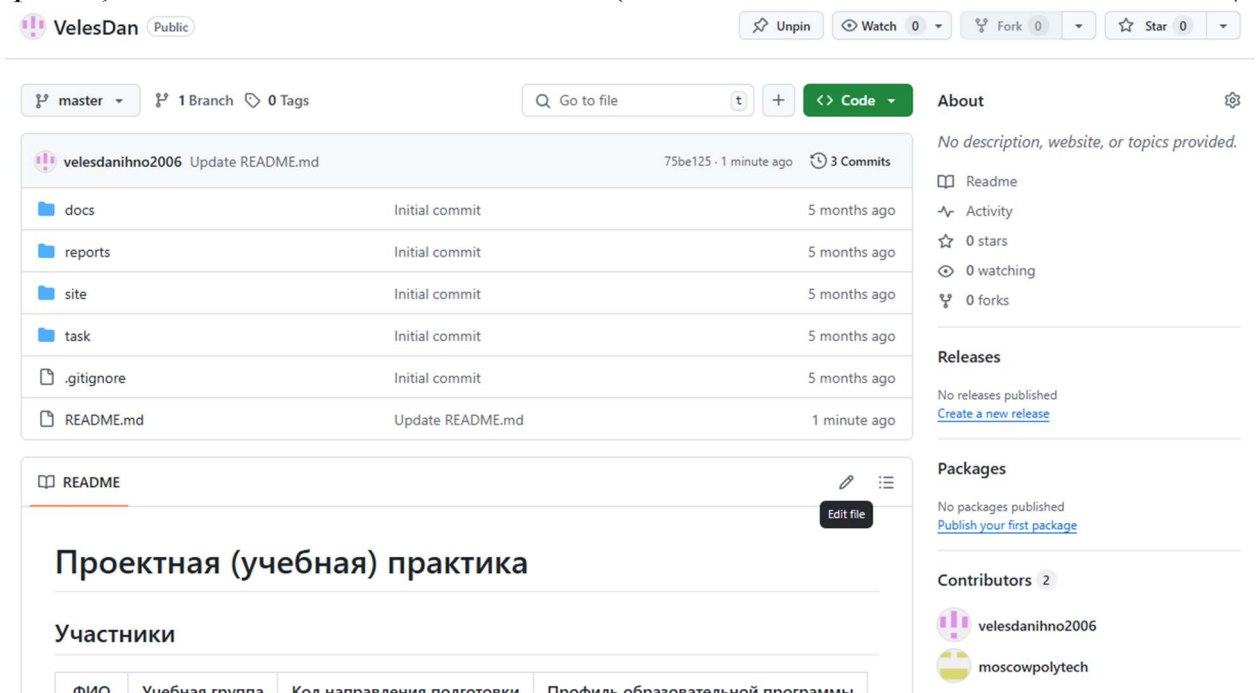


Рисунок 1.6 – Выбранный репозиторий

В открывшемся окне внесём изменения в файл. После внесения необходимых изменений нажмём кнопку «Commit changes...» (Рис. 1.7).

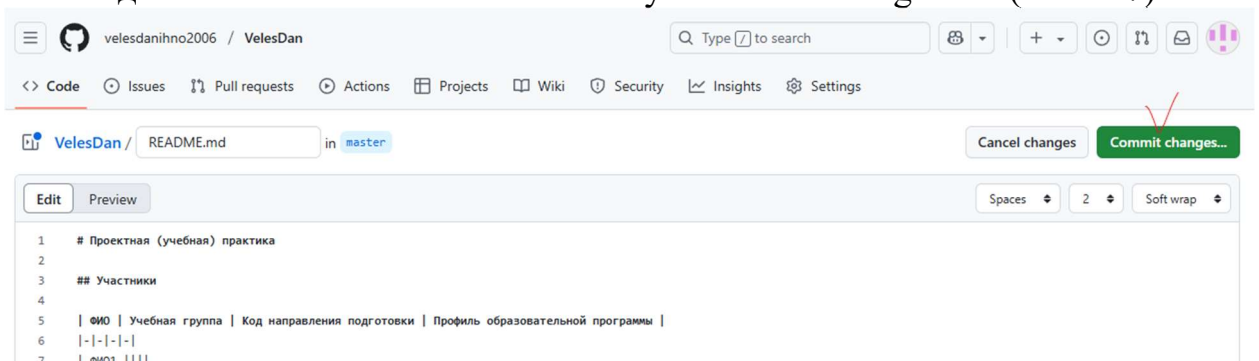


Рисунок 1.7 – Внесение изменений в репозиторий

Откроется окно с запросом оставить сообщение с описанием и выбор внизу: оставить изменение в главной ветке или создать новую ветку с пул запросом (Рис. 1.8).

Commit changes

Commit message

Update README.md

Extended description

Add an optional extended description...

☒ Commit directly to the main branch

☐ Create a new branch for this commit and start a pull request [Learn more about pull requests](#)

Cancel Commit changes

Рисунок 1.8 – Commit changes

Если выбрать создать новую ветвь, то будет предложено присвоить имя новой ветви. Если нажать «Propose changes», то откроется окно созданием запроса (Рис. 1.9).

Propose changes

Commit message

Update README.md

Extended description

Add an optional extended description...

☐ Commit directly to the main branch

☒ Create a new branch for this commit and start a pull request [Learn more about pull requests](#)

velesdanihno2006-patch-1

Cancel Propose changes

Рисунок 1.9 – Окно созданием запроса

Заполняем заголовок запроса, добавляем описание и нажимаем кнопку «Create pull request» (Рис. 1.10).

velesdanihno2006 / VelesDan

base: master ← compare: velesdanihno2006-patch-1

Add a title

Update README.md

Add a description

Write Preview H B I `<>`

Запрос на изменение файла.

Markdown is supported Paste, drop, or click to add files

Create pull request

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Рисунок 1.10 – Окно редактирования запроса

Откроется запрос, который мы можем принять и объединить с главной веткой нажав на кнопку «Merge pull request» (Рис. 1.11).

Update README.md #1

Open velesdanihno2006 wants to merge 1 commit into `master` from `velesdanihno2006-patch-1`

Conversation 0 Commits 1 Checks 0 Files changed 1

velesdanihno2006 commented 1 minute ago Owner ...

Запрос на изменение файла.

Update README.md Verified 1ed8781

No conflicts with base branch
Merging can be performed automatically.

Merge pull request You can also merge this with the command line. [View command line instructions.](#)

Add a comment

Write Preview **H B I**

Add your comment here...

Markdown is supported Paste, drop, or click to add files

Close pull request **Comment**

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

ProTip! Add `.patch` or `.diff` to the end of URLs for Git's plaintext views.

Рисунок 1.11 – Объединение запросов

Заполним сообщение и расширенное описание и, нажав кнопку «Confirm merge» получим подтверждение изменения файла в главной ветке (Рис. 1.12).

Commit message

Проверка изменения файла

Extended description

Update README.md

This commit will be authored by velesdanihno2006@gmail.com.

Confirm merge Cancel

Рисунок 1.12 – Расширяем описание

1.3. Создание новой ветки.

Ключевой концепцией в Git является ветвление (branching). Ветви позволяют вести разработку новых функций или исправлять ошибки в изолированной среде, не затрагивая стабильную основную ветку (master/main). Это предотвращает поломку рабочего кода и дает возможность нескольким разработчикам трудиться над разными задачами параллельно.

Для создания новой ветки необходимо выбрать вкладку «Branch» (Рис. 12).

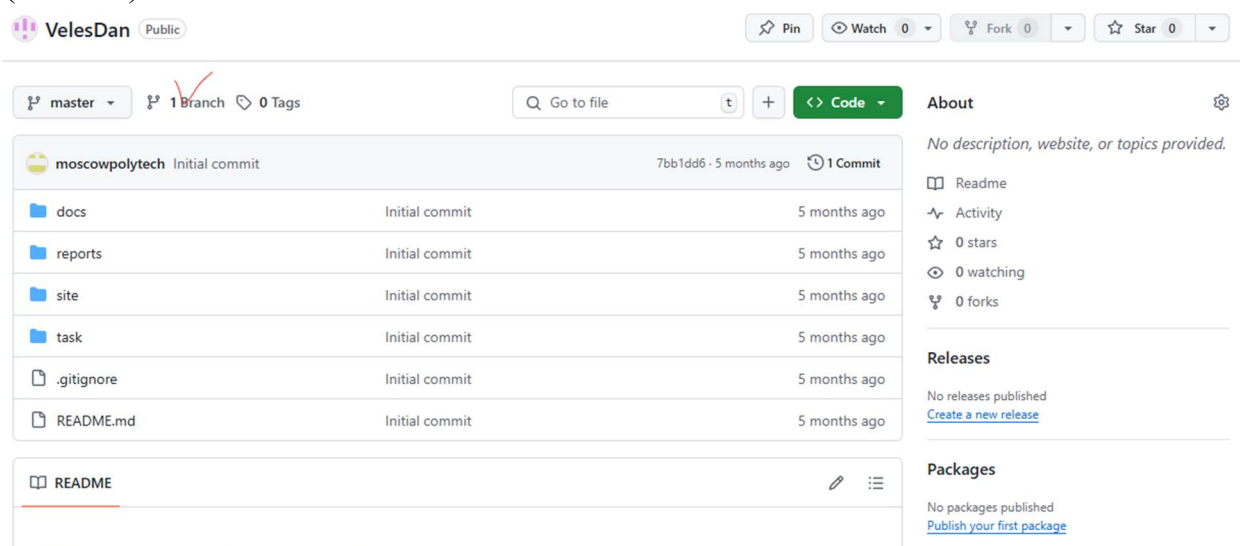


Рисунок 1.13 – Вкладка «Branch»

В открывшемся окне нажмем кнопку «New branch» (Рис. 1.14).

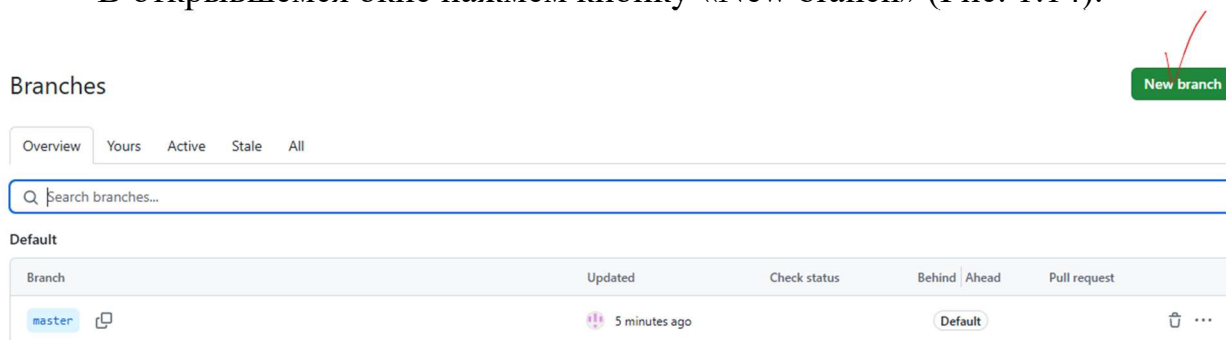


Рисунок 1.13 – Кнопка «New branch».

Придумаем имя новой ветке, и нажимаем на кнопку «Create new branch» (Рис. 1.15).

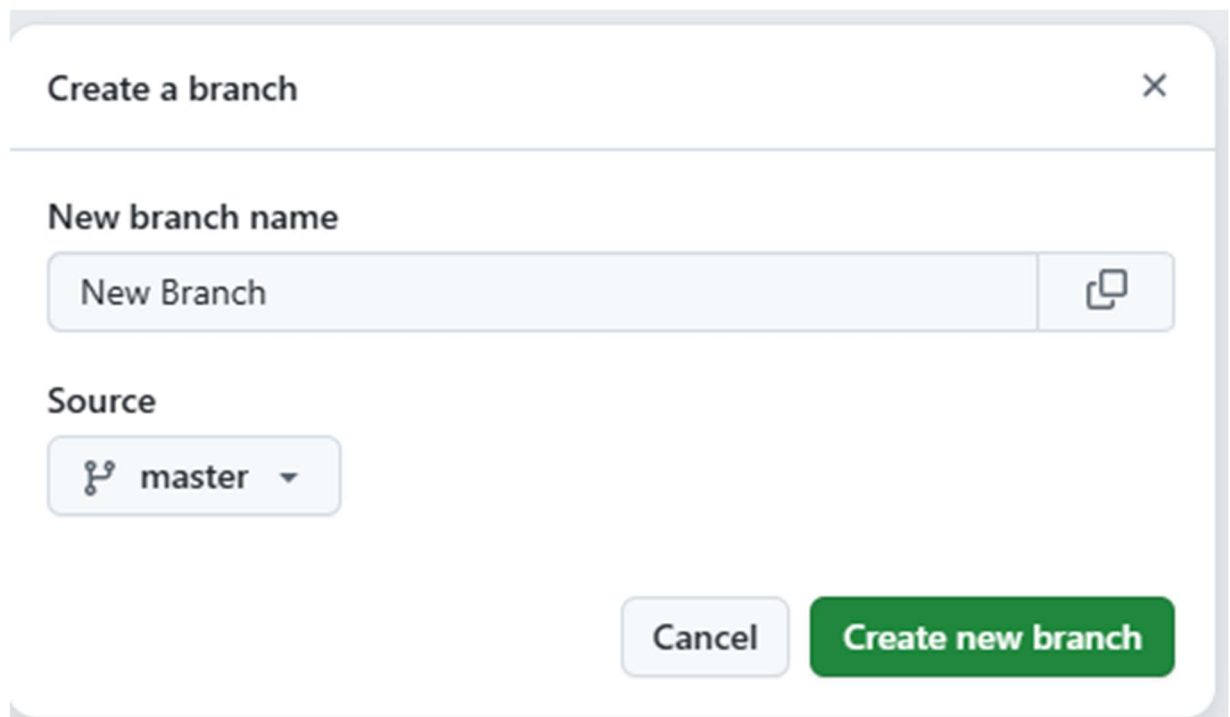


Рисунок 1.15 – Создание «New branch».

И, как результат, в списке веток уже 2 ветки (Рис. 1.16).

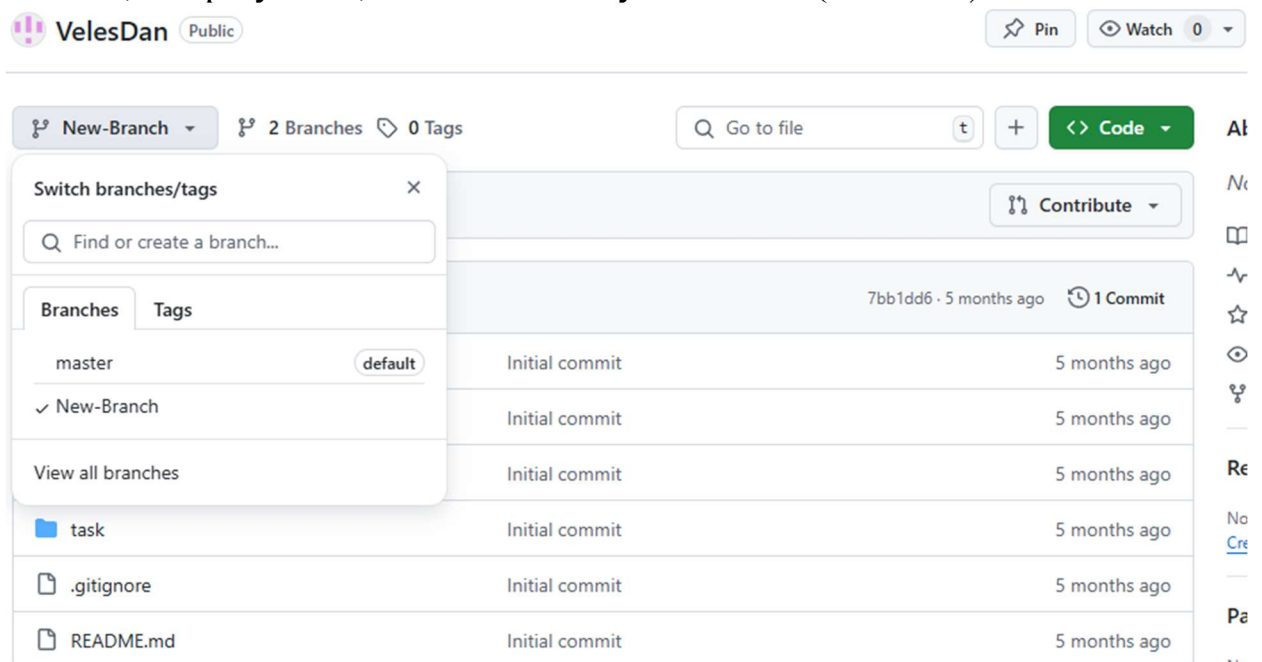


Рисунок 1.16 – Результат

Таким образом, в ходе выполнения данного раздела практики были освоены и продемонстрированы фундаментальные операции для работы с системой контроля версий Git. Были получены практические навыки по созданию и импорту репозитория, внесению изменений через коммиты, использованию веток для изолированной работы и применению Pull Request'ов как основного инструмента для совместной разработки и код-ревью. Эти навыки являются базовыми и необходимыми для любого современного IT-специалиста.

2. НАПИСАНИЕ ДОКУМЕНТОВ В MARKDOWN

2.1. Синтаксис Markdown.

Качественная документация является неотъемлемой частью любого успешного IT-проекта. Она помогает новым участникам команды быстрее войти в курс дела, объясняет, как пользоваться продуктом, и служит центральным источником информации о проекте. Язык разметки Markdown был создан как простой и интуитивно понятный инструмент, позволяющий быстро создавать структурированные и хорошо читаемые документы, которые легко конвертируются в HTML. Благодаря своей простоте и универсальности, он стал стандартом для написания файлов README, ведения проектных вики, описания задач (issues) и документации на таких платформах, как GitHub.

В данном разделе рассматриваются основные элементы синтаксиса Markdown и демонстрируется их практическое применение при создании главного описательного файла проекта.

Markdown — это простой язык разметки для создания форматированного текста.

Ниже перечислим основные элементы синтаксиса.

2.1.1. Заголовки.

Уровень заголовка задается количеством символов # определяет в начале строки(от 1 до 6).

Пример синтаксиса:

Заголовок 1 уровня

Заголовок 2 уровня

Заголовок 3 уровня

Результат:

Заголовок 1 уровня

Заголовок 2 уровня

Заголовок 3 уровня

2.1.2. Форматирование текста

Жирный: Две звездочки ** или два подчеркивания __.

Курсив: Одна звездочка * или одно подчеркивание _.

Зачеркнутый: Две тильды ~~.

Пример синтаксиса:

Это курсив

И это тоже курсив

Это жирный текст

__И это тоже жирный__

А это жирный курсив

~~Это зачеркнутый текст~~

Результат:
Это курсив
И это тоже курсив
Это жирный текст
И это тоже жирный
А это жирный курсив
~~Это зачеркнутый текст~~

2.1.3. Списки

Используйте *, - или + в начале строки.

Пример синтаксиса:

- Первый элемент
- Второй элемент
 - * Вложенный элемент
- * Третий элемент

Результат:

- Первый элемент
- Второй элемент
 - Вложенный элемент
- Третий элемент

Можно использовать цифры с точкой. Порядковый номер не важен, Markdown автоматически расставляет их по порядку.

Пример синтаксиса:

1. Первый шаг
2. Второй шаг
1. Третий шаг (номер исправится автоматически)

Результат:

1. Первый шаг
2. Второй шаг
3. Третий шаг (номер исправится автоматически)

2.1.4. Ссылки

Пример синтаксиса: [Текст ссылки] (URL-адрес)

Это ссылка на [Яндекс](https://yandex.ru).

Результат:

Это ссылка на [Яндекс](https://yandex.ru).

2.1.5. Изображения

Похоже на ссылку, но с восклицательным знаком ! в начале.

Пример синтаксиса: ![Альтернативный текст](URL-адрес изображения)

![Логотип

Google](https://www.google.com/images/branding/googlelogo/2x/googlelogo_col

or_272x92dp.png)

Результат: (Рис. 2.1)



Рисунок 2.1 – Логотип

2.1.6. Цитаты

Используйте знак "больше" > в начале строки.

Пример синтаксиса:

> Это цитата. Каждая новая строка в цитате

> также должна начинаться с символа >.

>

>> Можно делать вложенные цитаты.

Результат:

Это цитата. Каждая новая строка в цитате
также должна начинаться с символа >.

Можно делать вложенные цитаты.

2.1.7. Код

Оберните код в обратные апострофы `.

Пример синтаксиса:

Используйте функцию `print()` для вывода текста.

Результат:

Используйте функцию print() для вывода текста.

2.2. Пример оформления файла в синтаксисе Markdown.

Теоретические знания синтаксиса Markdown были применены для решения ключевой практической задачи — создания файла описания README.md. Этот файл служит "лицом" репозитория и является первым документом, с которым знакомится любой посетитель: будь то член команды, потенциальный работодатель или пользователь.

Поэтому грамотное оформление README.md имеет решающее значение. Использование заголовков, списков, таблиц и форматирования текста позволяет структурировать информацию и сделать ее максимально понятной и доступной. Ниже приведен пример оформления файла README.md для текущего проекта, который объединяет все изученные элементы синтаксиса.

Для выполнения второй задачи изменим файл описания README.md под свой проект в формате Markdown.

Проектная (учебная) практика

Участники

| ФИО | Учебная группа | Код направления подготовки | Профиль
образовательной программы |
| - | - | - | - |
| Данихно В.И. | 241-324 | 09.03.01 | Интеллектуальные беспилотные
системы |

Тема практики

***Проектная практика в информационных технологиях и информационной
безопасности***

Задание

1. Знакомство с местом проведения
2. Настройка Git и репозитория
3. Написание документов в Markdown
4. Создание статического веб-сайта
5. Вариативная часть задания
6. Оформление отчета по практике

Вариативная часть задания

Ответственный по проектной (учебной) практике

Сивцев А.О., кафедра СМАРТ-технологий.

Проектная деятельность

Проектная (учебная) практика проводилась в связке с выполнением
проекта «***Киберполигон***» по дисциплине «Проектная деятельность».

Куратор по проектной деятельности _Георгий Невзоров_.

Листинг 2.2 – файл описания README.md

Результат обработки синтаксиса формата Markdown (Рис. 2.2):

Проектная (учебная) практика

Участники

ФИО	Учебная группа	Код направления подготовки	Профиль образовательной программы
Данихно В.И.	241-324	09.03.01	Интеллектуальные беспилотные системы

Тема практики

Проектная практика в информационных технологиях и информационной безопасности

Задание

1. Знакомство с местом проведения
2. Настройка Git и репозитория
3. Написание документов в Markdown
4. Создание статического веб-сайта
5. Вариативная часть задания
6. Оформление отчета по практике

Вариативная часть задания

Проектирование моделей города и поиск оптимальных вариантов по указанным критериям

Ответственный по проектной (учебной) практике

Сивцев А.О., кафедра СМАРТ-технологий.

Проектная деятельность

Проектная (учебная) практика проводилась в связке с выполнением проекта «*Киберполигон*» по дисциплине «Проектная деятельность».

Куратор по проектной деятельности *Георгий Невзоров*.

Рисунок 2.2 – Результат примера

Таким образом, в рамках данного этапа практики был успешно освоен синтаксис языка разметки Markdown и продемонстрировано умение применять его для создания профессиональной технической документации. Этот навык является фундаментальным для эффективной коммуникации в команде разработчиков и качественной презентации результатов своей работы.

3. СОЗДАНИЕ СТАТИЧЕСКОГО ВЕБ-САЙТА

Процесс разработки статического веб-сайта

Создание сайта для проекта «Киберполигон» велось в несколько последовательных этапов, что позволило создать логичный, структурированный и визуально привлекательный продукт.

3.1. Планирование и структура

На начальном этапе была спроектирована структура сайта, представленная на Рисунке 3.1. Было решено разделить контент на несколько ключевых страниц: главная (index.html), информация о проекте (about.html), команда (team.html), новостной журнал (journal.html) и полезные ресурсы (resources.html). Такое разделение обеспечивает логичную навигацию для пользователя. Все стили были вынесены в отдельный файл style.css, а изображения помещены в папку «images» для поддержания порядка в проекте.

3.2 Структура сайта:

— index.html	(Домашняя страница) (Рис. 3.1)
— about.html	(О проекте) (Рис. 3.2)
— team.html	(Участники) (Рис. 3.3)
— journal.html	(Журнал) (Рис. 3.4)
— resources.html	(Ресурсы) (Рис. 3.5)
— style.css	(Файл стилей для всех страниц)

Все стили были вынесены в отдельный файл style.css (полный код приведен в Приложении А)

└─ images/	
— logo.jpg	(Фоновое изображение для главной) (Рис. 3.6)
— about-logo.jpg	(Изображение для страницы "О проекте") (Рис. 3.7)
└─ journal-post.jpg	(Картинка для постов в журнале) (Рис. 3.8)

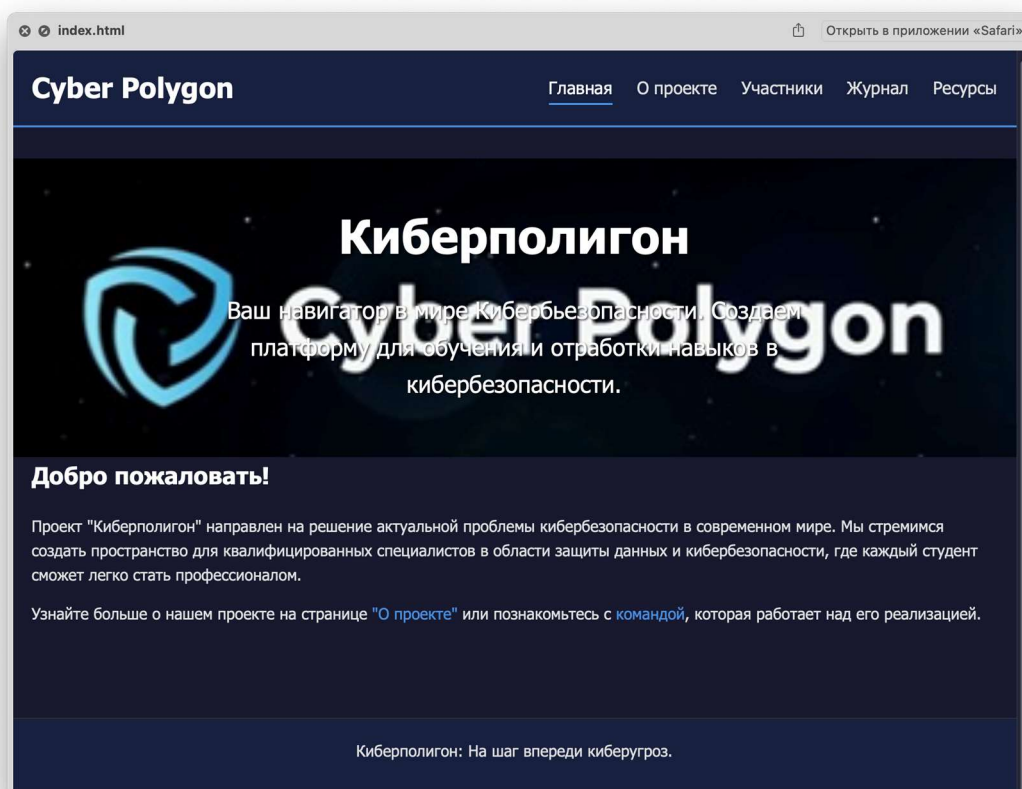


Рисунок 3.1 – index.html

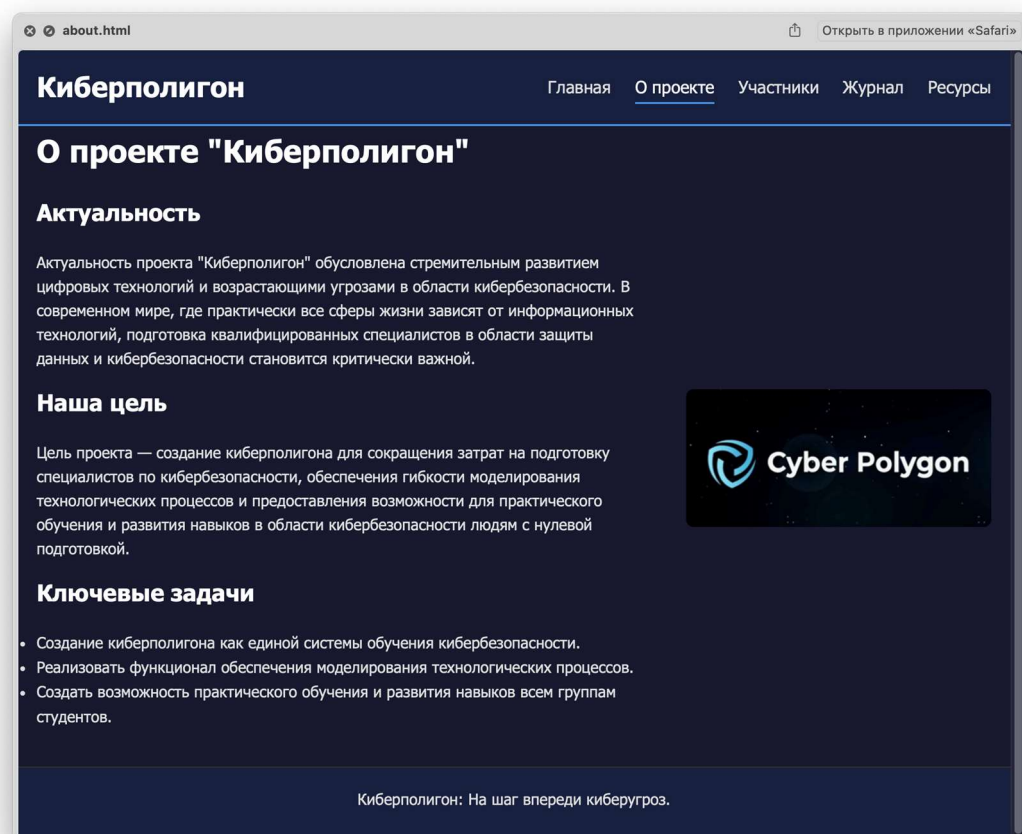


Рисунок 3.2 – about.html

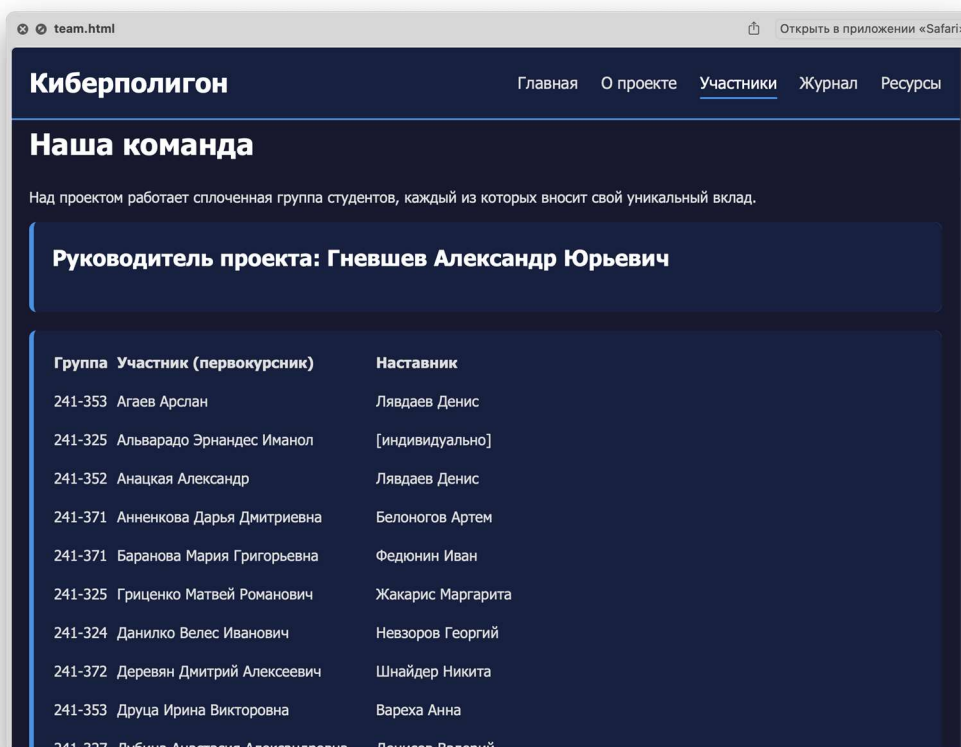


Рисунок 3.3 – team.html

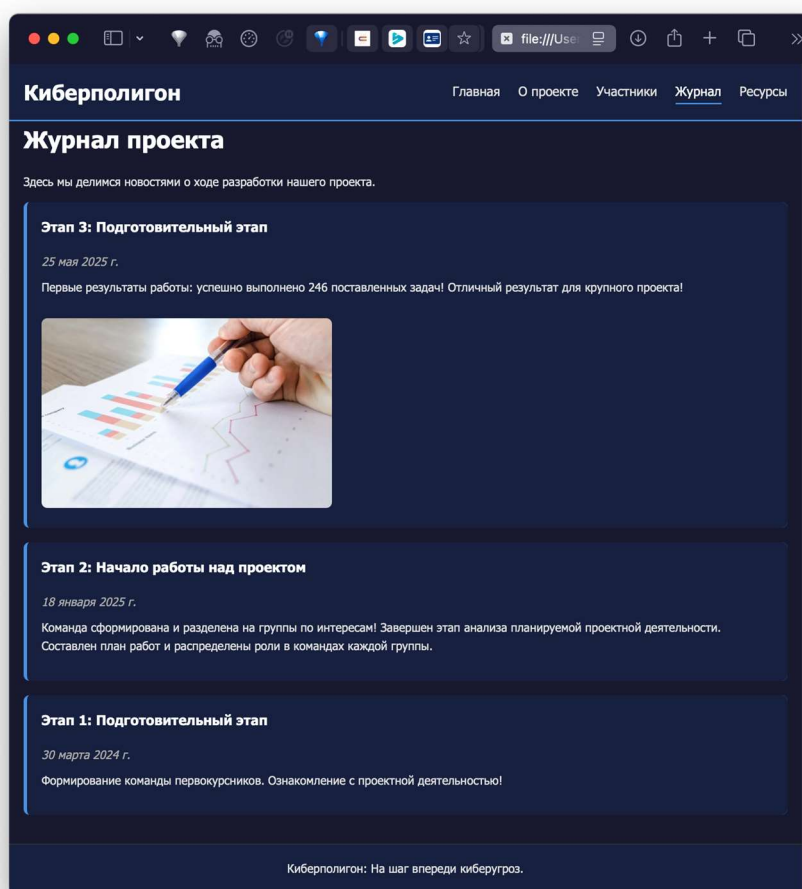


Рисунок 3.4 – journal.html

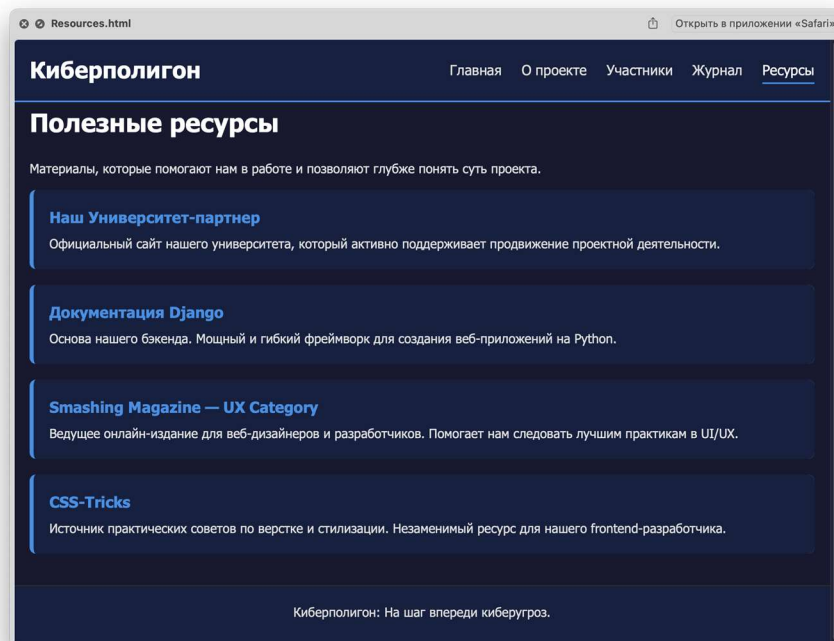


Рисунок 3.5 – Resources.html



Рисунок 3.6 – logo.jpg



Рисунок 3.7 – about-logo.jpg



Рисунок 3.8 – journal-post.jpg

3.3. HTML-верстка.

Каждая страница была сверстана с использованием семантических тегов HTML5 (`<header>`, `<main>`, `<footer>`, `<nav>`, `<section>`), что улучшает доступность сайта и его индексацию поисковыми системами. Основной контент был структурирован с помощью заголовков, параграфов, списков и изображений. Особое внимание было уделено созданию общего для всех страниц header (шапки сайта) с навигацией и footer (подвала).

3.4. Стилизация с помощью CSS.

Для придания сайту современного и профессионального вида был разработан дизайн в темной цветовой гамме с яркими синими акцентами. В файле `style.css` были применены следующие подходы:

- **CSS-переменные:** В блоке «`:root`» были определены основные цвета проекта (`--primary-color`, «`--dark-bg`» и т.д.). Это позволило легко управлять цветовой схемой и обеспечило консистентность дизайна на всех страницах.
- **Flexbox и Grid Layout:** Для позиционирования элементов использовались современные технологии. Flexbox применялся для выравнивания элементов в шапке сайта, а Grid Layout — для создания адаптивной сетки карточек на страницах «Команда» и «Журнал».
- **Интерактивные эффекты:** Были добавлены плавные: «`hover`» эффекты для ссылок и карточек (легкое смещение и тень), что делает интерфейс более отзывчивым и приятным для пользователя.

- **Адаптивность:** Базовые стили обеспечивают корректное отображение на большинстве экранов. Контейнер «container» ограничивает ширину основного контента, предотвращая его растягивание на слишком широких мониторах.

В результате был создан полнофункциональный статический сайт, который наглядно представляет проект «Киберполигон» и демонстрирует полученные навыки в области веб-разработки.

4. ВАРИАТИВНАЯ ЧАСТЬ

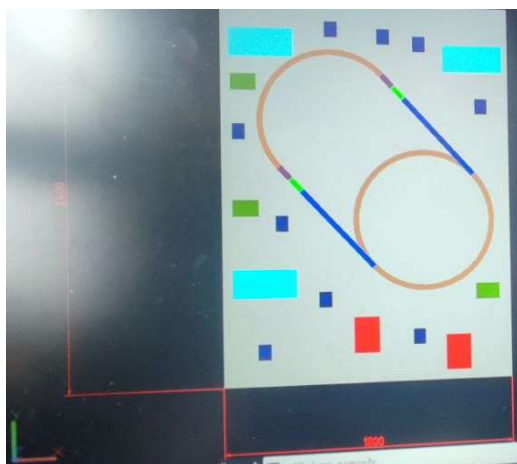
В рамках работы над проектом была поставлена задача по улучшению и детализации макета городского ландшафта в масштабе 1:87. Работа включала в себя несколько ключевых этапов: от концептуального планирования до подбора конкретных элементов и материалов.

4.1 Генерация идей и разработка концепции макета

Первым шагом стала генерация идей по улучшению макета. В соответствии с заданием, была разработана предварительная концепция («предположение») зонирования будущего макета. Для создания логичного и реалистичного городского пространства было предложено разделить его на несколько функциональных зон:

- **Деловой центр:** с высотными офисными зданиями и бизнес-центрами.
- **Жилой квартал:** включающий многоэтажные дома.
- **Промышленная зона:** представленная моделями железных дорог, заводов или фабрик.
- **Общественные пространства:** например, административные здания, такие как полицейский участок.

Результатом этого этапа стал эскизный план-схема, определяющий примерное расположение ключевых объектов, дорожной сети и других элементов для создания визуально привлекательной и гармоничной композиции (Рис. 4.1).



4.2. Поиск и анализ моделей зданий (масштаб 1:87)

Следующим этапом стал целенаправленный поиск подходящих моделей зданий. Был проведен анализ рынка моделей для железнодорожных макетов и диорам по категориям, соответствующим разработанной концепции.

- **Описание размеров:**

- Для каждой найденной потенциальной модели были зафиксированы точные габаритные размеры (длина, ширина, высота). Этот шаг был критически важен для проверки того, как объекты впишутся в план макета, и для соблюдения правильных пропорций между зданиями.

- **Поиск оптимальных вариантов:**

- На основе собранных данных (внешний вид, размеры, стилистика) был проведен сравнительный анализ. Были отобраны наиболее оптимальные варианты, которые не только соответствуют масштабу 1:87, но и гармонично сочетаются друг с другом, создавая единый архитектурный ансамбль. В шорт-лист вошли модели как высотных офисных зданий, так и промышленных объектов и зданий служб (Рис. 4.2).



Рисунок 4.2 – Макет здания

4.3. Подбор материалов для основания макета

Параллельно с поиском зданий велась работа по подбору материала для основания, на котором будут размещаться все элементы. В соответствии с задачей найти практичное и недорогое решение, были исследованы следующие варианты:

- **Технический тент:** Прочный, износостойкий материал, который может служить черновой основой (Рис. 4.3).



Рисунок 4.3 – Технический тент

- **Однотонная ПВХ-плёнка:** Гладкая плёнка (серого, зеленого или черного цвета) может имитировать асфальт или служить нейтральным фоном для дальнейшего нанесения ландшафта (травы, разметки)

Был сделан вывод, что оптимальным решением на начальном этапе будет использование плотной однотонной плёнки белого/серого цвета, так как она обеспечивает ровную поверхность и является идеальной базой для последующего декорирования.



Рисунок 4.4 – ПВХ-плёнка

Итог работы:

В результате проделанной работы была сформирована целостная концепция обновленного макета, подготовлен эскизный план его зонирования, подобраны оптимальные модели зданий с описанием их характеристик и найдены подходящие материалы для создания основы. Данный этап является прочным фундаментом для дальнейшей практической реализации проекта.

ЗАКЛЮЧЕНИЕ

В ходе прохождения учебной (проектной) практики были успешно решены все поставленные задачи, что позволило закрепить теоретические знания и приобрести ценные практические навыки в области информационных технологий.

В рамках работы были освоены ключевые инструменты современного IT-специалиста:

1. **Система контроля версий Git:** Были изучены и применены на практике основные операции, такие как клонирование репозитория, создание веток, фиксация изменений (коммиты) и объединение веток через pull-запросы. Это заложило основу для понимания принципов командной разработки.
2. **Язык разметки Markdown:** Был освоен синтаксис Markdown для создания структурированной и легко читаемой технической документации, что было продемонстрировано при оформлении файла README.md.
3. **Основы веб-разработки:** Были получены практические навыки верстки статических веб-сайтов с использованием HTML5 и CSS3. В ходе создания сайта проекта «Киберполигон» были применены семантическая верстка, современные методы компоновки элементов (Flexbox, Grid) и стилизация с помощью CSS-переменных.
4. **Проектно-исследовательская деятельность:** В рамках вариативной части была проведена работа по концептуальному проектированию макета, включающая анализ рынка, подбор моделей и материалов по заданным критериям.

Проектная практика позволила получить целостное представление о жизненном цикле небольшого проекта: от постановки задачи и планирования до реализации, документирования и представления результатов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Чейкон, С. Профессиональный Git / С. Чейкон, Б. Штрауб. – М.: ДМК Пресс, 2021. – 492 с.
2. MDN Web Docs [Электронный ресурс] // Mozilla. – URL: <https://developer.mozilla.org/ru/docs/Web> (дата обращения: 20.05.2024).
3. Gruber, J. Markdown: Syntax [Электронный ресурс] // Daring Fireball. – URL: <https://daringfireball.net/projects/markdown/syntax> (дата обращения: 15.05.2024).
4. Флэнаган, Д. JavaScript. Подробное руководство. 7-е издание. – СПб.: Питер, 2021. – 720 с.
5. CSS-Tricks [Электронный ресурс] // A comprehensive guide to CSS Grid. – URL: <https://css-tricks.com/snippets/css/complete-guide-grid/> (дата обращения: 18.05.2024).
6. Auhagen. Каталог моделей и аксессуаров для макетов [Электронный ресурс]. – URL: <https://www.auhagen.de/en/> (дата обращения: 22.05.2024).
7. GitHub Docs [Электронный ресурс] // GitHub. – URL: <https://docs.github.com/> (дата обращения: 14.05.2024).

ПРИЛОЖЕНИЕ А. Листинг файла Index.html

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Киберполигон – Главная</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    <div class="container">
      <a href="index.html" class="logo">Cyber Polygon</a>
      <nav>
        <ul>
          <li><a href="index.html" class="active">Главная</a></li>
          <li><a href="about.html">О проекте</a></li>
          <li><a href="team.html">Участники</a></li>
          <li><a href="journal.html">Журнал</a></li>
          <li><a href="resources.html">Ресурсы</a></li>
        </ul>
      </nav>
    </div>
  </header>

  <main>
    <section class="hero">
      <h1>Киберполигон</h1>
      <p>Ваш навигатор в мире Кибербезопасности. Создаем
платформу для обучения и отработки навыков в
кибербезопасности.</p>
    </section>
```

```
<section class="container">
  <h2>Добро пожаловать!</h2>
  <p>Проект "Киберполигон" направлен на решение актуальной
проблемы кибербезопасности в современном мире. Мы стремимся
создать пространство для квалифицированных специалистов в
области защиты данных и кибербезопасности, где каждый студент
сможет легко стать профессионалом.</p>
  <p>Узнайте больше о нашем проекте на странице <a
href="about.html">"О проекте"</a> или познакомьтесь с <a
href="team.html">командой</a>, которая работает над его
реализацией.</p>
</section>
</main>
<footer>
  <div class="container">
    <p>Киберполигон: На шаг впереди киберугроз.</p>
  </div>
</footer>
</body>
</html>
```


ПРИЛОЖЕНИЕ Б. Листинг файла style.css

```
/* --- Глобальные стили и переменные --- */
:root {
  --primary-color: #4a90e2; /* Яркий синий акцент */
  --dark-bg: #1a1a2e;      /* Темно-синий фон */
  --light-bg: #16213e;     /* Чуть светлее фон для карточек */
  --text-color: #e0e0e0;   /* Основной цвет текста */
  --header-color: #0f3460; /* Цвет заголовков */
}
/* --- Сброс стилей и базовые настройки --- */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;}
body {
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
  background-color: var(--dark-bg);
  color: var(--text-color);
  line-height: 1.6;}
.container {
  max-width: 1100px;
  margin: auto;
  padding: 0 20px;}
h1, h2, h3 {
  color: var(--header-color);
  margin-bottom: 20px;
  color: #fff;}
p {
  margin-bottom: 15px;}
a {
  color: var(--primary-color);
  text-decoration: none;}

a:hover {
  text-decoration: underline;}
/* --- Шапка сайта (Header) --- */
header {
  background: var(--light-bg);
```

```

padding: 1rem 0;
border-bottom: 2px solid var(--primary-color);
position: sticky;
top: 0;
z-index: 100;}
header .container {
display: flex;
justify-content: space-between;
align-items: center;}
.logo {
font-size: 1.8rem;
font-weight: bold;
color: #fff;}
nav ul {
list-style: none;
display: flex;}
nav ul li {
margin-left: 25px;}
nav ul li a {
color: var(--text-color);
font-size: 1.1rem;
padding-bottom: 5px;}
nav ul li a:hover,
nav ul li a.active {
color: #fff;
border-bottom: 2px solid var(--primary-color);}
/* --- Основной контент (Main) --- */
main {
padding: 40px 0;}

/* --- Стили для главной страницы (Hero) --- */
.hero {
background: url('images/logo.jpg') no-repeat center center/cover;
height: 40vh;
display: flex;
flex-direction: column;
justify-content: center;
align-items: center;
text-align: center;

```

```

    color: #fff;
    padding: 0 20px;}
.hero h1 {
    font-size: 3rem;
    text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.7);}
.hero p {
    font-size: 1.5rem;
    max-width: 700px;
    text-shadow: 1px 1px 2px rgba(0, 0, 0, 0.7);}
/* --- Стили для карточек (Участники, Журнал) --- */
.card-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
    gap: 30px;}
.card {
    background: var(--light-bg);
    border-radius: 8px;
    padding: 25px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
    transition: transform 0.3s ease, box-shadow 0.3s ease;}
.card:hover {
    transform: translateY(-5px);
    box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);}
.card img {
    width: 100px;
    height: 100px;
    border-radius: 50%;
    margin-bottom: 15px;
    object-fit: cover;
    border: 3px solid var(--primary-color);}
.card h3 {
    margin-bottom: 10px;}
.card .role {
    font-weight: bold;
    color: var(--primary-color);
    margin-bottom: 10px;}
/* --- Стили для страницы "О проекте" --- */
.about-content {
    display: flex;

```

```

    gap: 40px;
    align-items: center;}
.about-content .text {
    flex: 2;}
.about-content .image {
    flex: 1;}
.about-content img {
    width: 100%;
    border-radius: 8px;}
/* --- Стили для Журнала и Ресурсов --- */
.post, .resource-list li {
    background: var(--light-bg);
    padding: 20px;
    margin-bottom: 20px;
    border-radius: 8px;
    border-left: 5px solid var(--primary-color);}
.post-date {
    font-style: italic;
    color: #aaa;
    margin-bottom: 10px;
    display: block;}
.resource-list {
    list-style: none;}
.resource-list a {
    font-size: 1.2rem;
    font-weight: bold;}
.resource-list p {
    margin-top: 5px;
    margin-bottom: 0;}

/* --- Нижняя часть сайта (Footer) --- */
footer {
    background: var(--light-bg);
    text-align: center;
    padding: 20px 0;
    margin-top: 40px;
    border-top: 1px solid #333;
}

```