

Федеральное государственное автономное образовательное учреждение высшего
образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра «Инфокогнитивные технологии»

Направление подготовки/ специальность: Разработка и интеграция бизнес-приложений

ОТЧЕТ

по проектной практике

Студент: Обеднин Ярослав Андреевич; Группа: 241-362

Студент: Яковлев Артур Олегович; Группа: 241-362

Место прохождения практики: Московский Политех, кафедра
«Инфокогнитивные технологии»

Отчет принят с оценкой _____ Дата _____

Руководитель практики: Кулибаба Ирина Викторовна

Москва 2025

ОГЛАВЛЕНИЕ

Оглавление

1. ВВЕДЕНИЕ	3
2. ОРГАНИЗАЦИЯ	4
3. ЗАДАНИЯ	5
3.1 Описание заданий	6
3.2 Описание достигнутых результатов по проектной практике	8
4. ИНДИВИДУАЛЬНЫЕ ПЛАНЫ УЧАСТНИКОВ.....	11
4.4 Обеднин Ярослав Андреевич.....	11
4.5 Яковлев Артур Олегович	12
5. САЙТ.....	13
6. ЗАКЛЮЧЕНИЕ	14
7. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	15

1. ВВЕДЕНИЕ

В условиях стремительной цифровизации образовательных процессов возрастает необходимость модернизации инструментов, обеспечивающих эффективное взаимодействие между студентами, кураторами проектов и университетскими структурами. В рамках проекта «Совершенствование цифровой инфраструктуры для проектной деятельности» была выдвинута идея создания интегрированной системы, которая решит ключевые проблемы организации проектной деятельности в Московском Политехе.

Актуальность проекта обусловлена ростом числа студентов, ежегодно поступающих в университет, и необходимостью сопоставления их навыков с требованиями действующих проектов. Однако текущая цифровая инфраструктура столкнулась с рядом ограничений: функция выбора проекта и подачи заявок не интегрирована в обновлённый интерфейс, а кураторы проектов лишены современного личного кабинета, что снижает оперативность и удобство работы с системой. Эти пробелы приводят к трудностям в сопоставлении потребностей сторон, увеличению нагрузки на административный аппарат и снижению вовлечённости студентов в проектную деятельность.

Цель проекта — модернизировать цифровую инфраструктуру проектной деятельности университета, расширив функциональные возможности сервисов и обеспечив их соответствие современным требованиям. Для достижения этой цели были поставлены задачи: анализ текущего состояния системы, разработка решений для интеграции новых функций, проектирование интерфейсов, включая личный кабинет кураторов, а также регулярное взаимодействие с заказчиком для согласования доработок.

Ключевым этапом реализации стало взаимодействие с сотрудниками университета, позволившее выявить приоритетные направления улучшений. Так, отсутствие обновлённого интерфейса кураторов проектов было признано одной из главных проблем, что легло в основу разработки нового дизайна личного кабинета. Апробация проекта проводилась в форме еженедельной отчетности перед

заказчиком — Центром проектной деятельности Московского Политеха, что обеспечило прозрачность процесса и оперативную обратную связь.

Проект направлен на создание единой цифровой экосистемы, где студенты смогут легко находить проекты, соответствующие их интересам и компетенциям, а кураторы — управлять командами и мониторить прогресс. Решение этих задач станет важным шагом в повышении эффективности проектной деятельности и укреплении позиций Московского Политеха как инновационного образовательного центра.

2. ОРГАНИЗАЦИЯ

Заказчиком проекта выступает Факультет информационных технологий Московского Политехнического университета — современное образовательное подразделение, ориентированное на подготовку высококвалифицированных специалистов в области IT и цифровых технологий. Факультет активно внедряет инновационные методы обучения и разрабатывает цифровую образовательную среду, направленную на повышение эффективности учебного процесса, включая проектную деятельность.

Структура факультета включает профильные кафедры (программная инженерия, информационная безопасность, обработка данных и другие), исследовательские лаборатории, проектные команды и административные службы. Каждое из этих звеньев отвечает за определённые направления деятельности: от разработки и реализации образовательных программ до участия в научных и технологических инициативах. Особое внимание уделяется практической подготовке студентов, включая проектную деятельность, которая позволяет формировать навыки, востребованные в реальной IT-индустрии.

Ключевыми направлениями деятельности факультета являются:

Образование на уровнях бакалавриата, магистратуры и аспирантуры по специализациям, связанным с цифровыми технологиями.

Развитие исследовательских и инновационных проектов, включая создание цифровых сервисов для образовательных и административных нужд университета.

Внедрение современных инструментов и платформ для оптимизации учебного процесса, в том числе автоматизации рутинных задач.

Проект «Совершенствование цифровой инфраструктуры для проектной деятельности» стал логичным шагом в рамках общей стратегии цифровизации, проводимой факультетом. Его реализация направлена на решение актуальных проблем студентов и преподавателей, связанных с организацией проектной деятельности. В условиях динамичного изменения потребностей сторон (студентов, кураторов проектов и университетских структур) существующая цифровая инфраструктура оказалась недостаточно гибкой. Проблемы включают:

Отсутствие интеграции функции выбора проекта и подачи заявок в обновлённый интерфейс.

Устаревший дизайн личных кабинетов кураторов проектов, что снижает удобство взаимодействия.

Недостаточная прозрачность и эффективность процессов сопоставления навыков студентов с требованиями проектов.

3. ЗАДАНИЯ

Проектная практика студентов первого курса, обучающихся по направлениям, связанным с информационными технологиями и кибербезопасностью, — это ключевой этап учебного процесса, направленный на развитие практических навыков, умение работать с инструментами версионного контроля, создавать цифровые продукты и взаимодействовать с профессиональной средой. Практика рассчитана на 72 академических часа и включает как базовые задачи, необходимые всем участникам, так и вариативные задания, позволяющие адаптировать обучение под индивидуальные цели и уровень подготовки.

Задание построено на модульной структуре: обязательная часть охватывает настройку репозитория, работу с Markdown, создание статических сайтов и взаимодействие с партнёрами, а вариативная часть открывает возможности для реализации индивидуальных проектов или глубокого изучения технологий. Это

обеспечивает баланс между стандартами обучения и свободой творчества, формируя у студентов навыки, востребованные в реальной IT-индустрии.

3.1 Описание заданий

1. Настройка Git-репозитория:

- Создать групповой или личный репозиторий на GitHub или GitVerse на основе предоставленного шаблона.
- Освоить базовые команды Git: клонирование, коммит, пуш, создание веток.
- Регулярно фиксировать изменения с осмысленными сообщениями к коммитам.

2. Написание документов в Markdown:

- Оформить все материалы проекта (описание, журнал прогресса и др.) в формате Markdown.
- Изучить синтаксис Markdown и подготовить необходимые документы.

3. Создание статического веб-сайта:

Сайт можно создать с помощью HTML и CSS (базовый уровень) или генератора статических сайтов, такого как Hugo (рекомендуется).

Сайт должен включать:

- Домашнюю страницу с аннотацией проекта.
- Страницу «О проекте» с описанием.
- Страницу «Участники» с вкладом каждого студента.
- Страницу «Журнал» с 3 постами о прогрессе.
- Страницу «Ресурсы» со ссылками на партнёров и материалы.

Дизайн и контент должны быть уникальными (минимум 50% оригинальности).

Добавить графические материалы (фото, схемы, диаграммы, видео).

4. Взаимодействие с организацией-партнёром:

- Участвовать в профильных мероприятиях.
- Подготовить отчёт в Markdown: описание опыта, знаний и связи с проектом.

- Добавить отчёт в репозиторий.

Стажировки и взаимодействие учитываются при оценке.

5. Вариативная часть задания:

5.1 Кафедральное индивидуальное задание:

- Выполнить задачи базовой части.
- Реализовать индивидуальное задание по теме, согласованной с ответственным за практику.
- Интегрировать результаты в репозиторий и сайт.

5.2 Практическая реализация технологии:

- Выбрать технологию из списка (или альтернативного источника, согласованного с ответственным).
- Согласовать тему и стек технологий внутри команды.
- Провести исследование: изучить реализацию и воспроизвести её.
- Создать техническое руководство в Markdown с пошаговыми инструкциями.
- Примерами кода.
- Диаграммами, схемами (3–10 штук).
- Модифицировать проект (творческий этап).
- Создать видеопрезентацию (цель, задачи, демонстрация).
- Задokumentировать проект в репозитории и разместить его на сайте.
- Подготовить финальный отчёт с хронологией этапов.

6. Итоговый отчёт:

Составить отчёт по шаблону в папке reports (файлы: Отчёт.docx, report.docx).

Сформировать PDF-версию отчёта.

Разместить оба файла в репозитории и загрузить их в СДО (LMS).

Описать этапы работы в хронологическом порядке.

Включить индивидуальные планы участников.

Форматирование и инструменты:

Управление версиями: Git.

Документация: Markdown.

Статический сайт: HTML/CSS или Hugo.

Платформы: GitHub/GitVerse.

Графика: диаграммы, схемы, иллюстрации.

3.2 Описание достигнутых результатов по проектной практике

Первым шагом в реализации проекта стало создание структуры для командной разработки. Мы организовали групповой репозиторий на GitHub, что позволило обеспечить прозрачность процесса и синхронизировать работу участников. Каждое изменение сопровождалось комментариями к коммитам, что облегчило отслеживание прогресса и упростило дальнейшую интеграцию кода.

Для повышения читаемости текстовых материалов проекта мы выбрали формат Markdown. Он оказался удобным как для написания технической документации, так и для оформления пользовательских сообщений и описаний разделов. Весь контент — от описания функционала бота до отчётов по этапам работы — был оформлен с использованием этого стандарта.

В рамках проекта была разработана подробная документация, охватывающая ключевые аспекты реализации. Для её оформления был выбран формат Markdown, который обеспечил структурированность, читаемость и универсальность текстовых материалов. Этот формат стал основным для внутренней документации, включая:

Описание функционала бота : детализация команд, логика обработки запросов, интерактивные элементы (кнопки, меню).

Технические аспекты: выбор стека технологий (Node.js, Telegraf, Markdown для контента), архитектура бота (модульная организация кода, обработка callback-запросов, навигация между разделами).

Инструкции по установке и запуску: пошаговое руководство для настройки среды (установка Node.js, настройка переменных окружения через .env, запуск бота).

Отчёты по взаимодействию с партнёрами: описание встреч, обратной связи от заказчика, согласование требований к функционалу и дизайну.

Документация также включала примеры кода (например, реализацию `navigationHandler.js`), схемы логики бота и ссылки на внешние ресурсы (прототипы интерфейсов в Figma, официальные страницы университета).

В рамках проекта было разработано статическое веб-приложение с использованием Hugo, генератора статических сайтов.

Сайт состоит из следующих страниц:

Главная (`index.md`) — главная страница с кратким описанием проекта.

О проекте (`about.md`) — страница «О проекте» включающая: цель, актуальность, задачи, проблематика.

Участники (`participants.md`) — информация о вкладе каждого студента.

Журнал (`journal.md`) — хронология ключевых событий и решений.

Ресурсы (`resources.md`) — ссылки на используемые инструменты, прототипы и материалы.

Технологии и особенности

- Markdown использовался для написания контента, что обеспечило единообразие с документацией бота.
- HTML/CSS применялись для оформления дизайна, с акцентом на читаемость и адаптивность.
- Hugo автоматизировал сборку сайта, позволяя сосредоточиться на содержании.
- Гибкая структура позволяла легко добавлять новые разделы и обновлять существующие.

Сайт стал важным элементом презентации проекта, позволяющим визуализировать результаты работы и демонстрировать их широкой аудитории. Его разработка дала возможность глубже погрузиться в тематику проекта, структурировать информацию и представить её в виде легко воспринимаемого и доступного контента. Использование Hugo позволило автоматизировать часть

процесса сборки сайта, сохранив при этом высокую степень контроля над дизайном и содержанием.

После завершения сайта была реализована его «интерактивная версия» — Telegram-бот, написанный на платформе Node.js с использованием библиотеки Telegraf. Бот обеспечивал динамическое взаимодействие с пользователями, позволяя получать информацию о проекте в удобной форме.

Основные функции бота включали:

- Отображение главного меню.
- Навигацию по разделам: "О проекте", "Участники", "Журнал прогресса", "Ресурсы".
- Вывод информации о цели, актуальности, задачах и проблематике проекта.
- Представление информации о вкладе каждого участника.
- Описание этапов работы и диаграммы Ганта.
- Ссылки на внешние ресурсы и прототипы.
- Интерактивные элементы : случайные интересные факты, кнопки навигации, переходы в подменю.

Также была реализована логика обработки нажатий на кнопки и переходов между разделами, что сделало интерфейс бота удобным и интуитивно понятным для пользователя.

Реализация сопровождалась настройкой среды (установка Node.js, настройка .env-файла, использование библиотеки telegraf/Markup для создания интерфейса). Все этапы были задокументированы в формате Markdown, а финальный отчёт включал описание вклада каждого участника и технических решений.

Финальный отчёт, составленный на основе шаблона, стал завершающим этапом реализации проекта и стал ключевым документом, отражающим весь цикл работ, вклад участников и технические решения.

Отчёт был разработан в формате Word (DOCX) и сопровождался PDF-версией, что обеспечило универсальность и удобство для дальнейшего хранения и использования.

4. ИНДИВИДУАЛЬНЫЕ ПЛАНЫ УЧАСТНИКОВ

4.4 Обеднин Ярослав Андреевич

<i>Задача</i>	<i>Время, ч</i>
Создание группового репозитория. Заполнение репозитория по шаблону.	2
Освоение Git.	4
Изучение синтаксиса Markdown	4
Изучение HUGO	5
Взаимодействие с организацией-партнером «Робостанция»	5
Создание видеопрезентации проекта	3.5
Написание документации сайта в формате Markdown	7
Написание руководства по созданию проекта в формате Markdown	7
Настройка и изучение статического веб-сайта	9
Заполнение сайта	8
Проведение исследования технологии	5
Заполнение бота, его разделов и функций	5
Написание финального отчёта	9

Итого данный студент затратил 73,5 часов на вклад в проект.

4.5 Яковлев Артур Олегович

<i>Задача</i>	<i>Время, ч</i>
Настройка репозитория	2.5
Изучение Git и настройка системы ветвления	4
Оформление материалов сайта в формате Markdown	5
Разработка и оформление страницы «Участники» и «Ресурсы»	7
Интеграция логики взаимодействия в Telegram-бот (callback, меню)	6
Исследование технологии Telegram-ботов и библиотеки Telegraf	5
Реализация интерактивных функций бота (факты, переходы)	5
Проведение тестирования Telegram бота	4
Взаимодействие с организацией-партнёром, подготовка отчёта	5
Подготовка видеопрезентации	3.5
Проведение исследования сторонней технологии	6
Создание технического руководства по развертыванию проекта	7
Финальное редактирование сайта и устранение багов	6
Написание финального отчёта и оформление PDF-версии	7

Итого данный студент затратил 73 часов на вклад в проект.

5. САЙТ

Сайт "Совершенствование цифровой инфраструктуры для проектной деятельности" — это многостраничный веб-ресурс, созданный с использованием генератора статических сайтов Hugo . Он служит платформой для представления идей по модернизации цифровой среды, поддерживающей проектную деятельность в Московском политехническом университете. На сайте размещена информация о ключевых направлениях развития цифровой инфраструктуры, примерах успешных решений, а также предложениях по внедрению новых инструментов и технологий.

Дизайн сайта выполнен в минималистичном стиле с использованием сдержанных цветов, что обеспечивает высокую читаемость и эстетическую гармонию. Интерфейс ориентирован на удобство пользователя: чёткая структура, логическое расположение элементов и современные шрифты позволяют легко находить нужную информацию.

Страницы:

Главная

Предназначена для краткого представления проекта. Здесь описывается его суть, цель и перечисляются основные разделы. Страница служит стартовой точкой для навигации по сайту.

О проекте

Содержит подробную информацию о целях, задачах и реализации проекта. Описываются ключевые функции, преимущества, а также возможные направления развития. Эта страница помогает лучше понять концепцию и значение проекта.

Участники

Представляет команду, участвовавшую в разработке. Здесь можно найти информацию о ролях каждого участника, их вкладе в проект и используемых технологиях.

Журнал

Отражает историю развития проекта. Публикуются заметки о важных этапах реализации, ключевых решениях и достигнутых результатах. Каждый блок

содержит описание этапа с возможностью раскрытия дополнительной информации.

Ресурсы

Содержит ссылки на полезные материалы, прототипы, внешние источники и партнёров проекта. Все ссылки снабжены анимацией: при наведении мыши элементы страницы реагируют, добавляя интерактивности.

6. ЗАКЛЮЧЕНИЕ

В ходе выполнения проекта наша команда реализовала комплексные решения по модернизации цифровой инфраструктуры проектной деятельности Московского Политеха. Были разработаны два ключевых компонента: Telegram-бот на Node.js и статический сайт на Hugo, которые вместе обеспечивают удобный доступ к информации о проектах, их участникам и ресурсам. На практике были освоены технологии, такие как версионный контроль через Git, создание интерактивных Telegram-интерфейсов с использованием библиотеки Telegraf, а также генерация статических сайтов с применением Markdown и HTML/CSS.

Telegram-бот предоставляет пользователям структурированную информацию о проекте, включая цели, актуальность, участников, этапы работы и ссылки на внешние ресурсы. Его функционал включает навигацию по разделам, отображение случайных интересных фактов и интеграцию с внешними ссылками.

Сайт, созданный с использованием генератора статических страниц, стал визуальной и информационной основой проекта. Он включает пять страниц: «Главная», «О проекте», «Участники», «Журнал прогресса», «Ресурсы».

Все участники проекта внесли вклад в код, структуру, интерфейс и документацию, благодаря чему результат является не только работоспособным, но и понятным для сторонних пользователей и разработчиков

Ценность проекта для заказчика заключается в следующем:

- Повышение эффективности взаимодействия между студентами, кураторами и проектными командами благодаря структурированному представлению информации.
- Упрощение процесса выбора проектов за счёт интеграции механизмов подбора студентов по навыкам и актуализации интерфейсов личных кабинетов.
- Демонстрация технической реализуемости решения, которое может быть масштабировано и интегрировано в существующие системы университета.
- Создание инструмента для презентации проектной деятельности, который может использоваться как образцовый материал для дальнейших инициатив.

7. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. GitHub Docs. Работа с репозиториями, ветками и Pull Request [Электронный ресурс]. – URL: <https://docs.github.com/ru>
2. Что такое Git: объяснение на схемах [Электронный ресурс]. – URL: https://skillbox.ru/media/code/что_такое_git_obyasnyаем_na_skhemakh/
3. Введение в Git. Бесплатный онлайн-курс [Электронный ресурс]. – URL: https://ru.hexlet.io/courses/intro_to_git
4. Руководство по синтаксису Markdown [Электронный ресурс]. – URL: https://ru.hexlet.io/lesson_filters/markdown
5. Генерация статических сайтов с Hugo – URL: <https://habr.com/ru/articles/700640/>