Федеральное государственное автономное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий  
Кафедра «Смарт-технологии»

Направление подготовки/ специальность: 09.03.01 «Информатика и вычислительная техника»

ОТЧЕТ

по учебной (проектной) практике

Студент: Феклин Роман Евгеньевич Группа: 241-325

Место прохождения практики: Московский Политехнический университет, кафедра«СМАРТ-технологии»

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

Руководитель практики: Гневшев Александр Юрьевич

Москва 2025

**Оглавление**

**Введение………………………………………………………………………....3**

1. **Общая информация о проекте…………………..……………...….….4**
2. **Общая характеристика деятельности** **организации..………………5**
3. **Описание задания по проектной практике..………………………...6**
4. **Описание достигнутых результатов по проектной практике…….9**
   1. Настройка Git и организация командной разработки…………...……..9
   2. Работа с документацией в формате MarkDown………….……………10
   3. Разработка статического веб-сайта…………………………….………11
      1. Главная страница…………………………………..……………………11
      2. Страница «О проекте»…………………………………………………..12
      3. Страница «Команда»……………………………………………………12
      4. Страница «Журнал»……………………………………………….…….12
      5. Страница «Ресурсы»……………………………………………….……12
   4. Взаимодействие с организацией-партнёром…………………………..13
   5. Тестирование и отладка веб-сайта……………………………………..13
   6. Тестирование и отладка телеграмм-бота……………………...……….15

**Заключение…………………………………………………………………….18**

**Список использованной литературы……………………...……………….20**

**Приложение……………………………………………………………………21**

**Введение**

В период с 3 февраля по 24 мая 2025 года я проходил учебную практику на кафедре «СМАРТ-технологии» Московского Политехнического Университета, участвуя в проекте, направленном на автоматизацию и цифровизацию учебного процесса. Основной целью практики было закрепление теоретических знаний и развитие практических навыков, необходимых для самостоятельной работы над ИТ-проектами.

В ходе практики я занимался разработкой двух ключевых решений:

* Веб-сайта с информацией о проекте, реализованного с использованием современных технологий веб-разработки.
* Telegram-бота, который позволяет удобно просматривать и фильтровать расписание сессии, получать напоминания о предстоящих экзаменах и быстро переходить к нужной информации.

В процессе работы были изучены и применены такие инструменты и технологии, как Python (для серверной логики бота), библиотека python-telegram-bot (для интеграции с Telegram), SQLite (для хранения данных), HTML, CSS, JS (для создания и стилизации веб-страниц), а также системы контроля версий Git и GitHub для совместной работы и хранения кода. Особое внимание уделялось вопросам структурирования кода, организации базы данных, созданию удобного пользовательского интерфейса и оформлению проектной документации с помощью разметки Markdown.

Данный отчет содержит подробное описание реализованных проектов, этапов их разработки, а также информации об организации, в которой проходила практика. В результате прохождения практики я приобрёл ценный опыт в области веб-разработки, создания прикладных решений и командной работы, что будет полезно для дальнейшего профессионального развития.

**1. Общая информация о проекте**

Название проекта: Киберполигон.

Цели данного проекта:

1. Создание и развитие киберполигона на базе университета как образовательной, исследовательской и практической платформы;
2. Обучение студентов навыкам кибербезопасности и повышение их конкурентно способности на рынке труда за счёт практической подготовки.

Задачи проекта:

1. Разработка фронтенда веб-сайта проекта с использованием React.
2. Разработка бэкенда на базе Django.
3. Программирование интерфейсов для взаимодействия с системами виртуализации и контейнеризации: VirtualBox, KVM, Docker.
4. Создание виртуального макета города, имитирующего ИТ-инфраструктуру для тренировки и тестирования навыков.
5. Формирование и оформление статического веб-сайта для публичного представления проекта.
6. Проведение научных исследований в области кибербезопасности.

**2. Общая характеристика деятельности** **организации**

Наименование заказчика: кафедра «СМАРТ-технологии» Московского Политехнического университета.

Организационная структура:

1. Заведующий кафедрой – отвечает за общее руководство деятельностью кафедры, организацию и реализацию образовательных программ, практик у студентов, а также курирует взаимодействие с внешними партнерами.
2. Профессорского-преподавательский состав – включает в себя профессоров, преподавателей и доцентов, которые ведут лабораторные, практические и лекционные занятия.
3. Аспиранты и научные сотрудники – проводят исследования и учувствуют в выполнение проектов, принимают участие в организации студенческой научной деятельности и проектной работы.
4. Сотрудники и вспомогательный персонал – осуществляют техническую поддержку учебного процесса.
5. Студенческое сообщество – включает в себя обучающихся, участвующих в научной, образовательной и внеучебной деятельности.

Описание деятельности: кафедра «СМАРТ-технологии» занимается подготовкой специалистов в программной инженерии, интернет вещей, интеллектуальных информационных систем. Основными направлением деятельности кафедры является проведение научно-исследовательских работ, разработка образовательных программ, организация учебных и производственных практик обучающихся.

**3. Описание задания по проектной практике**

В рамках прохождения учебной практики мне было выдано задание — разработать два программных продукта: статический веб-сайт с информацией о проекте и Telegram-бот для автоматизации работы с расписанием сессии студентов. Для реализации этих задач были использованы современные технологии и инструменты:

1. GitHub — платформа применялась для создания репозитория проекта, ведения истории изменений, а также для организации командной работы и хранения исходного кода.

2. MarkDown — использовался для оформления проектной документации, написания инструкций по запуску и описания функционала.

3. HTML — применялся для структурирования страниц сайта, создания навигационного меню, размещения текстовой и графической информации о проекте.

4. CSS — использовался для стилизации веб-страниц, создания современного и адаптивного интерфейса, обеспечивающего корректное отображение сайта на различных устройствах.

5. Python — основной язык программирования для создания Telegram-бота, отвечающего за обработку запросов пользователей и взаимодействие с базой данных.

6. python-telegram-bot — библиотека для интеграции с Telegram API и реализации логики бота.

7. SQLite — база данных для хранения расписания сессии и работы с данными внутри Telegram-бота.

Основные этапы, которые были выполнены в ходе работы над проектом:

1. Создание репозитория на GitHub и настройка локального рабочего окружения для ведения проекта.

2. Разработка клиентской части сайта: создание структуры страниц, реализация основных элементов интерфейса, наполнение ключевых разделов содержимым.

3. Оформление документации и инструкций с помощью разметки MarkDown.

4. Разработка Telegram-бота: проектирование структуры базы данных, написание серверной логики, реализация функций для работы с расписанием и отправки уведомлений.

5. Проведение тестирования сайта и бота для проверки корректности работы, удобства интерфейса и стабильности функционирования на различных устройствах и в разных условиях.

В результате был создан удобный веб-сайт с информацией о проекте и полностью функционирующий Telegram-бот, который автоматизирует процесс получения расписания сессии для студентов.

**4. Описание достигнутых результатов по проектной практике**

**4.1 Настройка Git и организация командной разработки**

В начале работы над проектом я создал собственный репозиторий на GitHub, что позволило удобно хранить и структурировать все файлы, связанные с разработкой сайта и Telegram-бота. Такой подход обеспечил быстрый доступ к коду с любого устройства и дал возможность отслеживать все изменения, вносимые в проект.

В процессе разработки я активно применял основные возможности системы контроля версий Git:

1. Загружал репозиторий на свой компьютер для локальной работы.
2. Сохранял этапы разработки с помощью коммитов, чтобы фиксировать каждое важное изменение.
3. Периодически отправлял изменения обратно на GitHub, чтобы не потерять прогресс и иметь резервную копию.
4. Для отдельных задач (например, добавления новых функций или исправления ошибок) создавал отдельные ветки, а после завершения работы объединял их с основной версией проекта.

Такой подход позволил мне параллельно развивать разные части проекта — сайт и Telegram-бота — не мешая основной работе и не создавая конфликтов в коде. Благодаря регулярной фиксации изменений я мог легко вернуться к предыдущим версиям, если это было необходимо, и всегда был уверен в сохранности данных.

В целом, использование GitHub стало для меня неотъемлемой частью процесса разработки, помогая поддерживать порядок в проекте и эффективно управлять всеми этапами работы.

**4. 2** **Работа с документацией в формате MarkDown**

В процессе работы над проектом я освоил основы разметки Markdown и применил эти знания при подготовке всей проектной документации. Такой формат оказался очень удобным для структурирования информации и оформления различных разделов.

В ходе практики я использовал следующие возможности Markdown:

1. Для создания заголовков разного уровня применял символ # (от одного до шести), что позволяло удобно делить текст на смысловые блоки.
2. Для отделения абзацев достаточно было оставить пустую строку, а для переноса строки — добавить два пробела в конце строки.
3. Чтобы выделить текст курсивом, использовал одну звёздочку или нижнее подчёркивание с обеих сторон слова или фразы.
4. Для выделения жирным шрифтом применял двойные звёздочки или двойное подчёркивание.
5. Если нужно было сделать текст одновременно жирным и курсивным, использовал тройные звёздочки.
6. Для зачёркнутого текста применялись двойные тильды.
7. Для оформления цитат ставил символ > в начале строки.
8. Списки формировал с помощью дефиса, звёздочки или плюса для маркированных, а для нумерованных — использовал числа с точкой.
9. Для вставки изображений использовал конструкцию вида ![описание](ссылка).

Markdown оказался очень простым и интуитивно понятным инструментом, который значительно ускорил процесс оформления текстов и сделал документацию более читабельной и структурированной.

**4.3 Создание статического сайта**

В ходе практики мной был разработан статический сайт, посвящённый проекту «Киберполигон». Этот сайт стал итогом проделанной работы и содержит всю основную информацию о проекте, его целях и этапах реализации.

Для создания сайта я использовал современные веб-технологии:

1. HTML — основной инструмент для построения структуры страниц, размещения текстового и графического контента.
2. CSS и JS — применялись для стилизации элементов, задания цветовой схемы, шрифтов, отступов, а также для добавления анимаций и адаптивности интерфейса.

**4.3.1 Главная страница**

На главной странице сайта представлены (см. Рис. 1):

1. Название проекта, сразу привлекающее внимание посетителя.
2. Краткое описание сути и задач проекта.
3. Иллюстрация, соответствующая тематике информационной безопасности и визуальному стилю сайта.
4. Кнопка меню, расположенная в левом верхнем углу для удобной навигации.

Основная задача при разработке этой страницы заключалась в том, чтобы заинтересовать пользователя с первых секунд, сделать интерфейс интуитивно понятным и мотивировать к дальнейшему изучению информации о проекте.

**4.3.2 Страница «О проекте»**

Данная страница содержит сжатую, но информативную сводку о сути проекта. Вся информация структурирована по нескольким ключевым блокам (см. Рис. 2):

1. Краткое описание направления и задач проекта.
2. Перечень целей, которые ставились перед разработчиками.
3. Иллюстрация, визуально поддерживающая тематику

информационной безопасности.

* + 1. **Страница «Команда»**

На этой странице представлен состав разработчиков сайта. Был создан отдельный блок участника, где указаны его фамилия и имя, роль в проекте, а также основные задачи, за которые он отвечал в процессе работы (см. Рис. 3).

**4.3.4 Страница «Журнал»**

В данном разделе собрана хронология работы над проектом. Здесь размещаются записи, каждая из которых содержит заголовок и перечень выполненных задач. Все посты сгруппированы по месяцам, что позволяет быстро проследить динамику и этапы реализации проекта (см. Рис. 4).

**4.3.5 Страница «Ресурсы»**

На этой странице собраны ссылки на полезные материалы. Пользователь может найти информацию об организации, где проходила практика, а также ознакомиться с тематическими статьями по кибербезопасности и перейти на сайт проекта «Киберполигон» (см. Рис. 5).

**4.4 Взаимодействие с организацией-партнёром**

В рамках Карьерного марафона, проходившего 22 апреля 2025 года, мне довелось поучаствовать во взаимодействии с представителями организации-партнёра в качестве волонтёра. Это позволило получить ценный опыт командной работы и познакомиться с реальными задачами, которые решаются в профессиональной среде.

**4.5 Проверка и отладка сайта**

Для того чтобы убедиться в корректной работе сайта и соответствии результата поставленным задачам, я провёл серию тестов:

**Тест 1. Проверка работы навигации**

Цель: Оценить, насколько корректно функционируют переходы между разделами сайта.

Действия:

* Открыл сайт в браузере, находясь на разных страницах.
* Перешёл по всем пунктам меню (Главная, О проекте, Команда, Журнал, Ресурсы).
* Проверил, что каждая ссылка ведёт на нужный раздел, а страницы открываются без ошибок.

Результат: Все переходы работают корректно, ошибки не обнаружены.

**Тест 2. Проверка отображения главной страницы**

Цель: Убедиться, что ключевые элементы стартовой страницы отображаются правильно.

Действия:

* Перешёл на главную страницу сайта.
* Проверил наличие и корректность отображения заголовка, описания, изображения и кнопки меню.

Результат: Вся информация отображается корректно, структура страницы не нарушена.

**Тест 3. Проверка адаптивности интерфейса**

Цель: Оценить, как сайт выглядит на различных устройствах.

Действия:

* Открыл сайт на компьютере, планшете и смартфоне.
* Проверил, что меню, основной контент и подвал сайта корректно отображаются на всех экранах, текст не выходит за границы, элементы не перекрывают друг друга.

Результат: Сайт адаптируется под разные размеры экрана, навигация и отображение информации остаются удобными.

**Тест 4. Проверка содержимого раздела «Журнал»**

Цель: Убедиться, что хронология событий отображается последовательно и понятно.

Действия:

* Открыл страницу с журналом.
* Проверил наличие всех временных блоков (Февраль, Март, Апрель, Май 2025), а также наличие заголовков и описаний для каждого этапа.

Результат: Вся информация представлена в правильном порядке, текст читаем, анимация работает.

**Тест 5. Проверка загрузки изображений и иконок**

Цель: Убедиться, что все графические элементы сайта отображаются корректно.

Действия:

* Перешёл на каждую страницу сайта.
* Проверил, что все изображения и фавикон отображаются, отсутствуют битые ссылки и пустые блоки.

Результат: Все изображения и иконки загружаются без ошибок, визуальная часть сайта полностью соответствует ожиданиям.

**4.6 Проверка и отладка Telegram-бота**

Для того чтобы убедиться в корректной работе Telegram-бота и его соответствии поставленным задачам, я провёл ряд тестов:

**Тест 1. Проверка запуска и авторизации**

Цель: Убедиться, что бот успешно запускается и реагирует на команды.

Действия:

* Запустил бота в локальном окружении.
* Открыл чат с ботом в Telegram и отправил команду /start.
* Проверил, что бот отвечает приветственным сообщением и предлагает выбрать группу.

Результат: Бот запускается без ошибок, начальное взаимодействие работает корректно.

**Тест 2. Проверка выбора группы и отображения расписания**

Цель: Оценить корректность работы механизма выбора учебной группы и вывода расписания.

Действия:

* Выбрал одну из доступных групп.
* Проверил, что бот отображает расписание для выбранной группы.
* Переключился на другую группу и убедился, что расписание меняется.

Результат: Расписание отображается корректно для обеих групп, переключение работает без сбоев.

**Тест 3. Проверка фильтрации расписания**

Цель: Убедиться, что фильтры по дате, дню недели и типу события работают правильно.

Действия:

* Использовал фильтры для отображения расписания по дате, дню недели и типу события.
* Проверил, что бот корректно выводит только те события, которые соответствуют выбранным параметрам.

Результат: Фильтрация работает корректно, отображаются только релевантные события.

**Тест 4. Проверка работы кнопки «Назад» и навигации**

Цель: Оценить удобство возврата к предыдущим шагам и навигации по меню.

Действия:

* Перемещался между разделами бота, использовал кнопку «Назад» для возврата к выбору группы и главному меню.
* Проверил, что навигация работает без ошибок и не приводит к зависаниям.

Результат: Навигация реализована корректно, возврат к нужным разделам происходит без проблем.

**Тест 5. Проверка отправки уведомлений**

Цель: Убедиться, что бот автоматически отправляет напоминания о предстоящих экзаменах.

Действия:

* Изменил дату экзамена в базе данных на текущую.
* Дождался времени отправки уведомления (или инициировал вручную).
* Проверил, что бот отправляет сообщение с напоминанием и быстрыми кнопками для перехода к расписанию.

Результат: Уведомления приходят вовремя, содержимое сообщений соответствует ожиданиям.

**Заключение**

В ходе учебной практики мной был реализован комплексный проект, включающий разработку статического веб-сайта и Telegram-бота, посвящённых теме «Киберполигон». Все этапы работы выполнялись в соответствии с поставленными задачами и требованиями.

В процессе работы я использовал современные инструменты: GitHub для управления версиями и организации разработки, HTML и CSS для создания структуры и оформления сайта, Markdown для подготовки документации, а также Python и библиотеку python-telegram-bot для реализации функционального Telegram-бота. Такой подход позволил не только освоить новые технологии, но и закрепить уже имеющиеся навыки в области веб- и ботовой разработки.

В результате был создан сайт с продуманной структурой и современным оформлением, а также Telegram-бот, который автоматизирует получение расписания и уведомлений для пользователей. Проведённое тестирование подтвердило корректную работу всех функций и удобство взаимодействия с обоими продуктами.

Практика дала возможность не только углубить технические знания, но и получить ценный опыт самостоятельной работы, а также взаимодействия с представителями организации-партнёра. Всё это, безусловно, пригодится в дальнейшем обучении и профессиональной деятельности.

Таким образом, все поставленные цели были успешно достигнуты, а полученные результаты полностью соответствуют ожиданиям и требованиям учебной практики.

**Список литературы**

1. <https://cyberpolygon.netlify.app/>
2. <https://gohugo.io/getting-started/quick-start/>
3. <https://projects.mospolytech.ru/tproduct/413315123-945301126692-kiberpoligon>
4. <https://slavaver.github.io/web-course-site/>
5. https://developer.mozilla.org/ru/docs/Learn\_web\_development/Core/CSS\_layout/Introduction

**Приложение**

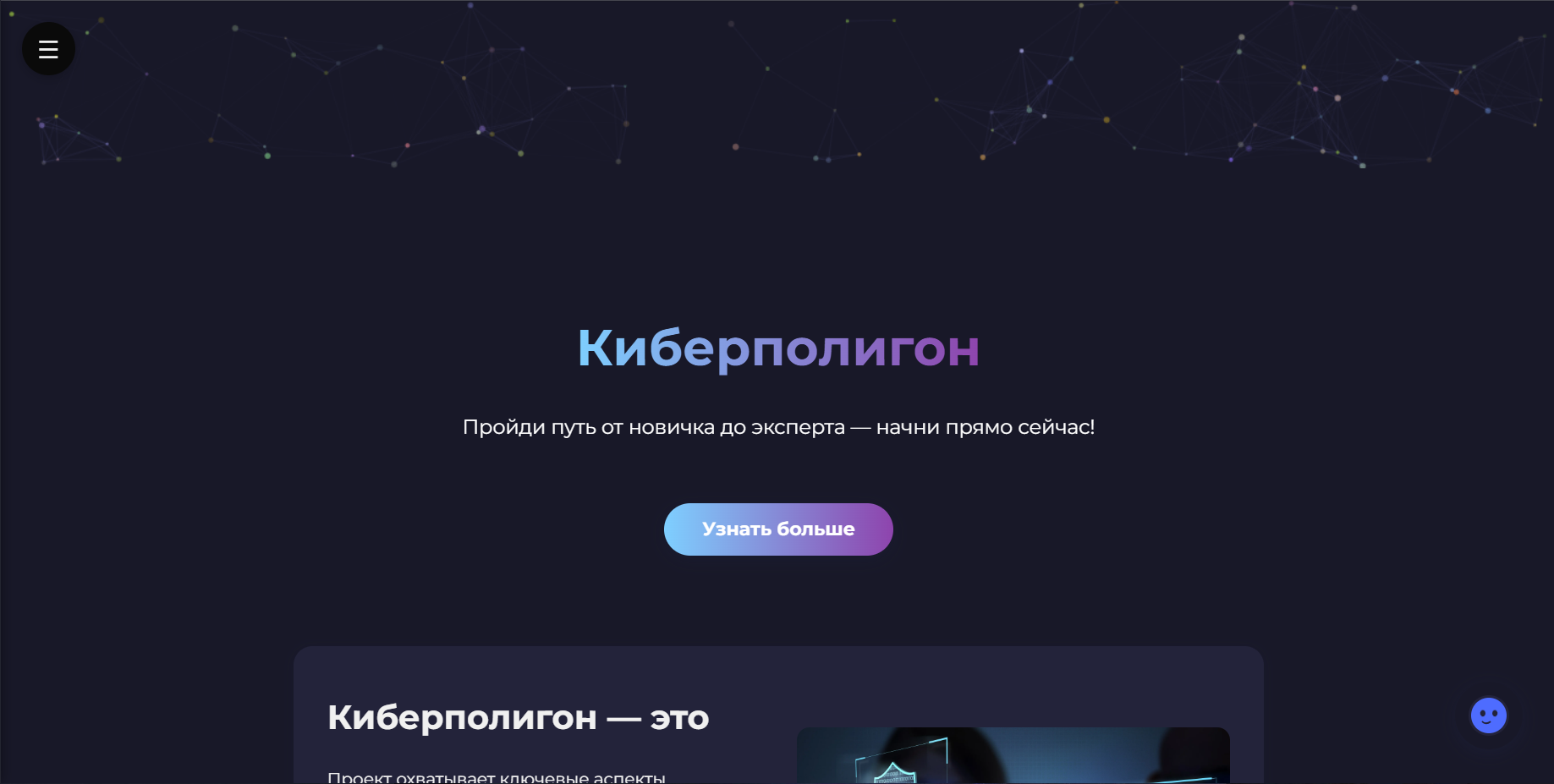
****

Рис. – Главная страница сайта

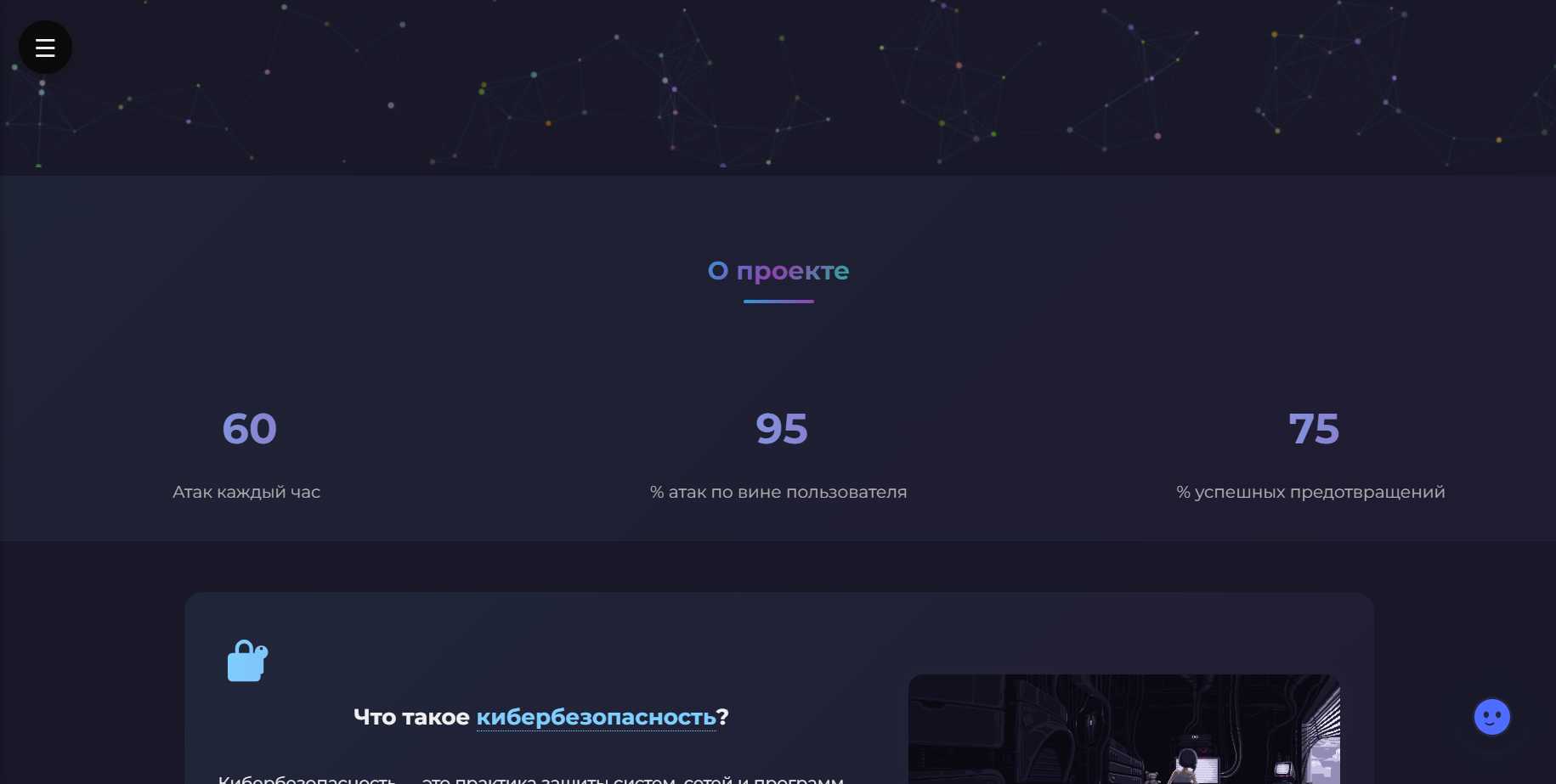


Рис. – Страница “О проекте”



Рис. – Команда разработки

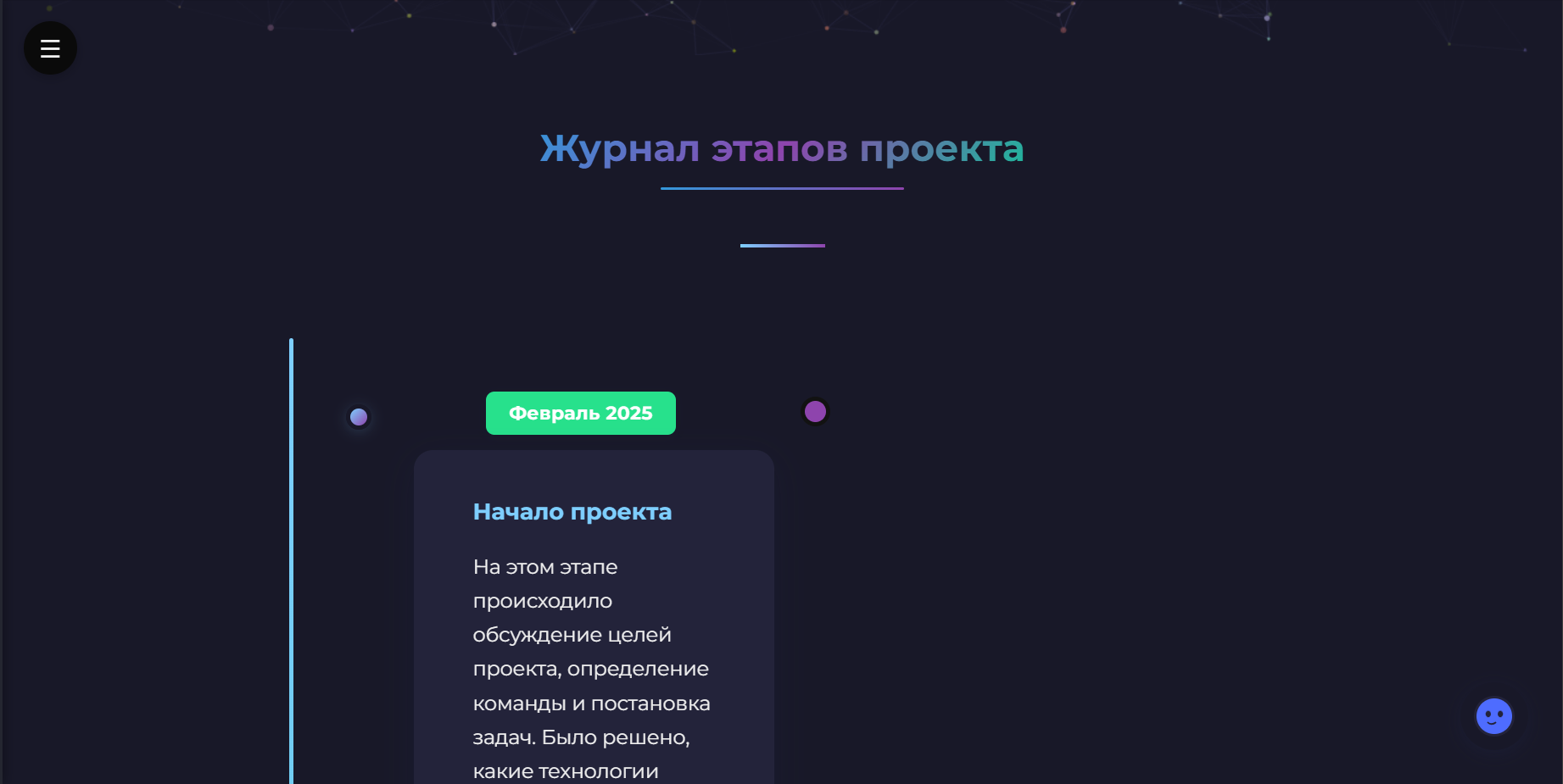


Рис. – Журнал этапов проекта

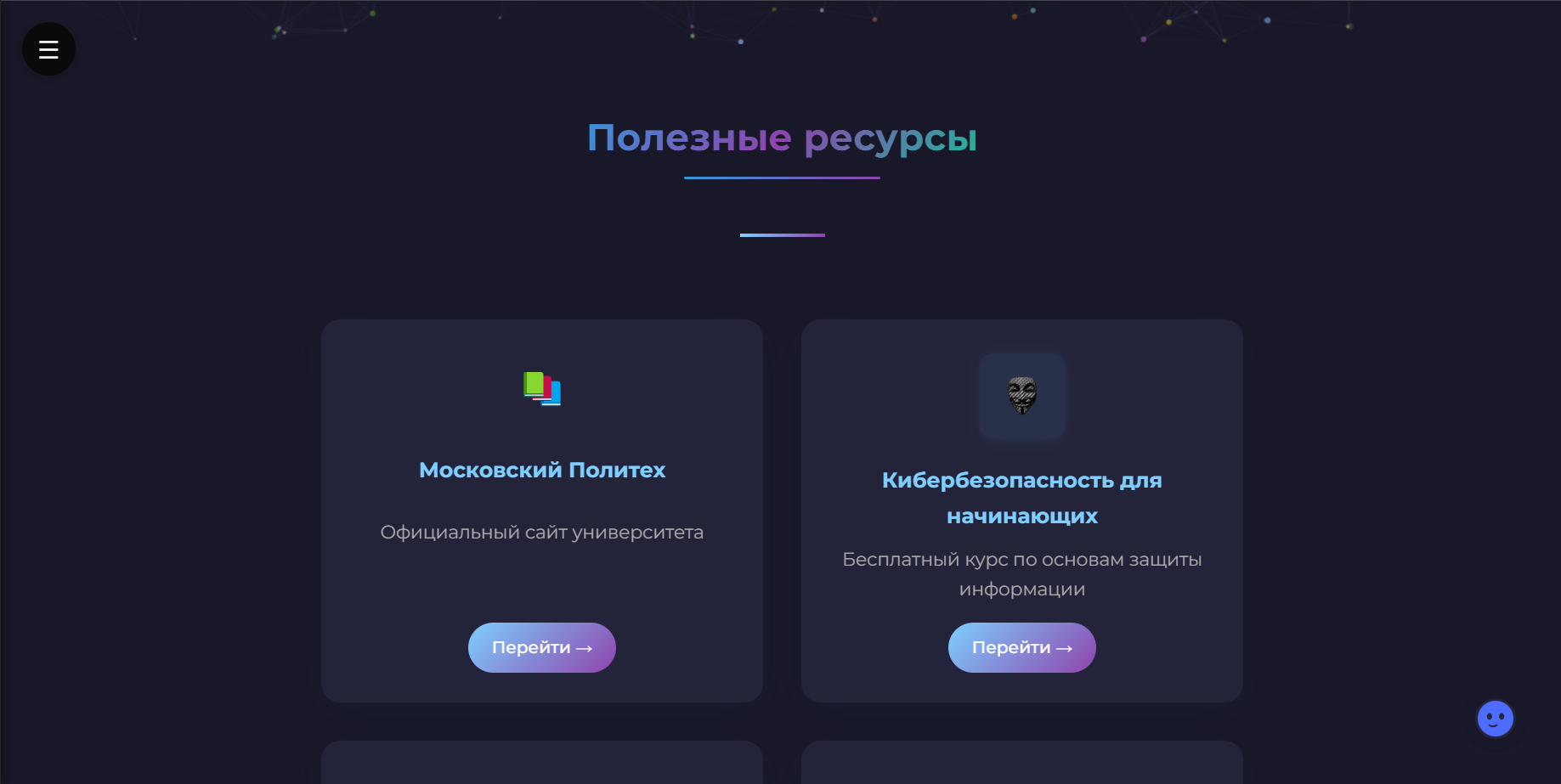


Рис. – Полезные ресурсы

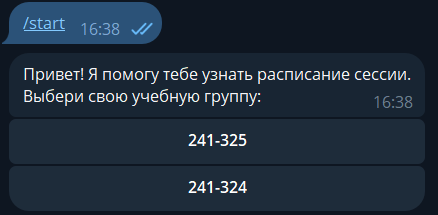


Рис. 6 – Первая страница Телеграм-бота с выбором группы

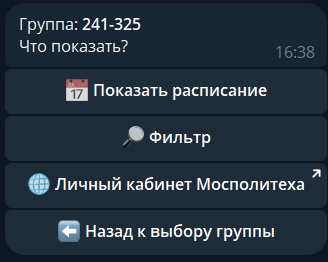


Рис. 7 – Страница после выбора группы

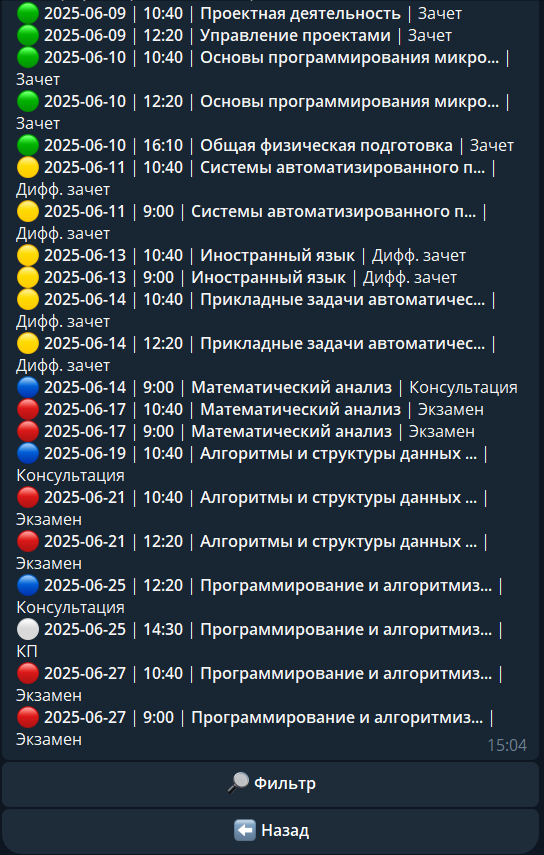


Рис. 7 – Расписание сессии

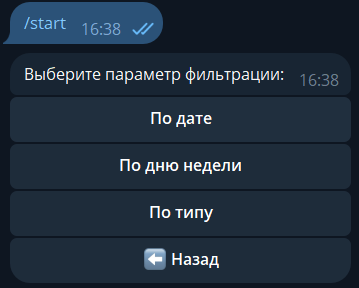


Рис. 8 – Возможность фильтрации экзаменов

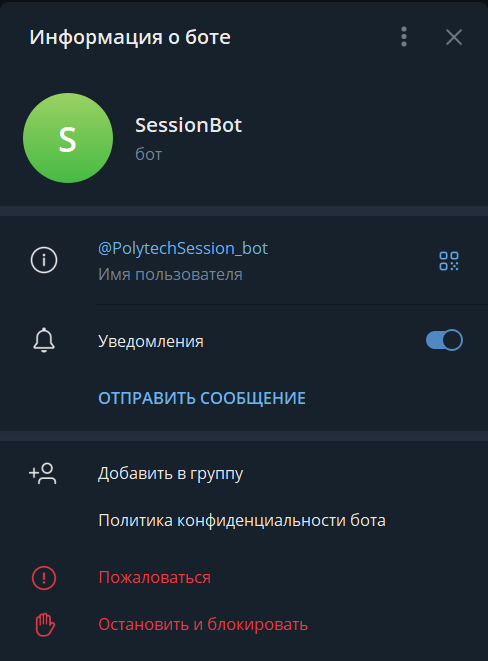


Рис. 9 – Телеграм-бот