

# **Отчёт по 1 этапу проекта**

**Сайт научного работника**

Фахми Джакси Гамал Адли

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>10</b>

## Список иллюстраций

2.1	Файл об авторе . . . . .	7
2.2	Файл для поста . . . . .	8
2.3	Файл для публикации . . . . .	9

## **Список таблиц**

# 1 Цель работы

Добавить к сайту данные о себе.

## **2 Выполнение работы**

Заполняю файл с данными о владельце сайта.

```

# Awards.
# Add/remove as many awards below as you like.
# Only 'title', 'awarder', and 'date' are required.
# Begin multi-line 'summary' with YAML's '|' or '|2-' multi-line prefix and indent 2 spaces below.
#awards:
# - title: Neural Networks and Deep Learning
#   url: https://www.coursera.org/learn/neural-networks-deep-learning
#   date: '2023-11-25'
#   awarder: Coursera
#   icon: coursera
#   summary:
---

## 0 себе

- Меня зовут Фахми Джакси Гамал Адли, я студент 1 курса РВУН.
- Учусь на факультете физико-математических наук, направление "Математика и механика".
- Интересуюсь математикой, программированием и вычислительной механикой.
- Люблю разбираться в сложных задачах и находить оптимальные решения.
- В свободное время изучаю C++, Linux и системы контроля версий.

```

Рис. 2.1: Файл об авторе

Заполняю файл с текстом поста.

## ## Итоги недели

На этой неделе произошло несколько интересных событий:

- Углубился в изучение численных методов и их применения в механике.
- Разбирался с основами комбинаторики и теории вероятностей – оказалось очень полезно!
- Улучшил навыки работы с `Git`: научился использовать `'rebase'` и `'cherry-pick'` для управления коммитами.
- Попробовал написать несколько алгоритмов на `C++` без использования `'std::random_device'` – пришлось проявить креативность!
- Параллельно осваивал новые возможности `Rocky Linux`, экспериментировал с настройкой окружения.

Неделя выдалась насыщенной, но продуктивной!

Рис. 2.2: Файл для поста

Заполняю файл с текстом публикации.



## ## Управление версиями. Git.

Git – это система контроля версий, которая позволяет отслеживать изменения в коде, работать в команде и безопасно вносить правки в проекты. Основные моменты:

- **\*\*Репозиторий\*\*** – хранилище проекта с полной историей изменений.
- **\*\*Коммиты\*\*** – зафиксированные изменения в коде с комментариями.
- **\*\*Ветки\*\*** – параллельные версии проекта, позволяющие разрабатывать новые функции без риска сломать основную ветку ('main' / 'master').
- **\*\*Слияние (merge)\*\*** – объединение изменений из одной ветки в другую.
- **\*\*Rebase\*\*** – перезапись истории коммитов для более чистой и логичной структуры.
- **\*\*Cherry-pick\*\*** – выборочное применение коммитов.

Основные команды:

```
'''sh
git init      # Создание нового репозитория
git clone URL # Клонирование удаленного репозитория
git add .     # Добавление всех изменений в индекс
git commit -m "Описание изменений" # Фиксация изменений
git push      # Отправка изменений в удаленный репозиторий
git pull      # Получение последних изменений
git branch    # Просмотр списка веток
git checkout -b новая_ветка # Создание и переключение на новую ветку
git merge ветка # Слияние ветки в текущую
git rebase main # Перенос коммитов поверх новой версии ветки
git log --oneline --graph # Просмотр истории коммитов в компактном виде
'''
```

Git – мощный инструмент, который помогает организовать работу над проектом, сохранять историю изменений и эффективно работать в команде. Освоение его возможностей – важный шаг для любого разработчика!

Рис. 2.3: Файл для публикации

Перекомпилирую сайт

## **3 Выводы**

Добавили к сайту данные о себе.