Отчет на тему 1

На начальном этапе я создал личный репозиторий, добавив:

* Файл README.md с описанием проекта.
* .gitignore для выбранного языка программирования.
* Папку src/ с базовым скриптом.

Все изменения фиксировались коммитами с понятными сообщениями, что позволило освоить ключевые функции GitHub.

Второй этап

Я исследовал возможности репозитория, изучив открытый проект на GitHub:

1. **Анализ истории коммитов**:
   * Обновления README.md.
   * Слияния через pull request.
   * Удаление дублирующихся шагов и исправление опечаток в data.json.
2. **Структура проекта**:
   * Папки: .github, contributing.md.
   * Файлы: README.md, data.json, .gitignore, документация и workflow-файлы.

Мини-ревью

Проект соответствует принципам чистого кода:

* Четкая логика и понятная структура.
* Единый стиль оформления.
* Лаконичные и информативные названия.
* Отсутствие избыточности, комментарии для сложных алгоритмов.
* Каждый метод решает одну задачу.

Качество коммитов

* **Сообщения**: Должны быть краткими, ясными и объяснять суть изменений. Плохие сообщения усложняют анализ истории.
* **Атомарность**: Каждый коммит — одна логическая правка. Это упрощает код-ревью и отслеживание изменений.

Инструменты код-ревью

* **GitHub Pull Request**: Для обсуждения и контроля изменений.
* **Автоматические проверки**: Анализ стиля, ошибок и уязвимостей.

**Критерии эффективности**:

1. Выявление ошибок.
2. Оптимизация производительности.
3. Соответствие стандартам.
4. Обмен знаниями в команде.

Проблемы в истории коммитов

* Смешанные правки (фичи, багфиксы, документация в одном коммите).
* Частичные или отмененные изменения.
* Неточные описания коммитов.
* Лишние файлы (логи, временные данные).

Отчет на тему 2

Цели ревью

* Проверка соответствия идиомам языка.
* Использование стандартных библиотек.
* Отсутствие логических ошибок.

Задачи ревьюера

* Проверить соответствие PEP-8.
* Оценить читаемость и именование.
* Исключить избыточную сложность.

Объекты проверки

* Читаемость и структура кода.
* Соответствие стилевым стандартам.
* Тесты и безопасность.
* Поддержка и документация.

Чек-лист

1. **Работоспособность**: Корректность логики и отсутствие ошибок.
2. **Понятность**: Ясные названия, комментарии.
3. **Стиль**: Соответствие PEP-8/Google Style.
4. **Избыточность**: Повторяющиеся участки.
5. **Обработка ошибок**: Логирование и валидация.
6. **Безопасность**: Проверка входных данных.
7. **Документация**: Описание функций и параметров.
8. **Тесты**: Покрытие основных сценариев.

План ревью

|  |  |  |  |
| --- | --- | --- | --- |
| **Этап** | **Действия** | **Срок** | **Участники** |
| Подготовка | Выбор кода, назначение ревьюера | 45 мин | Tech Lead, Ревьюер |
| Анализ | Проверка по чек-листу | 2 часа | Ревьюер |
| Обсуждение | Разбор вопросов с автором | 30 мин | Автор, Ревьюер |
| Исправление | Правки кода | 1-2 дня | Автор |
| Повторная проверка | Анализ исправлений | 1 час | Ревьюер |
| Финальное утверждение | Подписание итоговой версии | 30 мин | Tech Lead, Автор |

**Параметры модуля**:

* 200 строк кода.
* Общее время: 4-5 часов.

Сложности и улучшения

**Проблемы**:

* Определение критериев оценки.
* Баланс времени между этапами.
* Риск упустить важные детали.

**Оптимизация**:

* Автоматизация (pylint, SonarQube).
* Шаблоны чек-листов.
* Регулярные ретроспективы.
* Обучение команды.
* Четкие временные рамки.

Отчет на тему 3

Сравнение офисных пакетов

|  |  |  |  |
| --- | --- | --- | --- |
| **Критерий** | **Microsoft Office 365** | **Google Workspace** | **LibreOffice** |
| Совместимость форматов | Хорошая (DOCX → ODT) | Умеренная | Отличная (нативный ODT) |
| Стоимость | От $6.99/мес | От $6/мес | Бесплатно |
| Совместное редактирование | Реальное время (OneDrive) | Лучшее (Google Docs) | Ограниченное (NextCloud) |
| Скрипты | VBA, Power Automate | Google Apps Script | Python, API |
| Офлайн-работа | Полная поддержка | Только с офлайн-режимом | Полностью офлайн |

**Выводы**:

* **Office 365**: Для корпораций и интеграции с Microsoft-сервисами.
* **Google Workspace**: Для онлайн-коллаборации и мобильной работы.
* **LibreOffice**: Для локального использования и конфиденциальности.

Сравнение браузеров

* **Chrome**: Лучшая библиотека расширений, но высокое потребление RAM.
* **Firefox**: Оптимизация энергопотребления, умеренная скорость.
* **Edge**: Быстрый запуск, совместимость с Chrome-расширениями.

Сравнение видеоплееров

* **VLC**: Лучшая поддержка кодеков и скриптов (Lua).
* **MX Player**: Оптимизация под Android, автопоиск субтитров.
* **KMPlayer**: Гибкие настройки для Windows.

**Рекомендации**:

* Экосистема: Microsoft/Google.
* Бесплатность: LibreOffice + Firefox + VLC.
* Мобильность: Google Workspace + MX Player.
* Профессиональное использование: Office 365 + Chrome/Edge + VLC.