МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

РАЗРАБОТКА ФРЕЙМВОРКА ДЛЯ ГЕНЕРАЦИИ СТАТИЧЕСКИХ САЙТОВ

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

На 15 листах

Действует с «27» ноября 2024 г.

СОГЛАСОВАНО и УТВЕРЖДЕНО

Власовым Дмитрием Викторовичем

кандидат физико-математических наук,   
доцент кафедры ИТиЭО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

Санкт-Петербург

2024

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_30j0zll)

[ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ 4](#_1fob9te)

[Требования к функциональным характеристикам 6](#_tyjcwt)

[Требования к надежности 6](#_3dy6vkm)

[Условия эксплуатации 7](#_1t3h5sf)

[Требования к техническим средствам 7](#_4d34og8)

[ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 9](#_26in1rg)

[ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ 12](#_35nkun2)

[ПРИЛОЖЕНИЕ А 14](#_1ksv4uv)

# ВВЕДЕНИЕ

**Наименование:** разработка фреймворка для генерации статических сайтов.

**Краткая характеристика области применения**: Программное изделие используется в сфере веб-разработки для создания и управления статическими веб-сайтами. Фреймворк позволяет разработчикам и контент-менеджерам автоматизировать процесс создания веб-сайтов из Markdown-файлов, применять шаблоны оформления и управлять структурой сайта без необходимости написания HTML-кода вручную. Система особенно эффективна для проектов, где важны безопасность, производительность и простота поддержки контента.

**Объект, в котором используется программа:**

* Системы управления контентом (CMS)
* Генераторы статических сайтов (Jekyll, Hugo)
* Инструменты веб-разработки
* Системы контроля версий (Git)
* Платформы хостинга (GitHub Pages, Netlify, Vercel)

Фреймворк интегрируется с существующими инструментами разработки и может быть использован как самостоятельное решение или как часть более крупного процесса разработки веб-проектов.

# ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

Приказ №\_\_\_\_\_\_\_\_ от \_\_.\_\_.\_\_\_\_, утвержденный проректором по образовательной деятельности В. И. Снегурова

НАЗНАЧЕНИЕ РАЗРАБОТКИ

**Функциональное назначение:** Фреймворк предназначен для автоматизированной генерации статических веб-сайтов из исходных Markdown-файлов и шаблонов. Основные функции включают:

* Преобразование Markdown в HTML
* Применение пользовательских шаблонов
* Автоматическая генерация навигации и индексов
* Управление метаданными и категориями контента
* Оптимизация ресурсов (изображения, CSS, JavaScript)
* Генерация карты сайта и RSS-потоков
* Поддержка многоязычности

**Эксплуатационное назначение:** Фреймворк будет использоваться веб-разработчиками, контент-менеджерами и техническими специалистами для:

* Быстрого создания и развертывания веб-сайтов
* Эффективного управления контентом через систему версионирования
* Обеспечения высокой производительности веб-ресурсов
* Упрощения процесса обновления и поддержки сайтов
* Снижения затрат на хостинг и обслуживание
* Обеспечения безопасности веб-ресурсов
* Автоматизации рутинных задач по обновлению контента

ТРЕБОВАНИЯ К ПРОГРАММЕ ИЛИ К ПРОГРАММНОМУ ИЗДЕЛИЮ

# Требования к функциональным характеристикам

**Функции:**

* Конвертация Markdown в HTML
* Применение пользовательских шаблонов
* Генерация навигации и индексных страниц
* Обработка метаданных
* Оптимизация медиа-контента
* Генерация карты сайта и RSS-лент

**Организация входных и выходных данных:** Входными данными являются Markdown-файлы, шаблоны, конфигурационные файлы YAML, а выходным данными являются HTML-файлы, оптимизированные ресурсы, служебные файлы.

**Временные характеристики:** Время сборки небольшого сайта (до 100 страниц) не более 5 секунд.

# Требования к надежности

Гарантированная работа при корректных входных данных. Валидация входных файлов и конфигураций. Логирование всех операций.

**Обработка ошибок:** Информативные сообщения об ошибках, возможность отката изменений, сохранение промежуточных результатов.

**Восстановление:** Автоматическое создание резервных копий. Время восстановления после сбоя не более 30 секунд.

# Условия эксплуатации

### Требования к окружению: Любая операционная система с поддержкой Python 3.8+, доступ к файловой системе, подключение к интернету (для загрузки зависимостей).

**Пользователи**: Веб-разработчики, контент-менеджеры, технические специалисты.

# Требования к техническим средствам

**Минимальные требования**: Процессор: 2 ядра, 2.0 GHz; ОЗУ: 4 GB; Дисковое пространство: 1 GB; Сетевое подключение: 1 Mbps.

**Программное обеспечение:** Python 3.8 или выше; Git (опционально); Текстовый редактор.

# Требования к информационной и программной совместимости

### Технологии и стандарты: Python 3.8+, Markdown, YAML, HTML5/CSS3, Jinja2.

### Безопасность: Валидация входных данных, безопасная обработка файлов, защита от XSS-атак.

# Требования к маркировке и упаковке

### Распространение: PyPI пакет, GitHub репозиторий, документация в формате Read the Docs.

### Маркировка: Версия продукта, список зависимостей, лицензия, контактная информация.

## Требования к транспортированию и хранению

* Распространение через PyPI
* Хранение исходного кода в Git-репозитории
* Резервное копирование в облачных сервисах

# ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

### Состав документации:

* Техническое задание
* Руководство пользователя
* API документация
* Руководство по разработке
* Примеры использования

### Характеристики документации:

* Формат: Markdown/HTML
* Язык: русский и английский
* Доступность: онлайн и офлайн
* Версионирование документации

### Требования к содержанию:

* Подробные инструкции по установке
* Примеры использования
* API референс
* Руководство по созданию плагинов
* Описание архитектуры
* Рекомендации по оптимизации

СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

| **Стадии** | **Этапы** |
| --- | --- |
| Исследовательская стадия | Анализ существующих генераторов статических сайтов |
| Изучение современных подходов к разработке фреймворков |
| Определение оптимальной архитектуры |
| Выбор технологий и библиотек |
| Разработка ядра фреймворка | Базовая функциональность:  * Разработка системы парсинга Markdown * Создание системы шаблонов * Реализация базовой CLI * Разработка системы конфигурации |
| Система плагинов:  * Разработка архитектуры плагинов * Создание базовых плагинов * Реализация системы хуков |
| Разработка дополнительных компонентов | Система кэширования |
| Генератор RSS |
| Система тегов и категорий |
| Поддержка многоязычности |
| Оптимизация изображений |
| тестирование | Модульное тестирование  * Написание unit-тестов * Интеграционное тестирование * Нагрузочное тестирование |
| Пользовательское тестирование  * Тестирование на реальных проектах * Сбор обратной связи * Исправление ошибок |
| Оптимизация | Профилирование производительности |
| Оптимизация кода |
| Улучшение пользовательского опыта |
| Подготовка к релизу |
| Разработка  документации | Создание технической документации |
| Написание руководства пользователя |
| Подготовка примеров использования |
| Документация API |
| Дополнительные задачи | Создание демонстрационных проектов |
| Подготовка материалов для публикации |
| Настройка CI/CD |
| Публикация в PyPI |

**Общий срок разработки** – 6 месяцев.

# ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

## Виды испытаний:

### Функциональное тестирование

* Конвертация Markdown в HTML
* Работа системы шаблонов
* Функционирование плагинов
* Генерация навигации и индексов
* Многоязычность
* CLI-интерфейс

### Производительность

* Тестирование на больших проектах
* Оценка времени сборки
* Использование памяти
* Работа кэширования

### Интеграционное тестирование

* Совместимость с Git
* Работа с различными платформами хостинга
* Тестирование на разных ОС

## Требования к приемке:

### Документация

* Техническая документация
* Руководство пользователя
* API документация
* Примеры использования

### Критерии приемки

* Покрытие тестами > 80%
* Соответствие PEP 8
* Отсутствие критических ошибок
* Полная документация

## Этапы приемки:

Предварительное тестирование

* Базовая функциональность
* Производительность
* Качество кода

Финальная приемка

* Демонстрация продукта
* Проверка соответствия ТЗ
* Оценка документации

# ПРИЛОЖЕНИЕ А

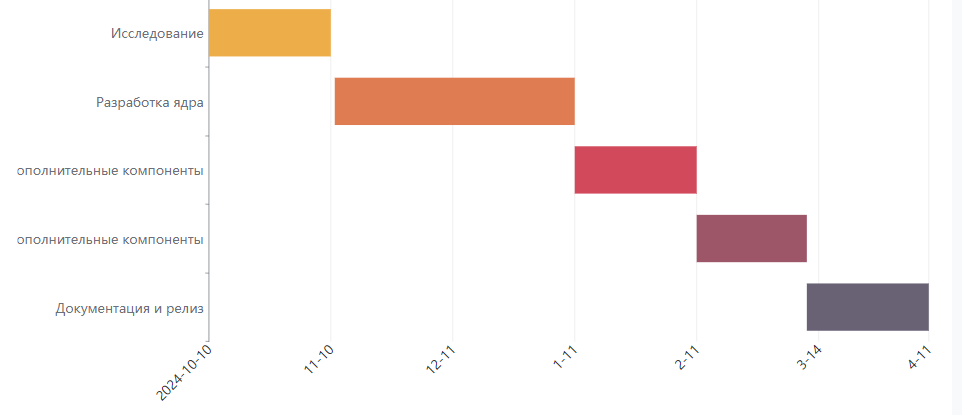


Рисунок 1 – Диаграмма Ганта

СОСТАВИЛ и ВЫПОЛНИЛ

Крючкова Анастасия Сергеевна

Студентка 4 курса, группа 1.2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«25» декабря 2024 г.