

Инструкции для коммита обновленной документации

Обзор изменений

Документация была синхронизирована с актуальной кодовой базой после исправления ошибок pylint (рейтинг улучшен с 8.37 до 8.53).

Измененные файлы

Обновленные файлы:

- `README.md` - обновлена архитектура, переменные окружения, функции безопасности
- `API.md` - полностью обновлен список endpoints и функций
- `README-DEPLOY.md` - добавлена информация о валидации переменных окружения

Новые файлы:

- `DOCUMENTATION_UPDATE_REPORT.md` - отчет об изменениях в документации
- `COMMIT_INSTRUCTIONS.md` - данный файл с инструкциями

Архивные файлы:

- `API_OLD.md` - старая версия API документации (для справки)

Команды для коммита

1. Проверка статуса

```
cd ~/workspace/baimuras_fixed
git status
```

2. Добавление измененных файлов

```
# Добавить обновленные файлы
git add README.md
git add API.md
git add README-DEPLOY.md

# Добавить новые файлы
git add DOCUMENTATION_UPDATE_REPORT.md
git add COMMIT_INSTRUCTIONS.md

# Добавить архивный файл (опционально)
git add API_OLD.md
```

3. Создание коммита

```
git commit -m "docs: Синхронизация документации с кодовой базой после pylint fixes"
```

- Обновлен README.md: архитектура проекта, новые модули безопасности
- Полностью обновлен API.md: 39 актуальных endpoints, новые функции
- Обновлен README-DEPLOY.md: валидация переменных окружения
- Добавлен отчет об изменениях в документации
- Pylint score: 8.53/10 (улучшен с 8.37)

Fixes: несоответствие документации актуальному коду

Adds: env_validator, security_middleware, secret_manager модули

Updates: API endpoints, переменные окружения, инструкции деплоя"

4. Создание ветки (если нужно)

```
# Создать новую ветку для изменений
```

```
git checkout -b docs/update-after-pylint-fixes
```

```
# Или переключиться на существующую
```

```
git checkout docs/update-after-pylint-fixes
```

5. Push изменений

```
# Push в текущую ветку
```

```
git push origin HEAD
```

```
# Или в конкретную ветку
```

```
git push origin docs/update-after-pylint-fixes
```

Описание для Pull Request

Заголовок PR:

```
docs: Синхронизация документации с кодовой базой после pylint исправлений
```

Описание PR:

Обновление документации

Синхронизация документации с актуальной кодовой базой после исправления ошибок pylint (рейтинг улучшен с 8.37 до 8.53).

Проведенный анализ

- Проверена актуальная структура проекта
- Извлечены реальные API endpoints из кода
- Выявлены новые модули безопасности
- Сравнена документация с фактическим состоянием

Основные изменения

README.md

- Обновлена архитектура проекта с новыми модулями
- Исправлен Pylint score (8.53 вместо 9.21)
- Добавлены новые модули безопасности
- Обновлена таблица переменных окружения

API.md

- Полностью обновлен список endpoints (39 маршрутов)
- Добавлены новые сервисные маршруты
- Документированы CRM endpoints
- Добавлена информация о функциях безопасности

README-DEPLOY.md

- Добавлена информация о валидации переменных окружения
- Требования к обязательным переменным
- Предупреждения о длине секретных ключей

Новые функции в документации

- Environment validation (env_validator.py)
- Security middleware (security_middleware.py)
- Secret management (secret_manager.py)
- Secure configuration (config_secure.py)

Технические улучшения

- Удален хардкод секретов
- Автоматическая валидация переменных окружения
- Улучшенная система логирования
- Дополнительные middleware безопасности

Статистика

- ****Файлов обновлено****: 3
- ****Новых файлов****: 2
- ****API endpoints****: 39 документированных
- ****Pylint score****: 8.53/10

Проверено

- [x] Все endpoints извлечены из реального кода
- [x] Переменные окружения соответствуют .env.example
- [x] Архитектура отражает фактическую структуру
- [x] Новые модули безопасности документированы

Результат

Документация теперь полностью соответствует актуальной кодовой базе и отражает все улучшения безопасности после pylint исправлений.

Важные замечания

1. Проверьте права доступа к репозиторию перед push
2. Убедитесь в корректности ветки для коммита
3. Проверьте `.gitignore` - не коммитьте служебные файлы
4. Создайте `backup` перед внесением изменений

После коммита

1. Создать Pull Request с описанием выше
2. Запросить review у команды
3. После merge обновить локальную ветку main
4. Удалить временную ветку (если использовалась)

Подготовлено: 21 июня 2025

Статус: Готово к коммиту

Версия документации: 2.1.0