# Тезисы доклада для предзащиты ВКР

1. Актуальность темы  
Современная разработка программного обеспечения требует высокой скорости и надёжности поставки. Автоматизация процессов развертывания через CI/CD позволяет минимизировать человеческий фактор и ускорить цикл поставки, что особенно важно при разработке веб-сервисов.

2. Цель и задачи работы  
Цель — разработка и внедрение CI/CD пайплайна для автоматического деплоя веб-приложения.  
Задачи:  
- Создание MVP приложения по подсчёту калорий (backend и frontend);  
- Контейнеризация компонентов с помощью Docker;  
- Настройка автоматического и ручного деплоя в разные среды (dev/prod);  
- Реализация безопасного и масштабируемого CI/CD пайплайна с GitHub Actions.

3. Архитектура решения  
- Backend: Django + DRF, REST API, хранение записей о приёмах пищи.  
- Frontend: React, отправка и визуализация данных.  
- Docker Compose: изолированные контейнеры.  
- Nginx: проксирование и маршрутизация.

4. CI/CD пайплайн  
- Три отдельных workflow: для backend, frontend и nginx;  
- Автодеплой на dev при коммите в main-ветку;  
- Ручной деплой на prod через GitHub interface (workflow\_dispatch);  
- Использование SSH и GitHub Secrets для защищённого доступа.

5. Особенности реализации  
- Использована модульная архитектура;  
- Все переменные окружения вынесены в .env;  
- Логирование ведётся через stdout и GitHub logs;  
- Учтены ошибки: SSH, Docker, CORS, переменные и конфликты версий.

6. Результаты работы  
- Приложение развёрнуто на двух стендах;  
- CI/CD обеспечивает быструю доставку и гибкий контроль релизов;  
- Возможность масштабирования и расширения проекта под новые задачи.

7. Выводы и перспективы  
- Разработанное решение может быть применено в коммерческих и образовательных проектах;  
- В перспективе — интеграция с мониторингом, логированием, тестированием, Kubernetes.