

## Оглавление

1	Введение.....	2
2	Разработка MVP .....	2
2.1	Авторизация .....	2
2.2	Инвентаризация .....	2
3	Разработка frontend.....	2
3.1	Стартовая страница .....	3
3.2	Страница «О нас» .....	3
3.3	Страница «Авторизации» .....	3
3.4	Страница «Инвентаризация».....	3
4	CI/CD-процесс .....	3
5	Тестирование .....	4
6	Проблемы и решения .....	5
7	Итоги разработки MVP.....	6

## **1 Введение**

«Storix» - веб-приложение для инвентаризации и управления складом, которое упрощает процесс проведения инвентаризаций за счёт использования ИИ и минимизирует вероятность ошибок.

Начало разработки MVP – 1 апреля 2025 года. Конец разработки MVP – 6 мая 2025 года.

## **2 Разработка MVP**

На данный момент реализован MVP продукт. Пользователь (рабочий) может авторизоваться и загрузить видео со склада.

### **2.1 Авторизация**

В рамках данного раздела была реализована авторизация пользователей (рабочих).

### **2.2 Инвентаризация**

Данный раздел отвечает за проведение инвентаризации по загруженному видео. Система позволяет увидеть сходства и расхождения в товарах и их местоположении на складе благодаря ИИ анализу.

## **3 Разработка frontend**

Разработка клиентской части велась с учетом макетов Figma. Фреймворком для реализации был выбран React.

Были разработаны следующие страницы:

- Стартовая страница
- Страница «О нас»
- Страница авторизации
- Страница инвентаризации.

### **3.1 Стартовая страница**

При попадании на сайт неавторизованный пользователь попадает на стартовую страницу. Здесь пользователь может кратко узнать о возможностях нашего сайта. Также с этой страницы можно перейти на две другие: «Авторизация» и «О нас».

### **3.2 Страница «О нас»**

Здесь неавторизованный пользователь может подробнее узнать о возможностях нашего сайта: цену и шаги, которые необходимо сделать для проведения инвентаризации. Также тут присутствует кнопка для совершения покупки.

### **3.3 Страница «Авторизации»**

На данной странице неавторизованный пользователь может войти в свой аккаунт по логину и паролю. После входа его перенаправляет на страницу «Инвентаризация».

### **3.4 Страница «Инвентаризация»**

На данной странице пользователь (рабочий) может загрузить своё видео для проведения инвентаризации.

Загрузка видео происходит при нажатии на соответствующую форму в центре. После загрузки появляется окно предпросмотра для проверки. После отправки видео происходит его анализ. Появляется таблица, показывающая сходства и расхождения в местоположении товаров.

## **4 CI/CD-процесс**

Для обеспечения стабильности и скорости релизов в нашем проекте реализован минимальный, но функциональный процесс CI/CD. В репозитории в `.github/workflows/ci.yml` настроили автоматический запуск при каждом push и pull\_request в ветку main.

Описание:

- Запускаем контейнер Postgres, чтобы тесты работали не на SQLite, а в окружении, максимально приближенном к продакшену.
- Создаём чистую базу данных под тесты.
- Прогоняем pytest, собираем отчёт по покрытию кода.

Таким образом, весь процесс от коммита до деплоя минимизирует участие команды, позволяя оперативно вносить изменения и снижать количество ошибок на продакшене.

## 5 Тестирование

В рамках этой задачи бизнес-логика была покрыта юнит- и интеграционными тестами.

Для backend-модулей написан набор юнит-тестов, покрывающий ключевые компоненты:

- Проверка моделей: убедились, что склады правильно отображаются по имени, что строки для видео и отчёта содержат понятные слова, и что у администратора и у рабочего правильно устанавливаются связи со складом.
- Проверка прав доступа: убедились, что только системный админ может выполнять sysadmin-операции, только админ — admin-операции, а только рабочий — worker-операции, и что они не заходят чужие роли.
- Проверка сериализатора пользователя: убедились, что при создании через API или сериализатор, пароль автоматически хэшируется и потом корректно проверяется.
- Проверка API для аккаунтов: системный админ может создавать администраторов. Обычный админ не может создавать других

админов. Админ создаёт склад и рабочего через API, и эти объекты появляются в системе с нужными связями. Проверка модели отчёта по инвентаризации: убедились, что новый отчёт стартует в статусе “pending” и без результата.

- Проверка сериализатора отчёта: смотрим, что при выводе отчёта через API включаются все важные поля: видео, конфиг, статус, время создания и т. д.
- Проверка функции анализа видео: при пустом видео она возвращает «ничего не найдено» и флаг, что всё совпадений равно пустому набору. При искусственных кадрах и пустом декодировании показывает, что ни коробка, ни стеллаж не найдены.
- Проверка API для инвентаризации: рабочий через API загружает видео и конфиг, получает созданный отчёт, а затем запускает на нём анализ — и получает ответ со статусом выполнения.

## **6 Проблемы и решения**

В процессе разработки проекта возникло несколько организационных и технических затруднений, которые повлияли на ход реализации.

При внедрении оплаты мы столкнулись с юридическими трудностями в виде регистрации ИП или Юридического лица, а также в необходимости прохождения модерации. Для MVP-версии кнопка оплаты не работает.

Render-сервере в бесплатном тарифе ограничен в мощности. Также бесплатные инстансы засыпают при простое, и при первом запросе происходит холодный старт, который занимает время.

Деплой на Render изначально использовал SQLite в контейнере, из-за чего при каждом пересоздании терялись все данные, поэтому мы перешли на

внешний PostgreSQL-сервис Render — теперь данные живут вне контейнера и не теряются.

Наконец, в процессе подготовки Docker-сборки мы обнаружили, что не передаётся секретный ключ, из-за чего статика не копировалась, и добавили передачу этого ключа в среду сборки, что решило проблему.

В итоге мы добились полной стабильности тестов, надёжной работы CI/CD и деплоя без потерь.

## **7      Итоги разработки MVP**

На момент 2 аттестации команда завершила основную разработку MVP.

- Были реализованы разделы авторизации и загрузки видео.
- Были сделаны стартовая страница, страница «О нас», страница авторизация и страница инвентаризации.
- Минимально подключен CI/CD процесс.
- Бизнес-логика покрыта юнит- и интеграционными тестами.
- Приложение успешно развернуто на удалённом Render-сервере.