

ООО Стандарт

Программа для ЭВМ
Automason WCS

Документация, содержащая описание
функциональных характеристик экземпляра
программного обеспечения

г. Москва 2025

Оглавление

Предназначение ПО	3
Состав ПО.....	4
Функциональная спецификация базовых компонент	5
errorsstorage	5
wms-facade.....	5
fleet-server	6
kartograph.....	6
pastuh.....	7
rms-facade	7
robot-manager.....	9
react-app	9
task-manager.....	10
telemetry	11
traffic-control.....	11
zavgar.....	12
Функциональная спецификация хранилищ	13
Функциональная спецификация вспомогательных компонент	14
Прочие функциональные особенности.....	15

Предназначение ПО

Серверное программное обеспечение Автомакон WCS (Warehouse Control System) предназначено для централизованного управления парком автономных AGV-роботов и автоматизированным оборудованием склада в режиме реального времени. Его основная задача — управление производственными процессами, координация перемещений, распределение задач и обеспечение бесперебойной работы всей системы.

Ключевой функцией WCS является управление приоритетами и очередностью задач. Система автоматически распределяет задания между роботами и оборудованием, оптимизируя общее время выполнения процессов. При изменении условий (например, появлении срочного заказа или неисправности оборудования) WCS мгновенно перераспределяет задачи, минимизируя простои.

Важным аспектом работы системы является разрешение конфликтов на транспортной сети. WCS анализирует маршруты всех AGV, прогнозирует точки пересечения и определяет приоритеты проезда, исключая столкновения и заторы.

Для интеграции с инфраструктурой предприятия WCS поддерживает подключение к внешним системам управления (WMS, ERP). Это позволяет автоматически получать производственные задания и передавать данные о выполнении операций. Совместимость с различным ПО обеспечивается за счет стандартных протоколов обмена данными, таких как REST API.

Для подключения к оборудованию разрабатываются индивидуальные интеграционные модули.

Дополнительные возможности WCS включают инструменты мониторинга и диагностики. Операторы могут отслеживать местоположение роботов, статус оборудования и текущие задачи в режиме реального времени. Система также автоматически обнаруживает неисправности, уведомляет персонал и инициирует процедуры обслуживания.

Состав ПО

Система представляет из себя набор связанных микросервисов, часть из которых stateless, часть хранит данные во внешнем хранилище данных.

В базовый набор компонент входят:

- errorsstorage
- wms-facade
- fleet-server
- kartograph
- pastuh
- rms-facade
- robot-manager
- react-app
- task-manager
- telemetry
- traffic-control
- zavgar

В качестве хранилищ используются:

- PostgreSQL
- Amazon S3 совместимое хранилище для файлов (MinIO, Yandex cloud service или другое)
- Elasticsearch

Вспомогательные компоненты:

- Keycloak
- Kafka
- Zookeeper
- Kafka-UI
- Logstash
- Kibana
- NginX

Функциональная спецификация базовых компонент

errorsstorage

Компонента предназначена для хранения информации о произошедших ошибках у AGV роботов и оборудования.

Функции:

- Получение описания ошибки
- Запись/обновление описание ошибки
- Проверка состояния

Зависимости:

- PostgreSQL
 - Используется для хранения информации

wms-facade

Компонента для интеграции с WMS (warehouse management system)

Функции:

- Методы для работы с задачами
- Методы для работы со статусами задач
- Методы для работы с причинами отмены задач
- Реализация AsyncAPI, если взаимодействие с WMS идёт через внешнего брокера сообщений.

Бизнес-логика:

- Получает задачи (и команды по задачам) из внешней системы, проводит их первичную валидацию и передаёт в task-manager.
- Получает в системе информацию об изменении статуса задачи и оповещает WMS об изменениях.
- Принимает запросы из WMS на получение информации по задачам собирает данные из системы для ответа.

Зависимости:

- Kartograph
 - Получение данных о карте склада
- task-manager
 - Получение данных о задачах и их статусах
 - Передача задач
- Kafka
 - Получение событий об изменении статусов задач

fleet-server

Компонента для общения с AGV роботами и их зарядными станциями. Реализована в виде websocket сервера.

Функции:

- Авторизация AGV роботов
- Передача в систему состояния робота
- Передача роботу миссий на выполнение
- Передача команд зарядным станциям роботов

Бизнес-логика:

- Контроль установленных соединений с роботами

Зависимости:

- Kafka
 - Получение команд для роботов
 - Передача состояния роботов

kartograph

Компонента для работы с картой.

Функции:

- Сервис для работы с местами парковки AGV роботов
- Сервис управления свойствами точек.
- Сервис управления путями и расчета расстояний.
- Сервис управления картами
- Сервис управления характеристиками объектов карты

Бизнес-логика:

Компонента реализует все необходимые другим компонентам функции для работы с картой.

Для пользователя предоставляет возможность настраивать объекты карты и обновлять карту.

Зависимости:

- Kafka
 - Уведомления об изменениях карты
- PostgreSQL
 - Хранение данных по объектам карты
- MinIO
 - Хранение файлов карты

pastuh

Компонента выполняет функцию диспетчеризации оборудования и AGV роботов (исполнителей) и контролирует выполнение бизнес-процессов по полученным задачам

Функции:

- Основные (по задачам)
 - Декомпозиция задач на миссии
 - Выбор порядка выполнения задач и назначение исполнителей
- Дополнительные (по правилам)
 - Контроль соответствия состояния исполнителей бизнес-правилам склада
- Коммуникационные
 - Получение информации о задачах, требующих выполнения
 - Передача результатов выполнения задач
 - Передача миссий на выполнение
 - Получение статуса выполнения миссий
 - Получение состояния исполнителей
- Технические
 - Сохранение и восстановление плана выполнения задач исполнителями при аварийном завершении работы сервиса

Бизнес-логика:

- Оптимизация выполнения задач с целью уменьшить общее время их выполнения и пробег AGV роботов
- Выбор порядка выполнения задач с учетом топологии склада
- Обеспечение стоянки AGV роботов в отведенных для этого местах
- Обеспечение зарядки AGV роботов для поддержания оптимального уровня заряда и оптимального количества доступных к выполнению задач AGV роботов

Зависимости:

- Kafka
 - Получение и передача всех событий и команд
- PostgreSQL
 - Хранение состояния компоненты для восстановления
- Kartograph
 - Работа с картой

rms-facade

Компонента является основным бэкэндом для web-интерфейса системы. Реализует синхронные (REST API) методы и web-socket сервер для получения на фронте данных в режиме реального времени. Кеширует информацию для оптимизации запросов.

Функции:

- API

- Управление задачами: создание, получение, фильтрация, отмена, повтор, продолжение, получение истории и состояний задач.
- Управление причинами отмены: добавление, получение, удаление причин отмены задач.
- Управление роботами: получение списка, установка статуса, получение истории состояний и статистики пробега.
- Управление точками: поиск и фильтрация точек.
- Получение справочной информации: доступные команды и статусы задач, уникальные имена задач, статистика по отменам и результатам состояний.
- WebSocket (сервер отправляет клиенту):
 - Текущее состояние всех роботов
 - Актуальные данные по задачам
 - Сводную статистику по задачам
 - Данные о карте
 - Обновления по состоянию роботов
 - Обновления по задачам
 - Обновления по групповой статистике роботов.
 - Историю состояний роботов за период
- WebSocket (сервер принимает команды от клиента)
 - Запрос актуальных данных по задачам.
 - Запрос сводной статистики.
 - Запрос актуальных задач.
 - Запрос данных о карте.
 - Запрос истории состояний роботов за период

Бизнес-логика:

- Валидация сообщений от клиента
- Фильтрация данных по параметрам
- Широковещательная отправка обновлений всем подключенным клиентам
- Преобразование исторических данных в формат, позволяющий просматривать изменения в пользовательском интерфейсе

Зависимости:

- Kafka
 - Получение событий в реальном времени
- Telemetry
 - Получение исторической информации по оборудованию и задачам
- Kartograph
 - Работа с картой
- Robot-manager
 - Получение данных по AGV роботам
- Task-manager
 - Получение данных по задачам
- Pastuh

- Функции администратора

robot-manager

Компонента отвечает за обработку входящих статусов оборудования и актуализацию данных о них в базе, формирование и отправку сообщений об ошибках оборудования, поддержку учёта производительности роботов (пробег, рабочее время) и отправку агрегированных данных по производительности.

Функции:

- Сервис управления роботами
 - CRUD-операции с роботами
 - Получение списка роботов
 - Установка статуса робота
 - CRUD-операции с типами роботов
 - Получение списка типов роботов
 - Массовое обновление карты у группы роботов
- Сервис управления зарядными станциями
 - CRUD-операции с зарядными станциями
 - Получение списка зарядных станций с фильтрацией и пагинацией
 - Установка статуса зарядной станции
 - CRUD-операции с типами зарядных станций
 - Получение списка типов зарядных станций

Бизнес-логика:

При обработке статусов ведёт статистику производительности оборудования. Отправляет её в телеметрию

Следит за статусами оборудования и формирует сообщения об ошибках на них.

Зависимости:

- PostgreSQL
 - Хранение данных
- Kafka
 - Получение статусов оборудования
 - Отправка значений производительности и информации об ошибках

react-app

Пользовательский веб интерфейс системы

Функции:

- Главная страница
 - Отображение карты
 - Отображение оборудования и их статусов в реальном времени
 - Отображение актуальных задач и их статусов в реальном времени
 - Фильтрация объектов
 - Отображение оперативной статистики

- Базовые функции управления оборудованием и задачами
- Список роботов
 - Отображение роботов в виде таблицы с описанием их статуса
 - Фильтры и поиск
- Страница робота
 - Детальная информация по роботу
 - Контроллеры управления роботом
- Список задач
 - Отображение задач в виде таблицы с описанием их параметров и статуса выполнения
 - Фильтры и поиск
- Страница задачи
 - Детальная информация по задаче
 - Отображение истории изменения статуса и исполнителей по задаче
 - Контроллеры управления задачами

Зависимости:

- Nginx
 - Web-сервер
- rms-facade
 - бэкэнд для всей отображаемой информации и методов взаимодействия с системой
- MinIO
 - Получение видео ошибок роботов и файлов конфигурации роботов

task-manager

Компонента предназначена для управления задачами (task manager) с поддержкой различных состояний задач, причин отмены, интеграции с Kafka, аутентификации, пагинации, логирования и взаимодействия через gRPC и REST API.

Функции:

- Сервис управления задачами
- Сервис управления состояниями
- Сервис управления причинами отмены
- Сервис статистики

Бизнес-логика:

При получении задач, проводит их валидацию на выполнимость. Определенные типы задач нельзя назначить на конкретного робота, другие можно выдать только свободному роботу, готовому к выполнению.

Зависимости:

- PostgreSQL
 - Хранение данных
- robot-manager

- Получение состояния роботов, зарегистрированных в системе
- Kafka
 - Чтение и передача информации по событиям

telemetry

Компонента для сбора и хранения информации о процессе работы оборудования. Хранит состояния задач, состояния оборудования, маршруты перемещения AGV роботов и другую информацию для анализа процессов.

Функции:

- Сервис получения истории
 - GetRobotHistory
 - GetExtendedHistory

Бизнес-логика:

Собирает из событий состояние оборудования на интервал времени, заполняя полное состояние для его границ и события изменения состояний для каждого экземпляра.

Зависимости:

- PostgreSQL
 - Хранение данных
- Kafka
 - Получение сообщений о состояниях объектов

traffic-control

Компонента контроля трафика AGV роботов. Получает данные о габаритах робота, его маршруте. Ищет транспортные коллизии и решает их через команды установки ограничения скорости вплоть до остановки. Реализована в виде websocket сервера.

Функции:

- Регистрация AGV роботов
- Получение маршрутов
- Получение статуса непредвиденной остановки (ручной режим, препятствие на пути)
- Отправка команд на изменение скорости
- Отправка точек планируемых точек остановки далее по маршруту

Бизнес-логика:

Оптимизирует простои, в приоритете замедляя того AGV робота, который находится дальше от коллизии

Если система стремится к состоянию, в котором AGV роботы начинают блокировать друг друга циклически, разрывает этот цикл через смену приоритета проезда одной или нескольких коллизий

Зависимости: отсутствуют

zavgar

Компонента для выдачи технических задач AGV роботам, связанных с процессами сервиса: ТО и Ремонта.

Функции:

- Сервис управления техническими задачами
- Сервис управления ошибками роботов
- Сервис управления плановым техобслуживанием (событиями)
- Сервис управления типами техобслуживания

Бизнес-логика:

Отслеживание статистики по AGV роботам и принятие решения о необходимости прохождения автоматического (калибровка) или ручного (ремонт и замена деталей) обслуживания.

Может запретить выход робота на выполнения бизнес-задач при наличии повторяющихся ошибок выполнения.

Зависимости:

- robot-manager
 - Получение и синхронизация информации о AGV роботах
- Kafka
 - Получение состояний роботов
 - Отправка технических задач
 - Получение статусов выполнения технических задач
- PostgreSQL
 - Хранение данных

Функциональная спецификация хранилищ

1. PostgreSQL

Требуется в стандартной конфигурации.

Основная задача – хранение данных компонент.

2. Amazon S3 совместимое хранилище.

Требуется в стандартной конфигурации. Рекомендуется использовать MinIO или сервис Yandex Cloud.

Хранит файлы карт всех версий, которые применялись в системе.

Хранит видеофайлы ошибок роботов, которые записываются самими роботами напрямую для последующей демонстрации пользователю в web-интерфейсе.

Хранит файлы конфигурации роботов для разбора ошибок

3. Elasticsearch

Необязательный компонент.

Используется в стандартной конфигурации с logstash и kibana. Используется для хранения логов всех компонент системы и роботов. При отсутствии, логи хранятся в файловой системе.

Функциональная спецификация вспомогательных компонент

Keycloak – система управления доступом. В стандартной настройке не используется, т.к. система WCS работает в закрытых сетях.

Kafka – распределенный программный брокер сообщений. Используется по прямому назначению для связи компонент между собой. Зависит от zookeeper.

Kafka-UI – используется для просмотра сообщений в Kafka при отладке и диагностике системы.

Logstash, Kibana используются совместно с Elasticsearch для организации данных логирования и анализа логов при отладке и диагностике системы.

NginX – основной web-server системы, используется для реализации пользовательского интерфейса.

Прочие функциональные особенности

Запуск и завершение работы системы

Automason WCS является серверным программным обеспечением. Периодический запуск и завершение работы системы в пользовательском интерфейсе не предусмотрено, т.к. подразумевается непрерывная работа.

Запуск осуществляется силами системного администратора с использованием команды

```
docker compose up -d
```

Завершение работы компонентов системы производится через команды `docker`, в частности

```
docker compose down
```

Обновление

У системы отсутствует функция автоматической проверки обновления и их установки. Обновление производится силами технической поддержки по запросу.

Резервное копирование и восстановление

У Automason WCS отсутствуют встроенные функции по резервному копированию и восстановлению системы. В инфраструктуре заказчика системы могут использоваться любые системы предусмотренные для этого:

- Для файлов настроек – системы резервного копирования
- Для баз данных – системы репликации и/или резервного копирования, предусмотренные СУБД
- Для образов системы – резервное копирование собственный `docker-registry`, образы можно получить повторно у компании разработчика.