

УТВЕРЖДАЮ

Генеральный директор

ООО «КОМИТАС»

/ Л.А. Григорян

04.11.2024



ПРОГРАММА ДЛЯ УПРАВЛЕНИЯ ЗАРЯДНОЙ СТАНЦИЕЙ
CSSCOMITAS (CHARGE STATION SOFTWARE COMITAS)

ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК

ЛИСТ УТВЕРЖДЕНИЯ

РОФ.ЛТПГ.00009-01 90 ЛУ

СОГЛАСОВАНО

Руководитель группы разработки
робототехнических систем

 П.В. Икомасов

«04» 11 2024 г.

Директор по инновациям,
научный руководитель НИОКР

 М.Н. Кочетков

«11» 11 2024 г.

Име. № подл.	Подпись и дата	Взам. име. №	Име. № дубл.	Подпись и дата

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ «КОМИТАС»

УТВЕРЖДЕНО

РОФ.ЛТПГ.00009-01 90 ЛУ

ПРОГРАММА ДЛЯ УПРАВЛЕНИЯ ЗАРЯДНОЙ СТАНЦИЕЙ
CSSCOMITAS (CHARGE STATION SOFTWARE COMITAS)

ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК

РОФ.ЛТПГ.00009-01 90

Москва, 2024г.

Име. №подл.	Подпись и дата
Взам. име. №	Име. №дубл.
Подпись и дата	Подпись и дата

АННОТАЦИЯ

Документ «Описание функциональных характеристик» предназначен для сотрудников эксплуатирующей организации и содержит перечень функциональных возможностей Программного обеспечения для управления зарядной станцией (CSSCOMITAS).

Документ актуален для указанной версии и для последующих версий вплоть до выпуска обновления описания функциональных характеристик.

Выход новой версии программного обеспечения сопровождается обновлением данного документа только в случае наличия в версии значительных изменений в перечне функциональных возможностей Программного обеспечения. Если таких изменений версия не содержит, то остается актуальным описание функциональных характеристик от предыдущей версии с учетом изменений, содержащихся в новой версии.

СОДЕРЖАНИЕ

АННОТАЦИЯ	3
СОДЕРЖАНИЕ	4
1 ВВЕДЕНИЕ	5
2 ФУНКЦИОНАЛЬНЫЕ ХАРАКТЕРИСТИКИ	6
2.1 Цели и назначение	6
2.2 Ключевые функции	6
3 ИНФОРМАЦИЯ НЕОБХОДИМАЯ ДЛЯ УСТАНОВКИ И ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	8
3.1 Системные требования	8
3.2 Требования к квалификации персонала.....	8
4 УСТАНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЕСАСОМІТАS	9
4.1 Клонирование репозитория ЕСАСОМІТАS	9
4.2 Необходимое ПО	9
4.3 Подготовка к запуску.....	9
4.4 Установка программного обеспечения на промышленный компьютер зарядной станции.....	10
4.5 Технические особенности	11
5 УСТАНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СSSCOMІТАS	12
5.1 Клонирование репозитория СSSCOMІТАS	12
5.2 Необходимое ПО	12
5.3 Установка программного обеспечения на промышленный компьютер зарядной станции.....	12
5.4 Технические особенности	12
6 ТЕХНИЧЕСКАЯ ПОДДЕРЖКА	14

1 ВВЕДЕНИЕ

Настоящий документ предназначен для сотрудников эксплуатирующей организации, в должностные обязанности которых входят функции использования или администрированию программы для управления зарядной станцией CSSCOMITAS.

2 ФУНКЦИОНАЛЬНЫЕ ХАРАКТЕРИСТИКИ

2.1 Цели и назначение

Программное обеспечение (ПО) для управления зарядным процессом роботов предназначено для автоматизации процессов, направленных на своевременное и быстрое восполнение заряда роботов сортировщиков.

ПО зарядной станции функционально интегрировано с ПО RMSCOMITAS, которое формирует очередь задач для отправки и дальнейшей обработки в ПО зарядной станции.

2.2 Ключевые функции

Внутреннее программное обеспечение зарядной станции позволяет обеспечить следующие основные функции:

- Управление зарядом роботов:

Встроенное ПО обеспечивает безопасность при заряде и возможность применять различные профили как к одному типу роботов, так и к широкому спектру их номенклатуры.

- Алгоритмы обеспечения безопасности:

Встроенное ПО постоянно отслеживает состояние оборудования, как во время простоя, так и во время работы по заряду и своевременно сигнализирует о неисправностях. При возникновении аварийных ситуаций, ПО автоматически отключит неисправное оборудование и прекратит работу.

- Взаимодействие со смежными системами:

Встроенное ПО зарядной станции обеспечивает возможность коммуникации и взаимодействия с системой RMS.

- Диагностические утилиты:

Встроенное ПО зарядной станции предоставляет инструменты для диагностики текущего состояния оборудования и статусов работы. К примеру, при возникновении проблем или неисправностей, станция может дать световой сигнал, а также отправить информацию о проблеме оператору либо на сервер управления.

- Сбор, запись и хранение информации (логирование) о процессе функционирования:

Встроенное ПО зарядной станции осуществляет постоянное логирование действий зарядной станции, что позволяет осуществлять отладку внутренних процессов, а также оперативно детектировать возможные проблемы. Помимо этого, благодаря логированию может быть осуществлен сбор внутренних метрик, позволяющий определить эффективность бизнес-процессов.

- Удаленный доступ и управление

Встроенное ПО зарядной станции предоставляет пользователю и оператору различные средства удаленного доступа и контроля. Это может быть применено для обновления программного обеспечения, настройки параметров работы, управления зарядом вручную, добавления или удаления профилей заряда.

3 ИНФОРМАЦИЯ НЕОБХОДИМАЯ ДЛЯ УСТАНОВКИ И ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

3.1 Системные требования

Программное обеспечение устанавливается непосредственно на промышленный компьютер, интегрированный в бортовую систему зарядной станции, и представляет собой неотъемлемый компонент комплектации данных робототехнических систем.

Системные требования промышленного компьютера:

- Оперативная память (RAM): 2 ГБ;
- Накопитель (SSD): 256 ГБ;
- Процессор: Intel Atom E3930 (1 ГГц, 2 ядра).

3.2 Требования к квалификации персонала

Для работы с программным обеспечением необходимо разделение пользователей на:

- Оператор:

Должен обладать компетенциями работы с автономными мобильными роботами и их зарядными устройствами. Должен курировать операционные процессы сортировки.

- Инженер информационных систем:

Должен отвечать за взаимодействие зарядного устройства с внешними системами типа RMSCOMITAS, а также решать возможные проблемы в системах связи либо в системах интеграции.

- Администратор:

Обладает максимальным уровнем доступа к ПО робота, имеет возможность осуществлять корректировку информации и расширенные настройки зарядной станции (при взаимодействии с сервисной службой).

4 УСТАНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЕСАСОМІТАS

4.1 Клонирование репозитория ЕСАСОМІТАS

Программное обеспечение применяет в своей работе дополнительные репозитории, поэтому для клонирования репозитория рекомендуется использовать ключ `--recursive`.

Для начала копирования введите следующую команду:

```
git clone --recursive https://git.comitas.ru/robotics/agv-amr/software/eca.
```

4.2 Необходимое ПО

Для успешной предварительной сборки и запуска проекта требуется установить ряд утилит:

- CMake версии 3.22 или выше;
- GCC/G++ версии 11 или выше;
- Node.js версии 18.18.2 (использование других версий может привести к проблемам при генерации сообщений).

4.3 Подготовка к запуску

Для запуска сервера и пользовательского интерфейса следует воспользоваться скриптом, который находится в ранее загруженном репозитории: `run-vue.sh`. Данный скрипт автоматически пересобирает проекты и запускает сервер.

Также следует обратить внимание, что для запуска сервера необходимы права суперпользователя, так как сервер может перенастраивать сетевые параметры, взаимодействовать с аппаратным обеспечением напрямую и использует для своей работы 80-й порт.

Если же необходимо запустить `ipc_proto_proxy-software` без прав суперпользователя, следует установить переменную окружения `NO_SUDO` в значение `true` следующим образом:

```
NO_SUDO=true ./run_vue.sh
```

Перед запуском также может потребоваться внести изменения в конфигурационный файл `irc_proto_proxy-software/config/config.toml` и запустить дополнительные программы.

4.4 Установка программного обеспечения на промышленный компьютер зарядной станции

Установка на промышленный компьютер робота осуществляется с помощью скрипта `deploy-eca.sh`, который предварительно собирает компоненты `eca` и `front-vue`.

При запуске скрипт запрашивает пароль пользователя с именем `robot` для подключения по протоколу SSH. Для корректной работы скрипта необходимо наличие установленного программного обеспечения Ansible. Установить необходимую версию Ansible можно следующим образом:

```
python3 -m venv venv
. venv/bin/activate
python3 -m pip install -r requirements.txt
```

```
python3 -m venv venv
. venv/bin/activate
python3 -m pip install -r requirements.txt
```

В дальнейшем запускать скрипт следует так:

```
. venv/bin/activate
./deploy-eca.sh
```

```
. venv/bin/activate
./deploy-eca.sh
```

Для выполнения установки ECACOMITAS из системы непрерывной интеграции (CI) можно использовать задачу с именем `deploy`.

4.5 Технические особенности

Служба eca управляется с помощью systemd, что обеспечивает удобство в её контроле. В случае необходимости остановить сервис, на работе следует выполнить команду:

```
sudo systemctl stop eca
```

Если потребуется перезапустить службу, используется команда:

```
sudo systemctl restart eca
```

Для просмотра архива событий сервиса достаточно ввести команду:

```
journalctl -u eca
```

А если необходимо следить за списком событий в режиме "реального времени", используется команда:

```
journalctl -f -u eca
```

Стоит отметить, что сервис автоматически запускается при старте промышленного компьютера. Если автоматический запуск нежелателен, можно его отключить, выполнив следующие команды:

```
sudo systemctl stop eca
```

```
sudo systemctl disable eca
```

Все необходимые файлы для запуска eca, включая бинарный файл, располагаются на работе по адресу репозитория:

```
/opt/comitas/eca.
```

5 УСТАНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ CSSCOMITAS

5.1 Клонирование репозитория CSSCOMITAS

Программное обеспечение применяет в своей работе дополнительные репозитории, поэтому для клонирования репозитория рекомендуется использовать ключ `--recursive`.

Для начала копирования введите следующую команду:

```
git clone --recursive https://git.comitas.ru/robotics/agv-  
amr/chargers/charger_software.git.
```

5.2 Необходимое ПО

Для успешной предварительной сборки и запуска проекта требуется установить ряд утилит:

- CMake версии 3.22 или выше;
- GCC/G++ версии 11 или выше;

5.3 Установка программного обеспечения на промышленный компьютер зарядной станции

Установка на промышленный компьютер робота осуществляется с помощью скрипта `deploy-charger.sh`, который предварительно собирает программное обеспечение CSSCOMITAS.

Для корректной работы скрипта необходимо наличие установленного программного обеспечения Ansible.

Запуск скрипта осуществляется аналогично пункту 2.4

5.4 Технические особенности

Служба `charger` управляется с помощью `systemd`, что обеспечивает удобство в её контроле. В случае необходимости остановить сервис, на работе следует выполнить команду:

```
sudo systemctl stop charger
```

Если потребуется перезапустить службу, используется команда:

```
sudo systemctl restart charger
```

Для просмотра архива событий сервиса достаточно ввести команду:

```
journalctl -u charger
```

А если необходимо следить за списком событий в режиме "реального времени", используется команда:

```
journalctl -f -u charger
```

Стоит отметить, что сервис автоматически запускается при старте промышленного компьютера. Если автоматический запуск нежелателен, можно его отключить, выполнив следующие команды:

```
sudo systemctl stop charger
```

```
sudo systemctl disable charger
```

Все необходимые файлы для запуска charger, включая бинарный файл, располагаются на роботе по адресу репозитория:

```
/opt/comitas/ charger.
```

6 ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Вопросы возникающие в ходе работы с программным обеспечением следует направлять в службу поддержки по адресу info@comitas.ru.

Все обращения рассматриваются в рабочее время (Европе GMT+3), ответы и оказание поддержки в штатном режиме предоставляются не позднее 1 часа с момента обращения.

СОСТАВИЛИ

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпи сь	Дата

СОГЛАСОВАНО

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпи сь	Дата