

СИСТЕМНЫЕ ТРЕБОВАНИЯ

Развертывание на GPU

8+ vCore  
32+ Гб RAM  
SSD 100 Гб свободного места  
Требования к видеокарте:  
24+ Гб VRAM  
Compute Capability 8.0+  
Linux с поддержкой Docker 22+ (совместимость проверена с RedHat, Debian, Ubuntu, CentOS)  
NVIDIA Docker Toolkit (nvidia-docker2) или NVIDIA CUDA Toolkit

Развертывание на CPU (не рекомендуется)

12+ vCore (рекомендуется 48+ vCore)  
64+ Гб RAM  
SSD 100 Гб свободного места  
Требования к видеокарте:  
24+ Гб VRAM  
Compute Capability 8.0+  
Linux с поддержкой Docker 22+ (совместимость проверена с RedHat, Debian, Ubuntu, CentOS)

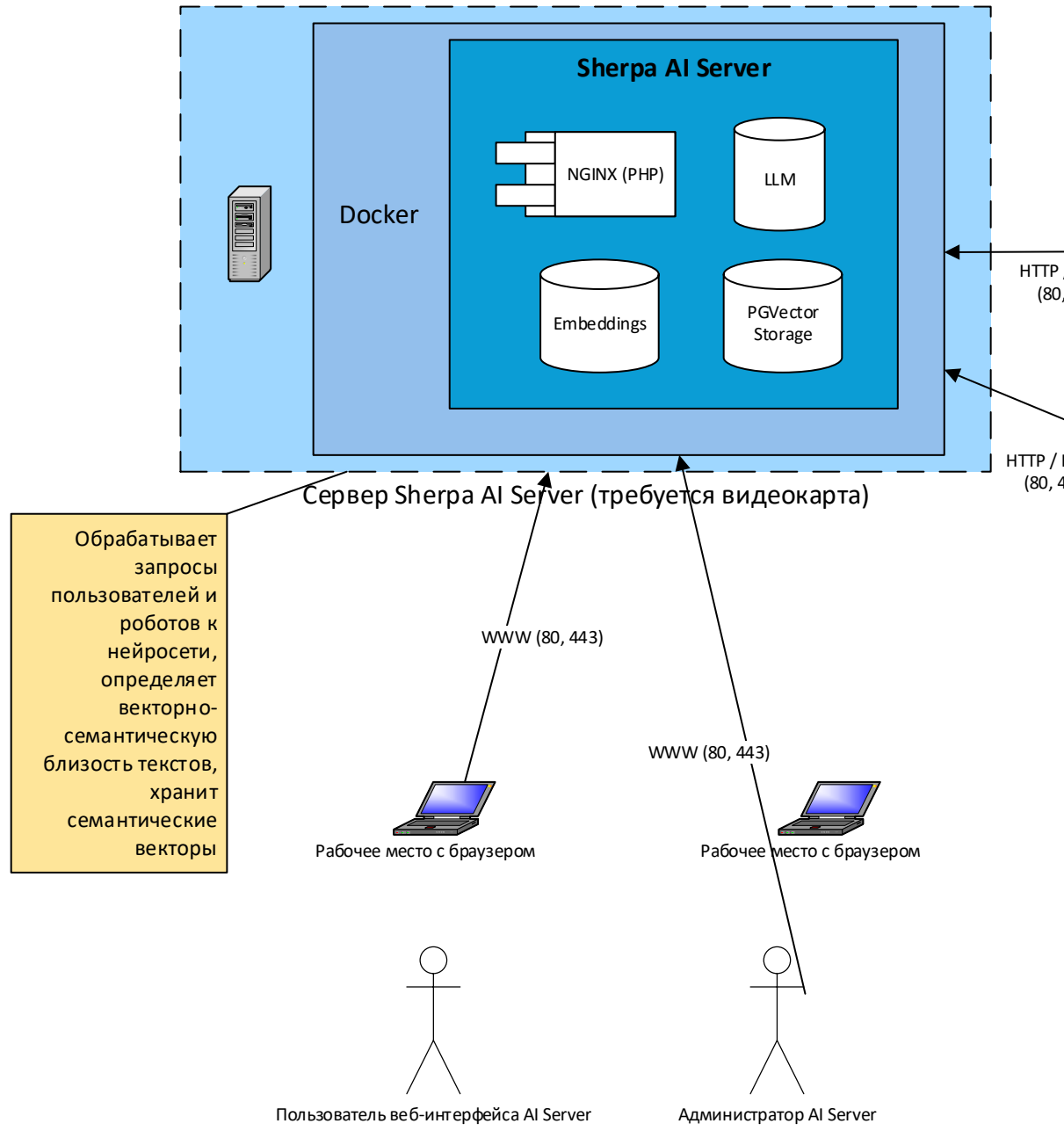
Совместимые модели видеокарт

**NVIDIA Data Center Products:** NVIDIA A2, NVIDIA A16, NVIDIA A10, NVIDIA A30, NVIDIA A40, NVIDIA A100, NVIDIA L4, NVIDIA L40, NVIDIA H100.  
**NVIDIA RTX Desktop:** RTX A4000, RTX A5000, RTX A6000, RTX 6000.  
**NVIDIA RTX Mobile:** RTX A2000, RTX A3000, RTX A4000, RTX A5000.  
**GeForce Products:** GeForce RTX 3060, GeForce RTX 3060 Ti, GeForce RTX 3070, GeForce RTX 3070 Ti, GeForce RTX 3080, GeForce RTX 3080 Ti, GeForce RTX 3090, GeForce RTX 3090 Ti, GeForce RTX 4070 Ti, GeForce RTX 4080, GeForce RTX 4090  
**GeForce Notebook Products:** GeForce RTX 3050, GeForce RTX 3050 Ti, GeForce RTX 3060, GeForce RTX 3060 Ti, GeForce RTX 3070, GeForce RTX 3070 Ti, GeForce RTX 3080, GeForce RTX 3080 Ti, GeForce RTX 4050, GeForce RTX 4060, GeForce RTX 4070, GeForce RTX 4080, GeForce RTX 4090  
**Jetson Products:** Jetson AGX Orin, Jetson Orin NX, Jetson Orin Nano

ТЕХНОЛОГИЧЕСКИЙ СТЕК (СПРАВОЧНО)

NGINX / PHP / Python/ C++ / Angular  
PostgreSQL + PGVectors  
Docker

Диаграмма развертывания Sherpa AI Server в связке с роботами



ОПИСАНИЕ:

Расширение схемы развертывания платформы Sherpa RPA описывает использование компонента Sherpa AI Server в составе платформы роботизации для реализации сценариев с использованием искусственного интеллекта и больших языковых моделей.

Диаграмма развертывания:

Все компоненты Sherpa AI Server устанавливаются локально в сети Заказчика, без связи с внешними серверами или службами SaaS. Развертывание Sherpa AI Server осуществляется с помощью Docker-контейнера.

Состав решения:

- Веб-интерфейс пользователя для непосредственного общения с большой языковой моделью
- Векторно-семантическое хранилище
- Инференс-сервер большой языковой модели
- Эмбединг-сервер большой языковой модели

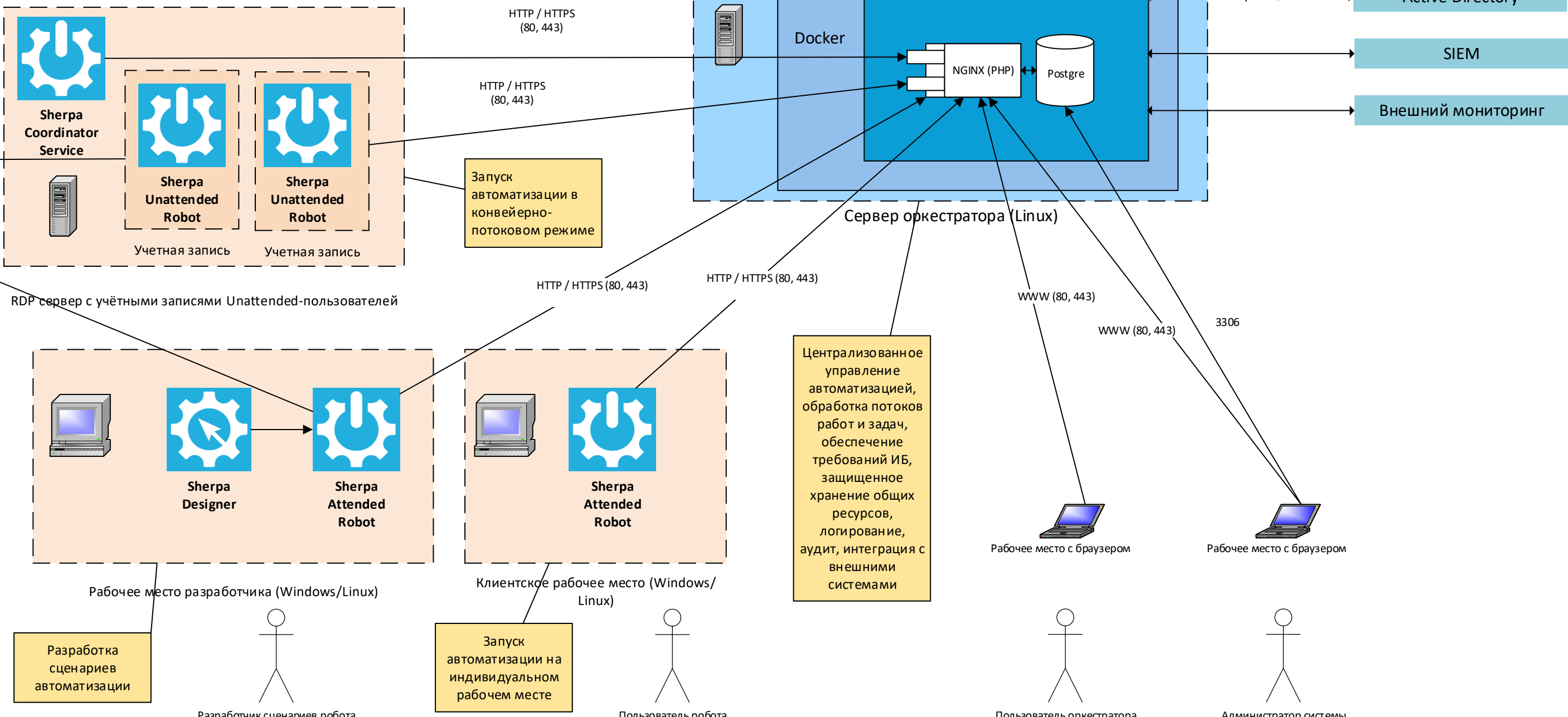
Конфигурация сети:

Конфигурация портов и сетевые протоколы Sherpa Sherpa AI Server могут быть настроены для поддержки всех общих требований брандмауэра. Для взаимодействия с веб-сервером используется https, опционально возможно http. При установке с помощью TLS клиент должен предоставить необходимые сертификаты, разместив их по пути /opt/app/config/certs/, переименовав их в aiserver.crt и aiserver.key.

Механизмы аутентификации

Аутентификация веб-пользователей в Sherpa AI Server производится с помощью логина и пароля. Аутентификация внешних приложений, включая роботов платформы Sherpa RPA, производится с помощью API-токена.

Диаграмма развертывания платформы Sherpa RPA под управлением оркестратора



ОПИСАНИЕ:

Схема описывает верхнеуровневую реализацию функционала, который обеспечивает автоматизацию бизнес-процессов, выполняемых на пользовательских рабочих местах и терминальных серверах путём роботизации (имитации действий пользователя при работе с информационными системами). Разработчики сценариев роботизации после тестирования передают сценарии в централизованное средство управления – оркестратор. Бизнес-пользователи на своих рабочих местах по требованию запускают attended-сценарии автоматизации (требующие интерактивного взаимодействия робота или бизнес-процесса с пользователем), запуск осуществляется путём выбора сценария из списка установленных на рабочем месте, нажатия горячей клавиши, также возможен запуск сценариев из локального планировщика Windows и в автозагрузке. Оркестратор по событиям или входящим данным из внешней среды (через API), по команде робота или бизнес-пользователя или по расписанию запускает unattended-сценарии роботизации (выполняемые в учётных записях терминальных серверов без участия пользователя). Роботы обоих типов обмениваются с оркестратором логami, задачами из очереди, централизованно хранимыми учётными данными и общими данными. В процессе исполнения сценария робот достигает поставленных задач путём имитации действий пользователя (включая действия с мышью и клавиатурой) в пользовательских интерфейсах десктоп-приложений, веб-приложений (исполняемых в браузере), а также путём взаимодействия с операционной системой и установленными приложениями и веб-сервисами с помощью разнообразных программных API. Уровень доступа робота к информационным системам и контурам клиента ограничивается и определяется уровнем доступа того пользователя, под учётной записью которого робот производит действия, описанные в сценарии. Для unattended-роботов рекомендуется создавать выделенные учётные с уровнями доступа, правами и ролями, определяемыми решаемой данным роботом бизнес-задачей.

Диаграмма развертывания:

Все компоненты Sherpa RPA устанавливаются локально в сети Заказчика, без связи с внешними серверами или службами SaaS. Возможность и необходимость доступа компонентов платформы к внутренним и внешним системам определяется решаемой в рамках бизнес-процесса задачей. Развертывание рабочего места разработчика, рабочего места attended и unattended-роботов производится вручную с помощью с соответствующих exe-инсталляторов либо автоматически с помощью msi-инсталлятора с использованием механизма GPO. Установка Sherpa Coordinator Service производится в администраторской учётной записи терминального сервера с помощью exe-инсталлятора. По умолчанию развертывание Sherpa Orchestrator осуществляется с помощью Docker-контейнера.

Последовательность шагов процесса:

- Разработчик сценария роботизации бизнес-процесса с помощью десктопного ПО Sherpa Designer и локального Sherpa Robot создаёт и отлаживает сценарий роботизации. Готовые сценарии роботизации передаются на рабочие места пользователей или роботов вручную путём копирования конфигурационного файла робота либо с помощью функции удаленной публикации новой версии сценария из Sherpa Designer в Sherpa Orchestrator.
- Серверный компонент Sherpa Orchestrator поддерживает связь с роботами, запущенными на клиентских машинах, хранит конфигурации роботов и версии сценариев, общие глобальные переменные и учетные данные, логи и скриншоты работы роботов, журналы аудита оркестратора, пользователей, роли и tenants самого оркестратора, лицензии всех компонентов платформы и статистику исполнения сценариев.
- Пользователи и администраторы Sherpa Orchestrator получают доступ к ресурсам, настройкам и статистике с помощью веб-приложения оркестратора, доступного через веб-браузер. Sherpa Orchestrator включает в себя веб-сервер Nginx, интерпретатор PHP и реляционную базу данных (по умолчанию - MariaDB).
- Локальные пользователи по необходимости запускают Sherpa Attended Robot на исполнение хранящихся локально сценариев. В процессе исполнения сценария Sherpa Attended Robot может передавать оркестратору текущий статус, задачи, логи, значения глобальных переменных и учетных данных или получать от него задачи, значения глобальных переменных и учетных данных.
- По расписанию, вызову API, команде робота либо по другому поддерживаемому триггеру Sherpa Orchestrator даёт задание Sherpa Coordinator Service соответствующего терминального сервера создать RDP-подключение к локальной или удаленной учётной записи, выделенной для unattended-робота, при этом на одном терминальном сервере может быть размещено и одновременно активно несколько таких учётных записей. Вход в учётную запись unattended-робота производится с предоставленными оркестратором логином и паролем. После входа в учётной записи запускается соответствующий экземпляр Sherpa Unattended Robot, подключается к оркестратору, получает задание на выполнение сценария и сам сценарий, хранящийся в оркестраторе. В процессе исполнения сценария Sherpa Unattended Robot может передавать оркестратору текущий статус, задачи, логи, значения глобальных переменных и учетных данных или получать от него задачи, значения глобальных переменных и учетных данных, команды для «мягкого» или «жесткого» завершения сценария. После завершения исполнения сценария Sherpa Unattended Robot выполняет logout из своей учётной записи.

Конфигурация сети:

Конфигурация портов и сетевые протоколы Sherpa RPA могут быть настроены для поддержки всех общих требований брандмауэра. Конфигурация порта по умолчанию выглядит следующим образом:  
\*Sherpa Attended Robot, Sherpa Unattended Robot, Sherpa Coordinator исходящие на Sherpa Orchestrator: 80 или 443  
\*Во всех сетевых коммуникациях инициатива установли подключения и первоначального запроса принадлежит только клиентским компонентам, то есть Sherpa Attended Robot, Sherpa Unattended Robot и Sherpa Coordinator. Sherpa Orchestrator по своей инициативе не выполняет запросы к клиентам.  
\*Связь с базой данных: 3306 и 1433-настраивается  
\*Доступ пользователя к веб-интерфейсу Sherpa Orchestrator: 80 или 443

Для взаимодействия с веб-сервером используется https, опционально возможно http. Sherpa RPA поддерживает защищенную связь (с использованием протокола TLS 1.2) между Attended Robot, Sherpa Unattended Robot, Sherpa Coordinator и Sherpa Orchestrator. При установке с помощью TLS клиент должен предоставить необходимые сертификаты, разместив их по пути /opt/app/config/certs/, переименовав их в orchestrator.crt и orchestrator.key.

Механизмы аутентификации

Аутентификация Sherpa Attended Robot, Sherpa Coordinator и Orchestrator осуществляется с помощью Bearer Token, передаваемого в заголовке запросов. Bearer Token сопоставляется с уникальным GUID каждого экземпляра Sherpa Attended Robot, Sherpa Unattended Robot, Sherpa Coordinator. Для аутентификации пользователей веб-интерфейса Orchestrator применяется авторизация с помощью пары логин-пароль. При повторном входе используется сессионная кука, имеющая ограниченный срок жизни.

Логирование

Для логирования используется компонент Monolog. События аудита и системные ошибки сохраняются в выделенную таблицу базы данных.

СИСТЕМНЫЕ ТРЕБОВАНИЯ

Sherpa Orchestrator:

4 vCore  
8 Гб RAM  
SSD 100 Гб свободного места  
Linux с поддержкой Docker 22+ (совместимость проверена с RedHat, Debian, Ubuntu, CentOS)

Sherpa Attended Robot / Sherpa Unattended Robot / Sherpa Coordinator:

2 vCore  
4 Гб RAM  
SSD / HDD 5 Гб свободного места  
Windows 7 – Windows 11 или Windows Server 2012 – 2022 или Linux (Debian, Ubuntu, AstraLinux)  
.NET Framework 4.8+  
PowerShell 5.1+

Sherpa Designer:

4 vCore  
8 Гб RAM  
SSD / HDD 10 Гб свободного места  
Windows 7 – Windows 11 или Windows Server 2012 – 2022  
.NET Framework 4.8+  
PowerShell 5.1+

ТЕХНОЛОГИЧЕСКИЙ СТЕК (СПРАВОЧНО)

Sherpa Orchestrator:

NGINX / PHP / Angular  
MariaDB / PostgreSQL  
Clickhouse (опция)  
Docker

Sherpa Attended Robot / Sherpa Unattended Robot / Sherpa Coordinator:

C# (.NET Framework 4.8+)  
PowerShell 5.1+

Sherpa Designer:

C# (.NET Framework 4.8+)