

DRE AI Call Center Assistant

Руководство по установке

Индекс	ССА-IG
Конфиденциальность	Публичный - L0
Ревизия	1.0
Статус	Согласован

Содержание

1. Аннотация	3
2. Общее описание	4
3. Требования к квалификации установщика	5
4. Минимальные системные требования	6
5. Установка и настройка системы	7
5.1. Создание новой среды	7
5.2. Настройка параметров установки в кластер	7
5.3. Настройка production.yaml	7
6. Развертывание системы	9

1. Аннотация

Документ предназначен для технических специалистов, занимающихся установкой, настройкой и поддержкой системы. Документ рассчитан на инженеров, обладающих специальными навыками и знаниями в области инсталляции программного обеспечения.

Данный документ содержит руководство по установке и первоначальной настройке системы DRE AI Call Center Assistant (далее - Система), а также описание системных требований для компонентов.

2. Общее описание

Сервис DRE AI Call Center Assistant (далее по тексту - DRE AI CCA) представляет собой приложение, предназначенное для автоматизированной обработки обращений клиентов в контактный центр. Приложение обеспечивает взаимодействие с клиентами по текстовым каналам (веб-чат, мессенджеры) и голосовым каналам (VoIP/SIP).

3. Требования к квалификации установщика

Для установки системы сотрудник обязан:

- иметь базовые представления и практические навыки работы с системой оркестрации Kubernetes (<https://kubernetes.io/docs/tutorials/kubernetes-basics/>) и пакетным менеджером Helm.
- иметь навыки работы с ОС семейства Linux, а именно:
 - установка пакетов;
 - создание и настройка сетевых подключений;
 - запуск служб, настройка автозапуска служб.
- иметь базовые представления и практические навыки работы с Git.

4. Минимальные системные требования

Для установки должны быть предварительно выполнены следующие требования:

- Установлен и настроен кластер Kubernetes.
 - Так как развертывание производится в кластере k8s, то необходим config file для доступа к кластеру.
 1. Если пользователь выполнял развертывание Kubernetes самостоятельно, то он сам должен создать config file (см. документацию Kubernetes).
 2. Если Kubernetes был развернут сторонними людьми, то необходимо получить config file у администратора кластера (установкой и администрированием системы Kubernetes занимаются сотрудники "Отдела поддержки инфраструктуры").
- Установлен kubectl (<https://kubernetes.io/docs/tasks/tools/install-kubectl/>).
- Установлен helm (<https://helm.sh/ru/docs/intro/install/>). Helm необходим в случае ручной установки элементов системы в кластер k8s. При стандартной установке с использованием gitlab-ci использование helm не предполагается.
- Для корректной работы Системы необходим доступ к следующим ресурсам:
 - chartmuseum (ссылка предоставляется по запросу заказчика),
 - gitlab (ссылка предоставляется по запросу заказчика).
- Необходим доступ к репозиторию (ссылка предоставляется по запросу заказчика), содержащему helmfile для развертывания Системы. Helm файл содержит инструкции, с помощью которых осуществляются настройки устанавливаемых компонентов. Сами компоненты поставляются в виде чартов (charts), из которых развертываются Kubernetes Pods. Данные берутся из gitlab.

5. Установка и настройка системы

Необходимо выполнить установку системы DRE AI Call Center Assistant как описано ниже.

5.1. Создание новой среды

Описание процесса создания и настройки описано в руководстве, в разделе "Описание работы с репозиторием"

Основные этапы создания среды следующие:

1. Создать отдельный проект в Gitlab. Необходимо использовать собственное название репозитория для продукта DRE AI Call Center Assistant.
2. Клонировать имеющийся проект для его локального редактирования.
3. Включить в клонированный репозиторий релизную версию конфигурации продукта DRE AI Call Center Assistant путем добавления субмодуля.
4. Создать файл `production.yaml` для последующего переопределения значений из файла `default.yaml`.
5. Создать `helmfile.yaml`.
6. Создать файл `gitlab-ci.yml` со стандартным содержимым.
7. Проверить конфигурацию продукта.
8. Синхронизировать локальный и удаленный репозитории.

5.2. Настройка параметров установки в кластер

См. руководство Gitlab, раздел "Gitlab runner"

Основные этапы создания среды следующие:

1. Подключить Gitlab runner.
2. Указать необходимое значение `kube_config`.

5.3. Настройка `production.yaml`

В файле `production.yaml` нужно настроить параметры, которые используются в промышленной эксплуатации в соответствии с требуемой конфигурацией (пример файла предоставляется по запросу заказчика). Файл `production.yaml` переопределяет переменные в файле `default.yaml`.

При настройке `production.yaml` обязательно нужно проверить и при необходимости изменить (привести в соответствие с имеющимися данными) следующие параметры:

1. Секция `ss-chat`- веб-чат:
 - a. `ingresses.<host>.hosts` - хост-имя, по которому будет доступен веб-чат;
 - b. `genesis_adapter.api_url` - адрес сервиса `genesys-adapter`;
 - c. `sip.target_uri`, `sip.local_user`, `sip.transport`, `sip.prefer_codec` - параметры SIP-подключения к `genesys-adapter`.
2. Секция `genesys-adapter` - голосовой адаптер:
 - a. `envs.ORCHESTRATOR_ADDRESS` - адрес сервиса `orchestrator`;
 - b. `envs.SIP_SERVER_PORT` - порт SIP-сервера;
 - c. `envs.SIP_SERVER_PUBLIC_ADDRESS` - внешний адрес SIP-сервера;
 - d. `envs.SIP_SERVER_RTP_PORTS` - диапазон RTP-портов;
 - e. `secretEnvs.HTTP_PROXY`, `secretEnvs.HTTPS_PROXY` - параметры прокси (при необходимости);
3. Секция `orchestrator` - ядро диалогов:
 - a. `knowledge.embeddings.api_url`, `.api_key`, `.model_id` - параметры провайдера эмбеддингов;
 - b. `llm.api_url`, `.api_key`, `.model_id`, `.temperature`, `.stream` - параметры провайдера LLM;
4. Секция `rag-indexer` - индексация статей в Qdrant:
 - a. `envs.RABBITMQ_HOST`, `envs.RABBITMQ_USER`, `envs.RABBITMQ_PASSWORD` - подключение к RabbitMQ;
 - b. `envs.QDRANT_URL` - адрес Qdrant;
 - c. `envs.HTTP_PROXY`, `envs.HTTPS_PROXY`, `envs.NO_PROXY` - параметры прокси (при необходимости);
 - d. `embedder.id`, `.api_key`, `.base_url`, `.dimensions` - параметры провайдера эмбеддингов;
5. Секция `l2u-adapter` - адаптер базы знаний KMS:
 - a. `env.SYSTEM_RABBIT_USER`, `env.SYSTEM_RABBIT_PASSWORD` - креды RabbitMQ;
 - b. `system.l2u_in_knowledge.api_url` - адрес KMS;
 - c. `system.rabbit.host`, `.port`, `.exchange` - подключение к RabbitMQ;
6. Секция `model-registry-service` - runtime ML-моделей:
 - a. `envs.HTTP_PROXY`, `envs.HTTPS_PROXY`, `envs.NO_PROXY` - параметры прокси (при необходимости);

Остальные параметры (если ниже не указано иное) можно оставить без изменений.

6. Развертывание системы

Развертывание DRE AI Call Center Assistant осуществляется в Gitlab CI/CD:

1. В боковом меню выбрать **Build** (на панели слева) -> **Pipelines**.
2. Запустить pipeline, путем нажатия на шаг `init` (кнопка ">>"). Во всплывающем окне нажать кнопку Play.
3. Дождаться окончания операции.

© ООО "Цифра", 2026

Документация "DRE AI Call Center Assistant. Руководство по установке" является объектом авторского права. Воспроизведение всего произведения или любой его части воспрещается без письменного разрешения правообладателя.