

ПРОГРАММНАЯ ЧАСТЬ ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА

«ЭВИРИС»

РУКОВОДСТВО АДМИНИСТРАТОРА

Правообладатель: Общество с ограниченной
ответственностью «ГК «Иннотех» (ООО «ГК «Иннотех»)

Место нахождения и адрес правообладателя: 125167, г.
Москва, вн.тер.г. муниципальный округ Аэропорт, пр-кт
Ленинградский, д. 36, стр. 41, помещ. 23

СОДЕРЖАНИЕ

1. Термины и сокращения	3
2. Общие сведения.....	4
3. Предварительные требования	5
3.1. Состав дистрибутива.....	5
3.2. Требования к программному обеспечению	5
4. Установка	6
5. Запуск и остановка	6
6. Основные операции администратора.....	7
6.1. Создание пользователя	7
6.2. Удаление пользователя.....	7
6.3. Создание организации	7
6.4. Удаление организации.....	7
6.5. Добавление пользователя к организации	8
6.6. Удаление пользователя из организации	8
6.7. Назначение прав доступа организации в проекте.....	8
6.8. Назначение прав доступа пользователю в проекте	8
6.9. Редактирование прав доступа в проекте.....	8
6.10. Смена пароля пользователя.....	8
6.11. Удаление сведений о наборе данных	9
6.12. Удаление сведений о пайплайне.....	9
6.13. Формирование и корректировка списка пользователей проекта	9

1. Термины и сокращения

Термины и сокращения, используемые в контексте настоящего документа:

Термин, сокращение	Определение
Модель машинного обучения	Математический алгоритм, который обучается на наборах данных для выполнения конкретной задачи, такой как классификация или регрессия, без явного программирования. Файл с моделью машинного обучения формируется в результате обучения в Пайплайне AUTO ML
Пайплайн AUTO ML	Настраиваемый в части параметров обучения и источников данных, сохраняемый, корректируемый, переиспользуемый алгоритм в системе, который предназначен для обучения моделей машинного обучения в автоматическом режиме, а также применения модели в batch-режиме
Пайплайн подготовки	Настраиваемый, сохраняемый, переиспользуемый алгоритм в системе, который предназначен для обработки наборов данных, подаваемых на вход алгоритма с сохранением результирующего набора данных на выходе
Проект	Выделенное рабочее пространство в ПАК, предназначенное для взаимодействия участников проекта в задачах: объединения данных, обучения моделей машинного, применения модели для получения прогнозов, имеющих бизнес-ценность для участников проекта
Организация-участник программы	Организация, заключившая договор с Владельцем прогэраммы на работу с функциями программы
Администратор	Доверенное лицо владельца программы, имеющее доступ к элементам программы (база данных, хранилище и т. п.) для настройки сопровождения работы программы
Пользователь	Лицо из организации-участника, имеющее учетную запись в программе с определенными правами доступа

2. Общие сведения

Программная часть программно-аппаратного комплекса «Эвирис» (далее – программа) предназначена для автоматизированного обучения (AutoML) и последующего применения моделей машинного обучения. При этом пользователи на всех этапах использования программы имеют доступ исключительно к структуре данных и метаданным, не имея доступа непосредственно к наборам данных, используемых для машинного обучения и применения ML-моделей.

Пользователи взаимодействуют с программой через интерфейс (UI). Описание доступных действий пользователя приведены в документе «Руководство пользователя».

Часть действий, необходимых для работы программы, осуществляется вручную администратором. Такие действия описаны в настоящем документе.

Для начала работы администратора в программе предварительно должны быть выполнены действия по развертыванию программы, описанные в документе «Руководство по установке и эксплуатации».

3. Предварительные требования

В разделе приводится описание состава дистрибутива и требования к программному обеспечению программы.

3.1. Состав дистрибутива

Дистрибутивы, необходимые для установки программы, находятся в каталоге `dist` основного проекта.

В каталоге `dist` находятся следующие `docker`-образы, необходимые для работы программы:

- `api_gw-dbms-interaction-module.tar.gz`;
- `api_gw-auth-module.tar.gz`;
- `api_gw-backend.tar.gz`;
- `api_gw-command-processing-module.tar.gz`;
- `api_gw-sse-module.tar.gz` `api_`;
- `gw-nginx.tar.gz` `core-hive.tar.gz`;
- `core-spark-history-server.tar.gz`;
- `core-livy.tar.gz`;
- `core-receiving-control-module.tar.gz`;
- `core-command-orchestrator.tar.gz`;
- `core-spark.tar.gz`;
- `user-frontend.tar.gz`;
- `user-preparation-module.tar.gz`.

3.2. Требования к программному обеспечению

К компонентам программы предъявляются следующие минимальные системные требования:

- аппаратное обеспечение:
 - процессор: 4 ядра x86_64 (рекомендуется Intel Xeon / AMD EPYC);
 - оперативная память: от 16 ГБ (рекомендуется 32 ГБ);
 - дисковое пространство: не менее 500 ГБ SSD;
 - сетевая карта: 1 Gbps (желательно наличие 10 Gbps для MinIO/Spark);
- Программное обеспечение:
 - операционная система: Astra Linux 1.8 SE (или совместимая);
 - платформа виртуализации: `astra-kvm`;
 - дополнительное программное обеспечение: `Docker`, `docker-compose`, `Helm`, `OpenVPN`, `PostgreSQL`, `open-iscsi`, `HAProxy`, `keepalived`, `NFS`, `Kubernetes v1.30+`.

4. Установка

Действия, необходимые для развертывания программы описаны в документе «Руководство по установке и эксплуатации».

5. Запуск и остановка

Действия, необходимые для запуска и остановки компонентов программы описаны в документе «Руководство по установке и эксплуатации».

6. Основные операции администратора

В разделе приводится описание следующих операций, доступных администратору программы:

- создание пользователя;
- удаление пользователя;
- создание организации;
- удаление организации;
- добавление пользователя к организации;
- удаление пользователя из организации;
- назначение прав доступа организации в проекте;
- назначение прав доступа пользователю в проекте;
- редактирование прав доступа в проекте;
- смена пароля пользователя;
- удаление сведений о наборе данных;
- удаление сведений о пайплайне;
- формирование и корректировка списка пользователей проекта.

Для выполнения описываемых ниже операций (действий) администратору необходимо выполнить подключение к базе данных с помощью `bash`-команды:

```
psql -U имя_пользователя -d имя_базы_данных.
```

6.1. Создание пользователя

Для создания пользователя необходимо выполнить следующий `sql`-запрос:

```
INSERT INTO users (username, password, organization_id, status)  
VALUES ('имя_пользователя', 'хэш_пароля', организация_id, 1)
```

6.2. Удаление пользователя

Для удаления пользователя необходимо выполнить следующий `sql`-запрос:

```
DELETE FROM users WHERE id = id_пользователя;
```

6.3. Создание организации

Для создания организации необходимо выполнить следующий `sql`-запрос:

```
INSERT INTO organizations (name, description, inn, ogrn, address, status)  
VALUES ('название', 'описание', 'ИНН', 'ОГРН', 'адрес', 1)
```

6.4. Удаление организации

Для удаления организации необходимо выполнить следующий `sql`-запрос:

```
DELETE FROM organizations WHERE id = id_организации
```

6.5. Добавление пользователя к организации

Для добавления пользователя к организации необходимо выполнить следующий sql-запрос:

```
UPDATE users  
SET organization_id = id_организации  
WHERE id = id_пользователя
```

6.6. Удаление пользователя из организации

Для удаления пользователя из организации необходимо выполнить следующий sql-запрос:

```
UPDATE users  
SET organization_id = NULL  
WHERE id = id_пользователя;
```

6.7. Назначение прав доступа организации в проекте

Для назначения прав доступа организации в проекте необходимо выполнить следующий sql-запрос:

```
INSERT INTO projects_organizations_roles (project_id, organization_id, role_id)  
VALUES (id_проекта, id_организации, id_роли)
```

6.8. Назначение прав доступа пользователю в проекте

Для назначения прав доступа пользователю в проекте необходимо выполнить следующий sql-запрос:

```
INSERT INTO users_projects_roles (user_id, role_id, project_id)  
VALUES (id_пользователя, id_роли, id_проекта)
```

6.9. Редактирование прав доступа в проекте

Для редактирования прав доступа пользователя и организации в проекте необходимо выполнить один из следующих sql-запросов:

- для пользователя:

```
UPDATE users_projects_roles  
SET role_id = новый_id_роли  
WHERE user_id = id_пользователя AND project_id = id_проекта
```
- для организации:

```
UPDATE projects_organizations_roles  
SET role_id = новый_id_роли  
WHERE organization_id = id_организации AND project_id = id_проекта
```

6.10. Смена пароля пользователя

Для смены пароля пользователя необходимо выполнить следующий sql-запрос:

```
UPDATE users
SET password = 'новый_хэш_пароля',
    password_expiration = CURRENT_TIMESTAMP
WHERE id = id_пользователя
```

6.11. Удаление сведений о наборе данных

Для удаления сведений о наборе данных необходимо выполнить следующий sql-запрос:

```
DELETE FROM datasets WHERE id = id_набора_данных
```

6.12. Удаление сведений о пайплайне

Для удаления сведений о пайплайне необходимо выполнить следующий sql-запрос:

```
DELETE FROM pipelines WHERE id = id_пайплайна
```

6.13. Формирование и корректировка списка пользователей проекта

Для формирования и корректировки списка пользователей проекта необходимо выполнить один из следующих sql-запросов:

- добавление нового пользователя в проект (см. п. 6.8):

```
INSERT INTO users_projects_roles (user_id, role_id, project_id)
VALUES (id_пользователя, id_роли, id_проекта);
```
- удаление пользователя из проекта:

```
DELETE FROM users_projects_roles
WHERE user_id = id_пользователя AND project_id = id_проекта;
```
- изменение роли участника в проекте (см. п. 6.9):

```
UPDATE users_projects_roles
SET role_id = новый_id_роли
WHERE user_id = id_пользователя AND project_id = id_проекта;
```