ОГЛАВЛЕНИЕ

1.Установка 2
1.1. Установка на Linux 2
1.2. Запуск с помощью docker 2
2. Описание основного меню 3
3. Настройка 3
3.1. Настройка лицензии 3
3.2. Синхронизация между устройствами 4
3.2.1. Клиент 4
3.2.2. Сервер 5
3.3. Обновление 6
4. Обучение модели 6
4.1. Управление классами и изображениями 7
4.2. Калибровка данных 8
4.3. Управление моделями 9
5. Тест модели 9
6. Использование API 10
6.1. GET STATUS 10
6.2. PREDICT 10

1.УСТАНОВКА

1.1. УСТАНОВКА НА LINUX

Скачиваем файл:

```
cd /opt
wget https://ai-dev.perfobit.ru/files/cls_perfo/cls_perfo.zip
```

Распаковываем архив:

```
mkdir cls_perfo
unzip cls_perfo.zip -d  cls_perfo
cd cls_perfo
chmod +x Server.Web.Starter
chmod +x Server.Web
```

Запускаем:

```
./Server.Web.Starter --urls "http://+:8090"
```

Приложение доступно по порту *8090*, например: http://localhost:8090

Чтобы создать службу на ubuntu нужно создать файл */etc/systemd/system/cls_perfo.service*

```
#------------файл cls_perfo.service----------------
[Unit]
Description=Perfobit AI-CLS service

[Service]
Type=notify
ExecStart=/opt/cls_perfo/Server.Web.Starter --urls "http://+:8090"

[Install]
WantedBy=multi-user.target
#------------------------------
```

Далее выполняем:

```
systemctl daemon-reload
systemctl start cls_perfo
```

Чтобы включить запуск службы при каждой перезагрузке, используем команду:

```
systemctl enable cls_perfo
```

1.2. ЗАПУСК С ПОМОЩЬЮ DOCKER

Установить докер по инструкции (для ОС Ubuntu)
https://docs.docker.com/engine/install/ubuntu/

Запуск:

```
docker run -dit --restart=unless-stopped --name perfobit_cls -p 8090:8090 -v
/opt/cls:/app/AppData perfobit/cls_perfo:latest
```

Папка /opt/cls_perfo должна быть доступна пользователю, от которого запущен docker на запись и чтение

2. ОПИСАНИЕ ОСНОВНОГО МЕНЮ

В верхней части экрана находится основное меню

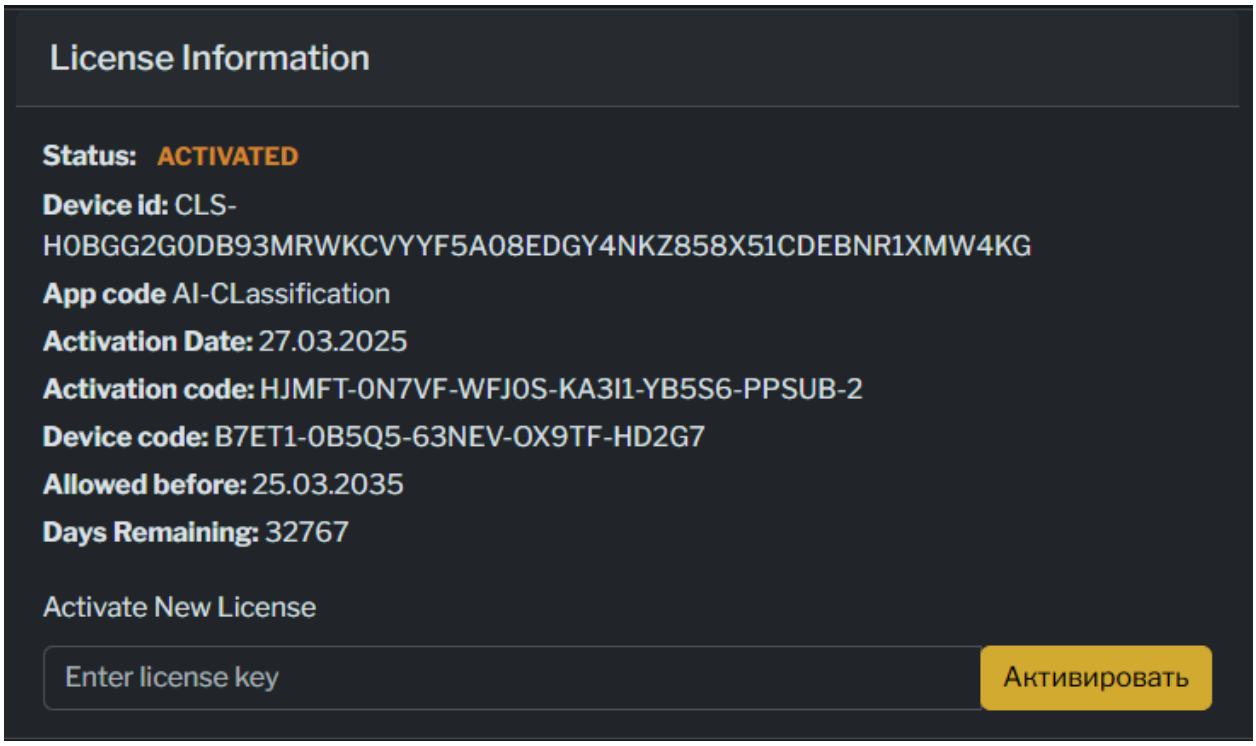
Элементы основного меню:

Номер	Описание
1	Переход на страницу тестирование моделей
2	Переход на страницу обучения моделей
3	Оповещения для пользователя. В верхнем правом углу отображается кол-во оповещений
4	Изменение языка визуального интерфейса. Поддерживается русский и английский языки
5	Переход на страницу настроек и информации о системе

3. НАСТРОЙКА

3.1. НАСТРОЙКА ЛИЦЕНЗИИ

На странице НАСТРОЙКИ в блоке ЛИЦЕНЗИЯ доступна информация о текущей активной лицензии



Поле **STATUS** отображает текущий статус лицензии

Поле **ACTIVATION DATE** отображает дату первого использования приложения

Поле **ALLOWED BEFORE** отображает дату последнего дня, когда лицензия активна

Лицензировать необходимо каждое устройство, которое используется как для тренировки, так и для распознавания

Также функция распознавания будет доступна, если лицензия на нужном устройстве была активна на момент тренировки модели.

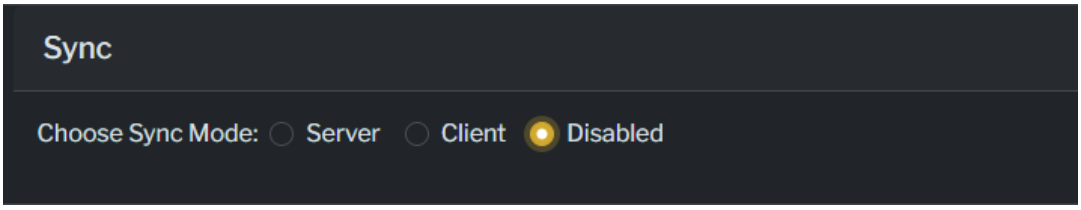
ВАЖНО! Для нормальной работы лицензий на устройстве должно быть выставлено правильное текущее время и дата. Если этого не сделать, то лицензия может не активироваться

Для получения новой лицензии необходимо прислать письмо по адресу info@perfobit.ru, тема письма *«Получение новой лицензии для функции распознавания AI-CLassification на устройстве»*

следующие поля: **DEVICE ID, APP CODE, ACTIVATION DATE, DEVICE CODE**

3.2. СИНХРОНИЗАЦИЯ МЕЖДУ УСТРОЙСТВАМИ

На странице НАСТРОЙКИ в блоке СИНХРОНИЗАЦИЯ доступна функция синхронизации между устройствами, которая работает по принципу клиент-сервер. Необходимо 1 устройство сделать сервером, а все остальные устройства – клиентами и у каждого клиента указать адрес сервера.

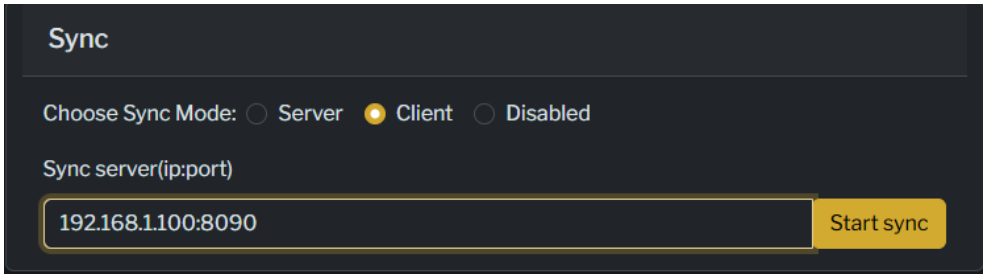


Режим Сервер оптимально включать на том устройстве, на котором происходит обучение новой модели, но можно использовать и другие схемы работы совместно с режимом выгрузки и загрузки модели, описанной в разделе Обучения

ВАЖНО! Если вы не хотите сломать режим распознавания на рабочих устройствах, то во время настройки, тестировании и обучении новой модели необходимо переключить сервер в режим Disabled

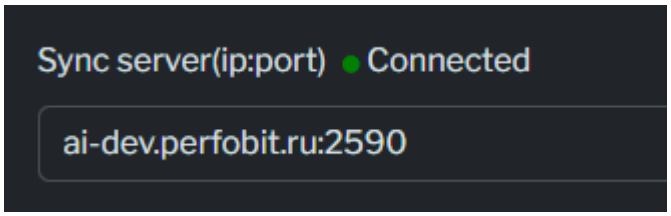
3.2.1. КЛИЕНТ

Для настройки клиента необходимо выбрать режим КЛИЕНТ, и прописать адрес сервера и порт



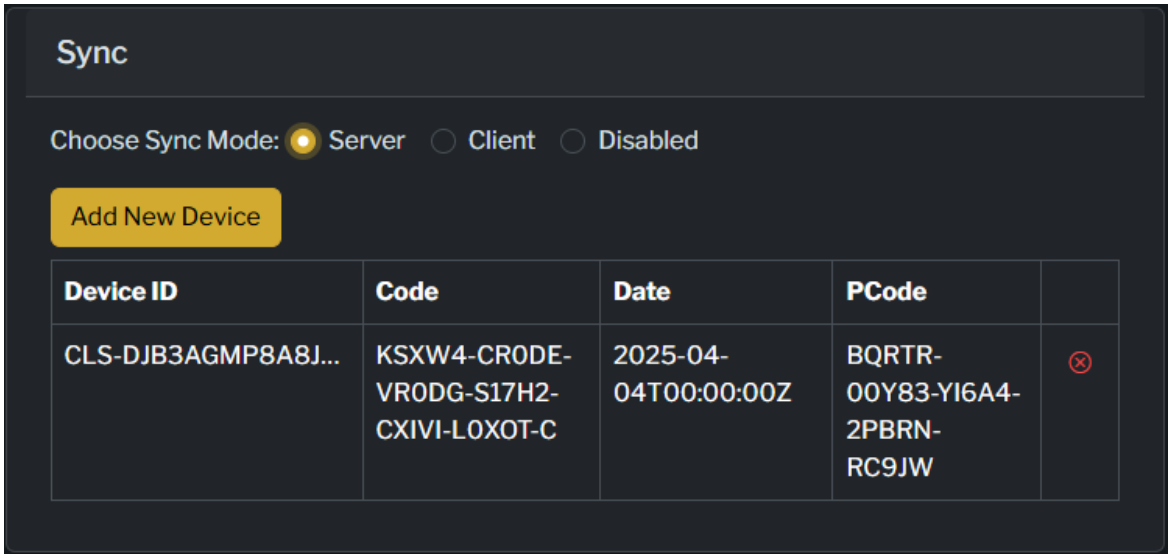
После настройки и нажатия на кнопку START SYNC все клиенты подключаются к серверу и автоматически устанавливают нужные лицензии и новые модели для распознавания

При успешном подключении к серверу на странице отображается статус подключения



3.2.2. СЕРВЕР

Для настройки сервера необходимо выбрать режим СЕРВЕР

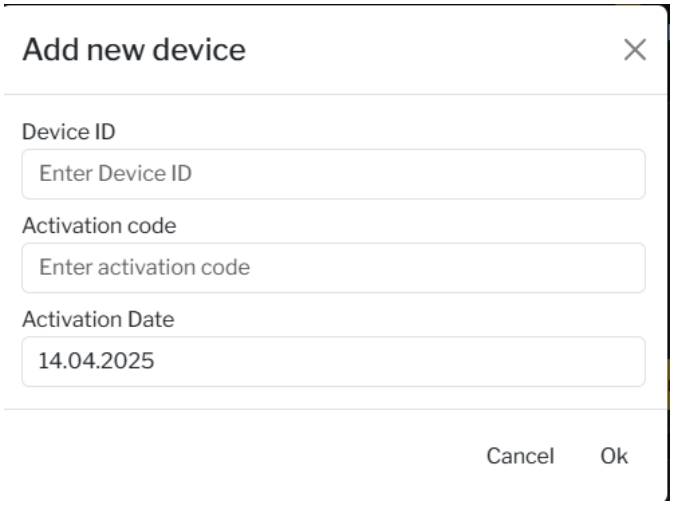


После переключения в режим СЕРВЕР, устройство автоматически готово к подключению клиентов

ВАЖНО! Необходимо убедиться, что все клиенты видят сервер по сети, и выполнить настройки во всех брандмауэрах и файрволлах на пути от клиента к серверу, открыв доступ к порту сервера по протоколу tcp, в наших примерах это порт 8090

В этом же блоке есть таблица с устройствами. В нее надо добавлять лицензии всех устройств, на которые необходимо доставить лицензию. Клиенты при подключении проверяют и скачивают обновленную лицензию.

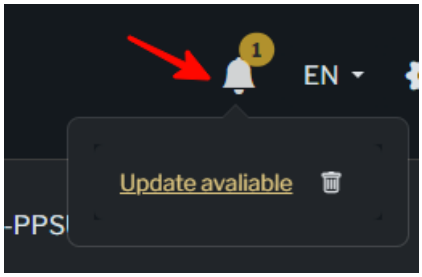
Для добавления устройства необходимо нажать на соответствующую кнопку сверху таблицы и заполнить поля в модальном окне.



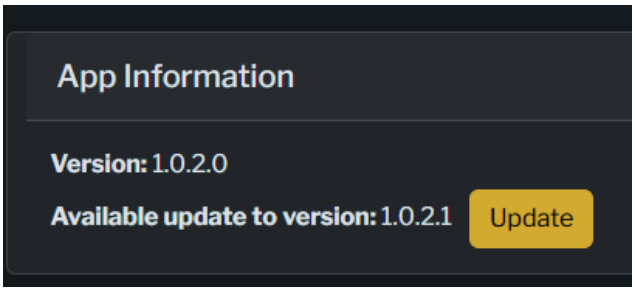
ВАЖНО! В поле **ACTIVATION DATE** необходимо вводить ту дату, которую получили в письме с лицензией. Если ввести неправильную дату, то лицензия работать не будет!

3.3. ОБНОВЛЕНИЕ

Если доступна новая версия программы, то вам приходит уведомление:



Для обновление необходимо перейти в НАСТРОЙКИ в раздел О ПРИЛОЖЕНИИ и нажать на кнопку ОБНОВИТЬ

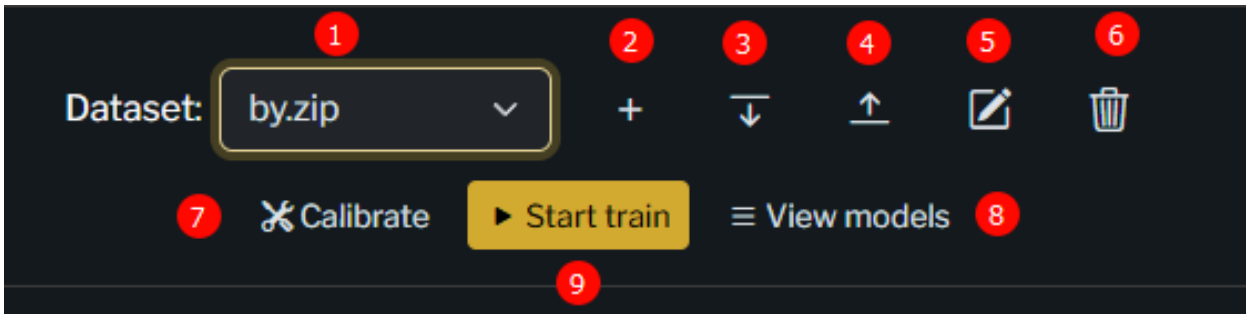


После завершения обновления станет доступна новая версия приложения

4. ОБУЧЕНИЕ МОДЕЛИ

Для обучения модели необходимо перейти в соответствующий раздел из основного меню.

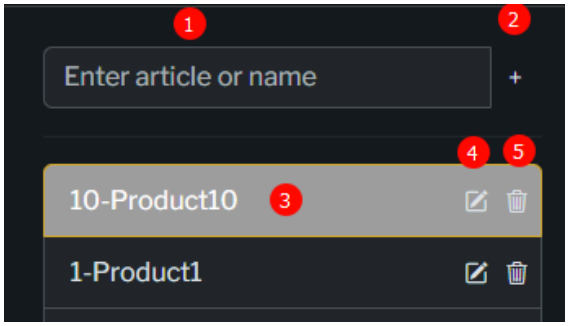
В верхней части доступно управление наборами данных, на основании которых происходит обучение новой модели распознавания



Номер	Описание
1	Выбор набора данных из списка
2	Добавление нового(пустого) набора данных
3	Импорт готового набора данных из zip архива. Архив должен состоять из папок (название класса) внутри которых должны лежать изображения
4	Экспорт набора в zip архив
5	Редактирование имени набора
6	Удаление набора
7	Калибровка данных
8	Запуск процесса обучения
9	Управление моделями

4.1. УПРАВЛЕНИЕ КЛАССАМИ И ИЗОБРАЖЕНИЯМИ

В левой части экрана расположен блок управления классами



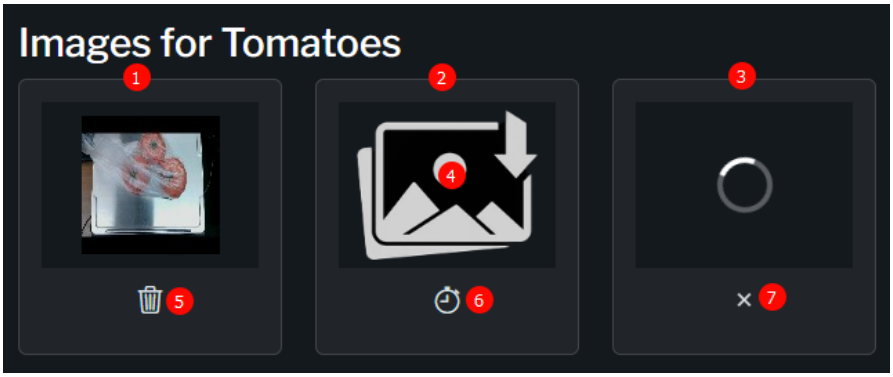
Для поиска необходимо набирать в текстовом поле (1) имя или артикул класса, список ниже будет фильтроваться в зависимости от введенного текста.

Для добавления нового класса необходимо нажать на (2) и заполнить имя и артикул класса.

Для редактирования используется кнопка (4), в модальном окне можно отредактировать имя и артикул

Для удаления нужно использовать кнопку (5), все связанные с этим классом изображения также будут удалены.

При нажатии на строку (3) справа отображаются 10 изображений данного класса. Каждая ячейка может быть представлена 3-мя состояниями:

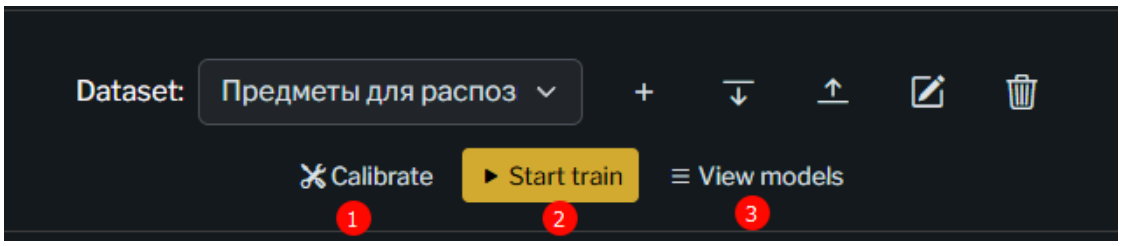


(1) изображение загружено, отображается его уменьшенная копия, при нажатии на изображение - оно отображается в полный размер
(2) изображение отсутствует
(3) ожидание изображения от устройства

Возможные действия с ячейками:

(4) добавить изображение из файла в файловой системе устройства
(5) удалить изображение
(6) запустить режим ожидания изображения от устройства (3), далее нужно положить предмет на платформу
(7) отменить ожидание (3)

ВАЖНО! Для корректной работы режима ожидания необходимо настроить отправку изображения от устройства (или его ПО) через метод **predict**



После загрузки изображений всех классов необходимо провести калибровку набора (1), а затем приступить к обучению модели (2). По завершению обучения результат можно посмотреть в списке моделей (3)

ВАЖНО! Чтобы модель была точнее, необходимо загрузить все 10 изображений класса, каждое изображение должно чем-то отличаться от предыдущего, например положением на платформе или количеством объектов на них

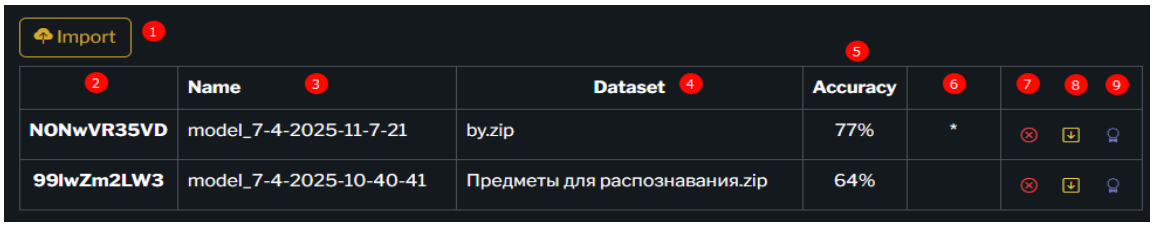
4.2. КАЛИБРОВКА ДАННЫХ

Для калибровки устройства необходимо нажать на соответствующую кнопку в верхнем блоке управления данными. На экране отобразится первое изображение из первого класса. С помощью инструмента калибровки (1) необходимо выделить область, в которой будет находиться объект для распознавания. Калибровка производится один раз сразу на весь набор данных. После выставления необходимо пробежать по всем изображениям с помощью (2) и (3)

ВАЖНО! Калибровку необходимо производить в самом конце, когда все изображения добавлены в класс.

4.3. УПРАВЛЕНИЕ МОДЕЛЯМИ

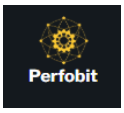
После окончания обучения в таблице моделей можно сравнить новую модель с предыдущими по колонке ACCURACY.



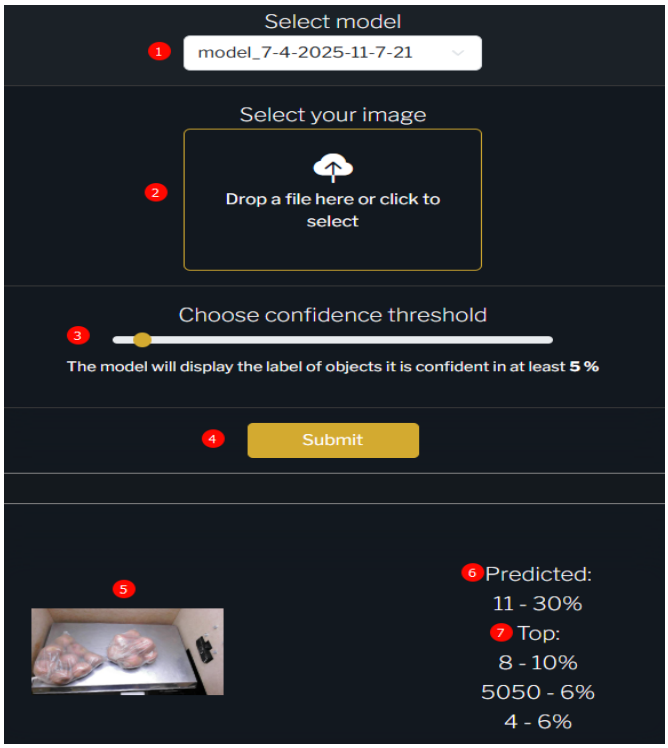
(1) Импорт выгруженной модели
(2) Идентификатор модели
(3) Наименование
(4) Данные, на которых модель была обучена
(5) Точность модели (посчитана методом перебора всех изображений текущего набора)
(6) Модель по умолчанию, значком (*) помечается основная модель, которая используется для распознавания.
(7) Удалить модель
(8) Экспорт модели в архив. Далее ее можно загрузить на других устройствах с помощью пункта (1)
(9) Установить модель по умолчанию, если нажать на этот значок, то модель становится основной для распознавания и помечается значком (*)

5. ТЕСТ МОДЕЛИ

Для тестирования модели необходимо нажать на логотип в заголовке



Далее выбираем нужную модель из списка (1)



Затем выбираем тестируемый файл (2) и выбираем необходимую точность (3)
Нажимаем на кнопку (4) и изображение (5) отправляется для распознавания
Далее смотрим на результат (6) – это класс, который обладает максимальным значением точности
(7) - список из других подходящих классов, которые попадают в предел точности (3)

6. ИСПОЛЬЗОВАНИЕ API

6.1. GET STATUS

Метод проверки доступности службы распознавания.

https://localhost:7139/get_status

Статусы ответа:

200 - ОК

4XX-5XX - любой код из этого диапазона является ошибкой, например 500, 502, 404, 400 и т.д.

6.2. PREDICT

Основной метод отправки данных для распознавания

https://ai-dev.perfobit.ru/predict?Treshold=5&Top=4

Данный метод сильно зависит от производительности оборудования...

Параметры запроса:

Параметр		Тип	Обяз.	Описание
Treshold	Query	int	+	Порог срабатывания, можно отправлять как в процентах (например, 5; 10; 15) так и в десятичных дробях (например, 0,05; 0,1; 0,15)
Top	Query	int	+	Количество значений в результате, которые попадают в порог (Treshold)
Model	Query	string	-	Выбранная модель, если не указывать, то будет использована модель по умолчанию
Тело запроса (Body)	Body	bytes	+	Файл с изображением в формате multipart/form-data

Пример запроса:

```
curl -X 'POST' \
                    'https://ai-dev.perfobit.ru/predict?Treshold=0%2C05&Top=5' \
                    -H 'accept: */*' \
                    -H 'Content-Type: multipart/form-data' \
                    -F 'File=@camera_3_20231120203424.jpg;type=image/jpeg'
```

Пример тела запроса:

```
------WebKitFormBoundaryy4txOvyoZitDW7YU
        Content-Disposition: form-data; name="File"; filename="camera_3_20231120203424.jpg"
        Content-Type: image/jpeg
        <data>
------WebKitFormBoundaryy4txOvyoZitDW7YU—
```

Статусы ответа:

200 - ОК

4XX-5XX - любой код из этого диапазона является ошибкой, например 500, 502, 404, 400 и т.д.

Параметры ответа

Параметр	Тип	Обяз.	Описание
classId	string	+	идентификатор класса, внутреннее значение
className	string	+	имя класса, введенное пользователем на форме создания карточки класса
article	string	-	артикул класса (Item Code), введенное пользователем на форме создания карточки класса
probability	int	+	вероятность в %, может принимать значения от 0 до 100
top	[]	-	массив с последующими вероятностями

Если в запросе было указано top=5, то в ответе основное значение (с наибольшим значением probability) будет первым, а остальные 4 значения будут в массиве top

Пример ответа:


```
{
   "classId": 11,
   "className": "Product1",
   "article": "000",
   "probability": 29,
   "top": [
      {
         "classId": 7,
         "className": "Product2",
         "article": "111",
         "probability": 25
      },
      {
         "classId": 23,
         "className": "Product3",
         "article": "222",
         "probability": 7
      },
      {
         "classId": 15,
         "className": "Product4",
         "article": "333",
         "probability": 6
      },
      {
         "classId": 13,
         "className": "Product5",
         "article": "444",
         "probability": 5
      }
   ]
}
```