

Инструкция по установке экземпляра программного обеспечения
«Система психографического и эмоционального анализа поведения людей
на основе видео-данных»

Минимальные системные требования и порядок конфигурации ПО описаны в инструкции по эксплуатации экземпляра ПО.

1. Установка ПО сервера видеоаналитики

1) Устанавливаем Docker Engine

Выполняем установку инструментов для добавления репозитория: >>

```
sudo apt install apt-transport-https ca-certificates curl gnupg lsb-release
```

Добавляем репозиторий Docker Engine для Ubuntu 20.04:

```
>> curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --  
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

```
>> sudo echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]  
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

```
>> sudo apt update
```

Устанавливаем требуемые пакеты Docker:

```
>> sudo apt install docker-ce docker-ce-cli containerd.io docker-compose
```

```
>> sudo systemctl --now enable docker
```

Проверяем работоспособность docker:

```
>> sudo docker run hello-world
```

Результат выполнения команды должен соответствовать рис. 1.

```
root@awnb-cv:~# sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b8dfde127a29: Pull complete
Digest: sha256:df5f5184104426b65967e016ff2ac0bfcd44ad7899ca3bbcf8e44e4461491a9e
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

Рисунок 1 – Результат корректного выполнения команды docker run hello-world

2) Установка ПО видеоаналитики

Скачиваем экземпляр ПО и копируем на сервер архив *stela.tag.gz*, который включен в архив с программным обеспечением (*emotion.tar.gz*).

Разархивируем архив с сохранением атрибутов каталогов

```
>> sudo tar -xzpf stela.tag.gz
```

Переходим в папку с ПО

```
>> cd stela/
```

Выполняем запуск докер-контейнеров с ПО:

```
>> sudo docker-compose up -d
```

Проверяем, что контейнеры запущены

```
>> sudo docker ps
```

Результат выполнения команды должен соответствовать рис. 2.

```
pngn@awnb-cv:~/stela$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
6defe0b5fa33  camera_0     "python3 app.py"        About a minute ago    Up 8 seconds          camera_0
6abb9a8711cb  recog        "python3 recognition..." About a minute ago    Up 8 seconds          recog
49d476ca5f1f  data_sender  "python3 data_sender..." About a minute ago    Up 8 seconds          data_sender
deca83caa475  param_server "python3 param_serve..." About a minute ago    Up 7 seconds          param_server
F1b0fcd4ec2d  cronjobs     "cronld -f -d 8"        About a minute ago    Up 9 seconds          cronjobs
8fa626df6f20  redis:latest "docker-entrypoint.s..." About a minute ago    Up 9 seconds          6379/tcp            redis
```

Рисунок 2 – Результат выполнения команды docker ps на сервере видеоаналитики

2. Установка ПО центрального сервера

1) Устанавливаем Docker Engine

Выполняем установку инструментов для добавления репозитория:

```
>> sudo apt install apt-transport-https ca-certificates curl gnupg lsb-release
```

Добавляем репозиторий Docker Engine для Ubuntu 20.04:

```
>> curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --  
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

```
>> sudo echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]  
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

```
>> sudo apt update
```

Устанавливаем требуемые пакеты Docker:

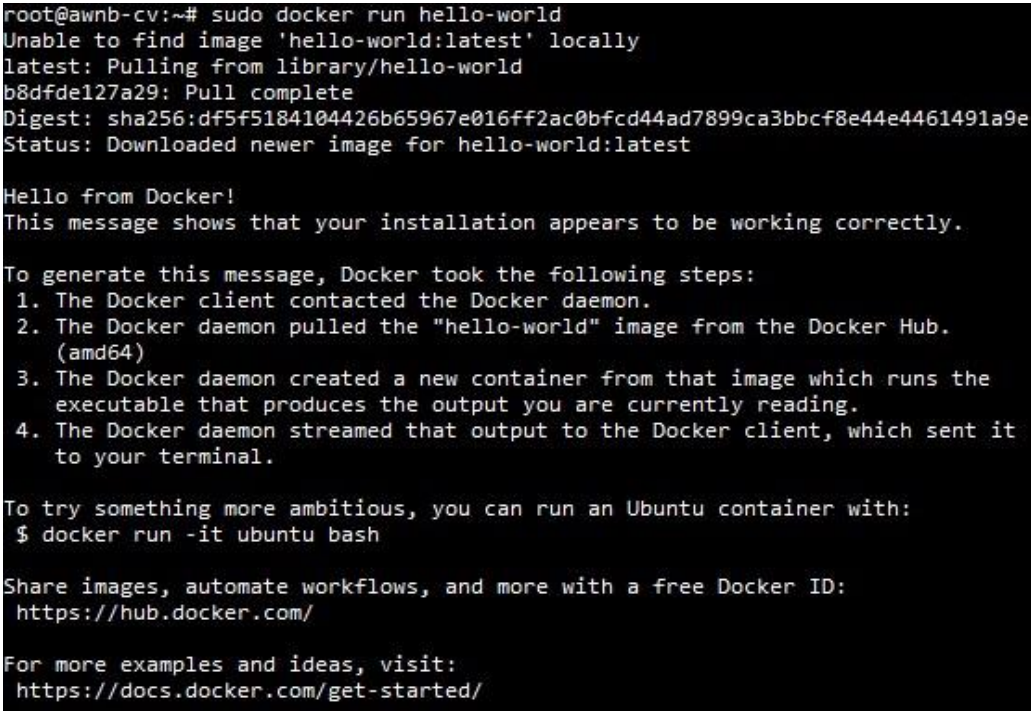
```
>> sudo apt install docker-ce docker-ce-cli containerd.io docker-compose
```

```
>> sudo systemctl --now enable docker
```

Проверяем работоспособность docker:

```
>> sudo docker run hello-world
```

Результат выполнения команды должен соответствовать рис. 3.

The image shows a terminal window with the following text:

```
root@awnb-cv:~# sudo docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
b8dfde127a29: Pull complete  
Digest: sha256:df5f5184104426b65967e016ff2ac0bfcd44ad7899ca3bbcf8e44e4461491a9e  
Status: Downloaded newer image for hello-world:latest  
  
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
  
To generate this message, Docker took the following steps:  
1. The Docker client contacted the Docker daemon.  
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
   (amd64)  
3. The Docker daemon created a new container from that image which runs the  
   executable that produces the output you are currently reading.  
4. The Docker daemon streamed that output to the Docker client, which sent it  
   to your terminal.  
  
To try something more ambitious, you can run an Ubuntu container with:  
$ docker run -it ubuntu bash  
  
Share images, automate workflows, and more with a free Docker ID:  
https://hub.docker.com/  
  
For more examples and ideas, visit:  
https://docs.docker.com/get-started/
```

Рисунок 3 – Результат корректного выполнения команды `docker run hello-world`

2) Установка ПО web-сервера

Скачиваем экземпляр ПО и копируем на сервер архив *stele.tar.gz*, который включен в архив с программным обеспечением (*emotion.tar.gz*).

Разархивируем архив с сохранением атрибутов каталогов

```
>> sudo tar -xzf stele.tar.gz
```

Переходим в папку с ПО

```
>> cd stele/
```

Исправляем права доступа к файлам:

```
>> sudo chmod 777 -R ./src
```

```
>> sudo chmod 777 -R ./images
```

```
>> sudo chown -R www-data:www-data ./src >>
```

```
sudo chown -R www-data:www-data ./images
```

Выполняем запуск docker-контейнеров с ПО:

```
>> sudo docker-compose up -d
```

Проверяем, что контейнеры запущены

```
>> sudo docker ps
```

Результат выполнения команды должен соответствовать рис. 4.

```
pngn@CV-MUC02:~/stela$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED    STATUS    PORTS                               NAMES
1b432e7d3052   webui_nginx   "/docker-entrypoint..." 2 minutes ago Up About a minute 0.0.0.0:80->80/tcp, :::80->80/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp stella-nginx
9e538a0fcafe   webui_postgres "docker-entrypoint.s..." 2 minutes ago Up About a minute 0.0.0.0:5434->5432/tcp, :::5434->5432/tcp stella-postgres
881d4b0cf25e   webui_php     "/docker-php-entrypoi..." 2 minutes ago Up About a minute 9000/tcp stella-php
```

Рисунок 4 – Набор контейнеров web-сервера Возвращаемся

в домашний каталог

```
>> cd ..
```

3) Установка ПО индексации лиц

Скачиваем экземпляр ПО и копируем на сервер архив `face_index.tar.gz`, который включен в архив с программным обеспечением (`emotion.tar.gz`).

Разархивируем архив с сохранением атрибутов каталогов

```
>> sudo tar -xzipf face_index.tar.gz
```

Переходим в папку с ПО

```
>> cd face_index/
```

Выполняем запуск docker-контейнеров с ПО:

```
>> sudo docker-compose up -d
```

Проверяем, что контейнеры запущены

```
>> sudo docker ps
```

Результат выполнения команды должен соответствовать рис. 5.

```
pngn@CV-MUC02:~/face_index$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED    STATUS    PORTS                               NAMES
9280bc5131e3   index_server   "nginx -b 0.0.0.0..."  About a minute ago Up 9 seconds 0.0.0.0:8080->80/tcp, :::8080->80/tcp index_server
6927e3040b2b   postgres:latest "docker-entrypoint.s..."  About a minute ago Up 9 seconds 5432/tcp postgres
```

Рисунок 5 – Набор контейнеров web-сервер