



Система управления  
информатизацией  
*Инструкция по установке экземпляра ПО*

# Содержание

1. Словарь терминов .....	1
2. Системные требования .....	2
2.1. Требования к серверу .....	2
2.2. Требования к клиенту .....	2
3. Порядок установки виртуальной машины: .....	3
4. Внешнее подключение к виртуальной машине .....	5
5. Функциональные характеристики экземпляра ПО .....	6
6. Описанием расположения файлов установленного ПО и его компонентов .....	7
6.1. Docker .....	7
6.2. Nginx .....	7
6.3. PostgreSQL .....	8
6.4. pgAdmin .....	8
6.5. Java 13 .....	8
6.6. Web-клиент .....	8

# 1. Словарь терминов

Термин/сокращение	Описание
ВМ	Виртуальная машина
ОЗУ	Оперативная память компьютера
ОС	Операционная система
ПО	Программное обеспечение
СУБД	Система управления базами данных
Docker	ПО для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации
systemd	Подсистема инициализации и управления службами Linux

## 2. Системные требования

### 2.1. Требования к серверу

1. Компьютер с ОС
  - Linux
    - Debian (Buster 10)
    - Ubuntu (Xenial 16.04 LTS)
  - MacOS (10.15 Catalina и выше)
  - Microsoft Windows 10
    - Professional или Enterprise 64-bit
    - Home 64-bit with WSL 2
2. Минимальные требования к физическому ОЗУ: 4ГБ, Рекомендуемые 8ГБ.
3. Минимальные требования к виртуальному ОЗУ (Память, выделяемая для виртуальной машины): 4ГБ, рекомендуемые: 12ГБ.
4. Свободная память на жестком диске (рекомендуется SSD): минимально 8ГБ, рекомендуется 30ГБ.

### 2.2. Требования к клиенту

Для доступа к интерфейсу пользователя необходим интернет-обозреватель с представленной в списке или более новой версией:

- Google Chrome - 63,
- Mozilla Firefox - 57,
- Apple Safari - 11,
- Microsoft Edge - 79.

## 3. Порядок установки виртуальной машины:

1. Установить систему виртуализации, поддерживающую формат **.ova** (в инструкции будет описан способ с использованием **VirtualBox**). Актуальную версию VirtualBox для вашей ОС можно загрузить по ссылке [www.virtualbox.org/wiki/Downloads](http://www.virtualbox.org/wiki/Downloads).
2. Загрузить файл виртуальной машины с предустановленной ОС Debian GNU/Linux 10.
3. Смонтировать образ в системе виртуализации:
  1. Открыть приложение VirtualBox;
  2. В меню выбрать **Файл** > **Импорт конфигураций**;
  3. В окне выбора конфигураций указать:
    - источник: **Локальная файловая система**;
    - файл: путь к файлу **inkontext.ova**;
  4. Следовать шагам, предлагаемым программой.
4. Настроить виртуальную машину до первого запуска:
  1. Выбрать в списке VM **inkontext** и нажать **настроить**
  2. Выбрать вкладку **Сеть**;
  3. Выбрать
    - тип подключения: **Сетевой мост**,
    - имя: **Ваш способ подключения компьютера к сети**;
  4. Проверить отсутствие ошибок.
    - В случае возникновения ошибок, связанных с отсутствием периферических, либо других устройств, отключить их в настройках виртуальной машины на вкладке **Порты**.

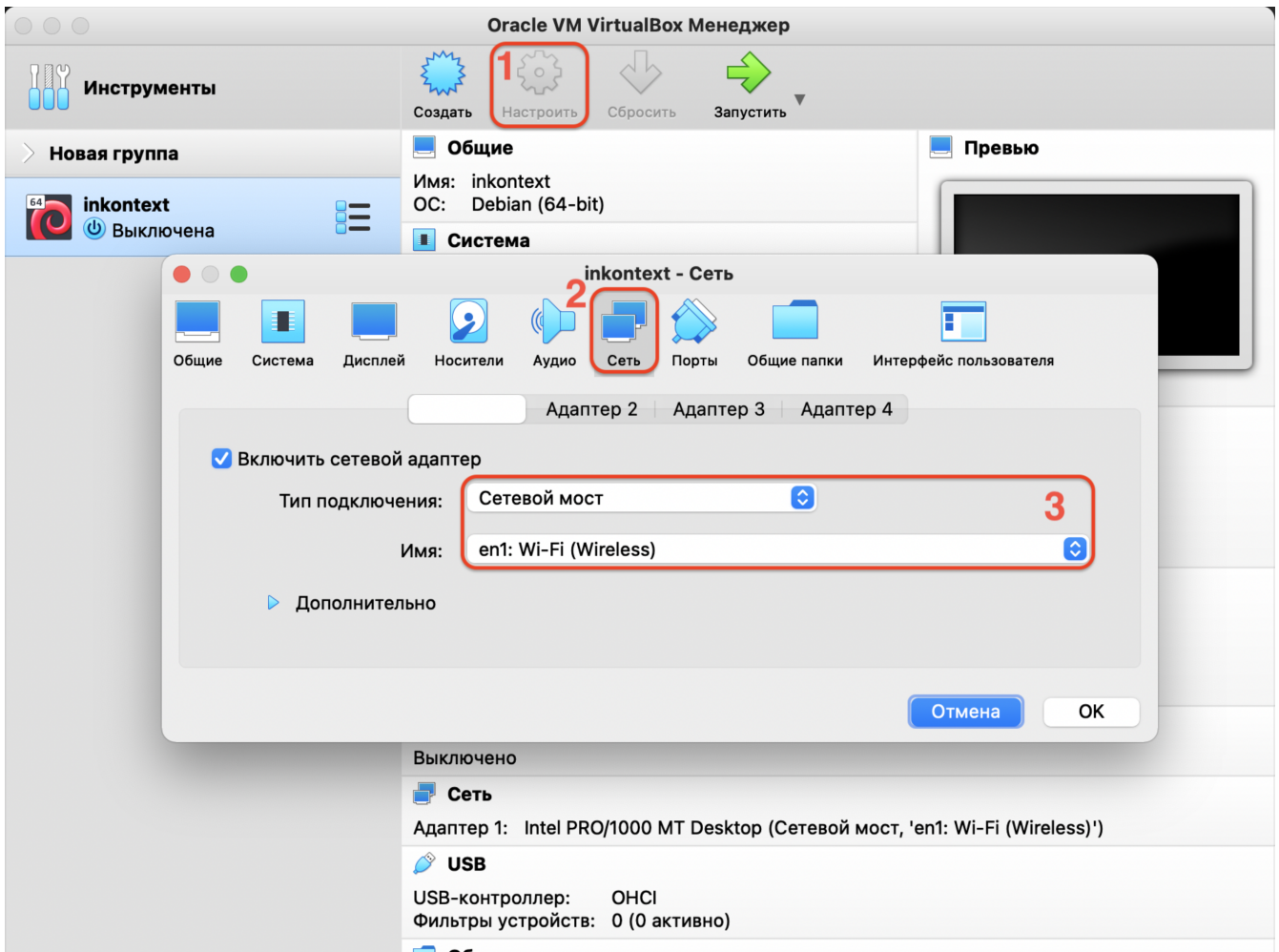


Рисунок 1. Настройки сети для подключения к VM VirtualBox

5. Запустить виртуальную машину
6. Подключиться к виртуальной машине:
  - login: `inkontext`
  - password: `debianinkontext`

## 4. Внешнее подключение к виртуальной машине

1. Проверить IP-адрес для внешнего подключения:
2. Введите команду:

```
ip addr show enp0s3
```

2. IP-адрес виртуальной машины для сетевого подключения `enp0s3` отображается в строке `inet`, как показано на снимке экрана. В данном примере IP-адрес: `10.0.1.29`

```
inkontext@inkontext:/opt/docker$ ip addr show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:14:2d:29 brd ff:ff:ff:ff:ff:ff
    inet 10.0.1.29/24 brd 10.0.1.255 scope global dynamic enp0s3
        valid_lft 86013sec preferred_lft 86013sec
    inet6 fe80::a00:27ff:fe14:2d29/64 scope link
        valid_lft forever preferred_lft forever
```

Рисунок 2. Вывод информации о сети

2. Проверить внешнее подключение к виртуальной машине через терминал по IP-адресу полученному выше:

```
ssh inkontext@10.0.1.29
```

Для удобства проверки расположения файлов установлена программа для управления файлами **Midnight commander**, который запускается с помощью команды:

```
mc
```

Установлена служба **sudo** для доступа к ресурсам виртуальной машины с правами `root`. Пароль `root`: `debianinkontext`

## 5. Функциональные характеристики экземпляра ПО

Проверить заявленные функциональные характеристики Системы управления информатизацией возможно с помощью браузера, представленного в системных требованиях к клиенту.

Для входа в программу необходимо подключиться по IP-адресу, полученному выше URL <http://10.0.1.29>



## 6. Описанием расположения файлов установленного ПО и его компонентов

Основные файлы программного обеспечения расположены в папке `/opt` и представлены на снимке экрана

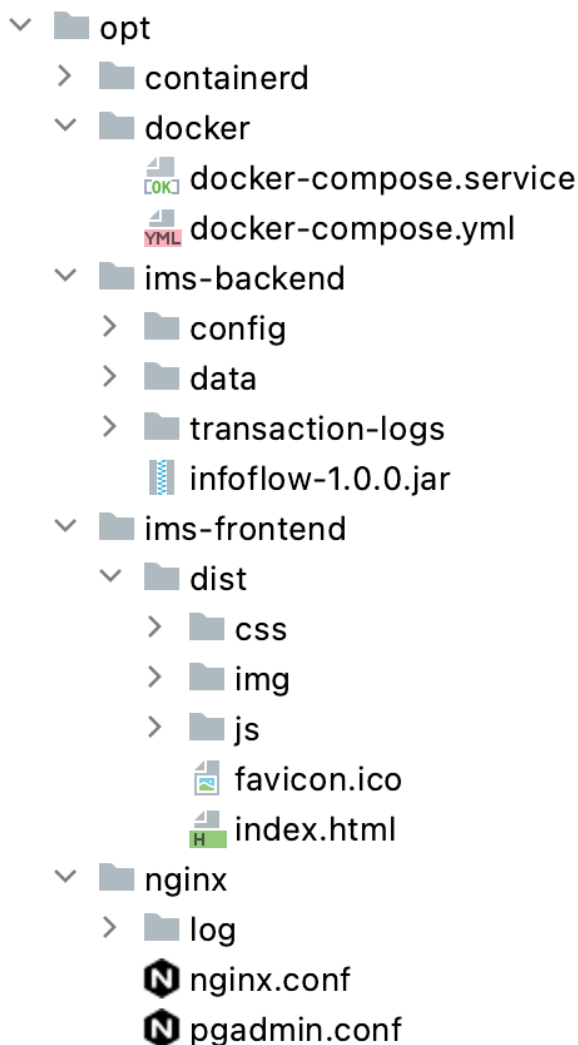


Рисунок 3. Расположение файлов программы

### 6.1. Docker

Запуск программы осуществляется с помощью **docker-compose**. Настройки представлены файле конфигурации `/opt/docker/docker-compose.yml`. Docker-compose запущен с помощью службы **systemd**. Файл настройки службы расположен по адресу `/etc/systemd/system/docker-compose.service`.

### 6.2. Nginx

HTTP-сервер и обратный прокси-сервер, почтовый прокси-сервер, а также TCP/UDP прокси-сервер общего назначения.

Файл конфигурации находится по адресу `/opt/nginx/nginx.conf`.

## 6.3. PostgreSQL

Система управления базой данных

Подключение к базе данных программы:

- Адрес сервера:
  - внешний: `10.0.1.29:5432`
  - внутренний: `postgres:5432`
- База данных: `medural`
- Имя пользователя: `medural`
- Пароль: `inkontextdb`

## 6.4. pgAdmin

**pgAdmin** - платформа для администрирования СУБД PostgreSQL с открытым исходным кодом.

- адрес подключения: `http://10.0.1.29/pgadmin`
- пользователь: `support@inkontext.ru`
- пароль: `inkontextdb`

Подключение к БД с помощью внутреннего адреса сервера (настройки выше).

## 6.5. Java 13

Бакэнд программы разработан на основе java-фреймворка с открытым исходным кодом **Spring Framework 5**.

Программа запущена с помощью **BellSoft Liberica Runtime Environment OpenJDK 13** в docker-контейнере `bellsoft/liberica-openjdk-alpine:13`.

При разработке программы были использованы java-библиотеки с открытым исходным кодом.

Главный jar-файл бакэнда программы расположен по адресу `/opt/ims-backend/infoflow-1.0.0.jar`

## 6.6. Web-клиент

Клиентская часть программы разработана с помощью **Vue.js**, javascript-фреймворка с открытым исходным кодом для создания пользовательских web-интерфейсов.

При разработке программы были использованы javascript-библиотеки с открытым исходным кодом.

Файлы клиентской части программы расположены по адресу `/opt/ims-frontend`.