**Инструкция по установке приложения**

**KPBS Booking**

Москва 2022

[1. Пререквизиты 2](#_Toc1)

[2. Пример подготовки операционной системы к установке приложения 2](#_Toc2)

[3. Установка web-сервера tomcat 2](#_Toc3)

[4. Инициализация базы данных 3](#_Toc4)

[5. Настройка файлового хранилища 4](#_Toc5)

[6. Установка backend-части приложения KPBS Booking 4](#_Toc6)

[7. Установка frontend-части приложения KPBS Booking 4](#_Toc7)

#### Пререквизиты

Минимальные требования для установки приложения KPBS Booking на виртуальную машину:

* Операционная система Ubuntu Server 20.04 LTS
* 2 vCPU
* 4ГБ RAM
* 20ГБ свободного места на жёстком диске

На операционной системе должны быть установлены следующие пакеты:

* openssh-server
* net-tools
* nginx
* php7.4
* php7.4-fpm
* php7.4-pgsql
* php7.4-zip
* php-json
* php-gd
* php-mbstring
* php-curl
* php-xml
* php-pear
* php-cli
* php-fpm
* php-pgsql
* php-zip
* openjdk-8-jdk
* openjdk-8-jre
* postgresql-12
* libapr1-dev
* libssl-dev
* ldap-utils

#### Пример подготовки операционной системы к установке приложения

1. Выполняется установка операционной системы Ubuntu Server 20.04 LTS с минимальным набором приложений и openssh-сервером.
2. При помощи ssh администратор системы входит на сервер и выполняет команды:
sudo apt update
sudo apt upgrade
sudo apt install net-tools nginx php7.4 php7.4-fpm openjdk-8-jdk openjdk-8-jre postgresql-12 php7.4-pgsql php7.4-zip php-json php-gd php-mbstring php-curl php-xml php-pear php-cli php-fpm php-pgsql php-zip libapr1-dev libssl-dev ldap-utils

#### Установка web-сервера tomcat

Одно из основных частей KPBS Booking использует web-сервер tomcat версии 7. Для его установки требуется:

* Скачать web-сервер по ссылке
cd /tmp/
wget <https://archive.apache.org/dist/tomcat/tomcat-7/v7.0.109/bin/apache-tomcat-7.0.109.tar.gz>
* Распаковать
tar xzf apache-tomcat-7.0.109.tar.gz
* Установить
mkdir /usr/local/tomcat/
mv /tmp/apache-tomcat-7.0.109 /usr/local/tomcat/
adduser tomcat
chown -R tomcat:tomcat /usr/local/tomcat
* Прописать параметры запуска в системные переменные в файле
/etc/environment
Необходимо добавить следующие строки
CATALINA\_BASE="/usr/local/tomcat/apache-tomcat-7.0.109"
CATALINA\_HOME="/usr/local/tomcat/apache-tomcat-7.0.109"
CATALINA\_TMPDIR="/usr/local/tomcat/apache-tomcat-7.0.109/temp"
JAVA\_HOME="/usr/lib/jvm/java-8-openjdk-amd64"
* Перезапустить ssh-сессию

#### Инициализация базы данных

В комплекте поставки программного обеспечения KPBS Booking заказчику передаётся дамп базы данных для её инициализации.

Шаги для настройки базы данных:

* Выполнить вход в приложение postgresql 12
sudo su
su - postgres
psql
* Создать базу данных
CREATE DATABASE meeting;
CREATE USER meeting ENCRYPTED PASSWORD '<ввести пароль>';
ALTER DATABASE meeting OWNER TO meeting;
\q
* Инициализировать базу данных из дампа, входящего в комплект поставки
psql meeting < /tmp/meeting\_dump
* Разрешить базе данных подключение с помощью созданного пользователя. Для этого внести изменения в файл /etc/postgresql/12/main/pg\_hba.conf. Пример конфигурации
local all postgres peer
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all md5
# IPv4 local connections:
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local replication all peer
host replication all 127.0.0.1/32 md5
host replication all ::1/128 md5
host meeting meeting 127.0.0.1/32 md5
* Перезапустить базу данных
sudo su
systemctl restart postgresql

#### Настройка файлового хранилища

Приложение хранит загруженные файлы в папке /app/uploads. Необходимо её создать и заполнить. Для этого:

* Создать требуемую папку
mkdir -p /app/uploads
* Заполнить папку начальными изображениями, если такие передавались в комплекте поставки
mv /tmp/uploads/\* /app/uploads/
* Выдать права на папки и файлы
chown -R www-data:www-data /app

#### Установка backend-части приложения KPBS Booking

Для установки backend-части приложения необходимо:

* Переместить файл mrbackend.war, входящий в комплект поставки, и перезапустить web-сервер tomcat
mv /tmp/mrbackend.war /usr/local/tomcat/apache-tomcat-7.0.109/webapps/
cd /usr/local/tomcat/apache-tomcat-7.0.109/bin
./startup.sh

#### Установка frontend-части приложения KPBS Booking

Для установки frontend-части необходимо:

* Скопировать содержимое папки html, входящей в комплект поставки, в корневую папку web-сервера nginx:
mv /tmp/html/\* /var/www/html/
chown -R www-data:www-data /var/www/html
* В файле /var/www/html/app/connect/meeting.php внести изменения в строки
$host="127.0.0.1";
$db\_username="meeting";
$db\_password="\*\*\*";
$db\_name="meeting";
указав в кавычках соответствующие значения для адреса базы данных, пользователя базы данных, пароля базы данных и имени баз данных, вводимых в предыдущих шагах по установке приложения.
* Удалить конфигурационные файлы сайтов web-сервера nginx
rm /etc/nginx/sites-available/\*
rm /etc/nginx/sites-enabled/\*

Создать новый конфигурационный файл сайта /etc/nginx/sites-available/meeting, прописав в нём доступы к frontend и backend-частям.
server {
 listen 80 default\_server;
 listen [::]:80 default\_server;
 root /var/www/html;
 index index.html index.htm index.nginx-debian.html;
 server\_name \_;
 location / {
 try\_files $uri $uri/ =404;
 }
 location ~ ^/(rooms|places|admin|login|reset-password|register|password) {
 try\_files $uri $uri/ /index.html;
 }
 location /mrbackend {
 proxy\_pass [http://localhost:8080/mrbackend;](http://localhost:8080/mrbackend)
 }

 location ~ [^/]\.php(/|$) {

 fastcgi\_split\_path\_info ^(.+?\.php)(/.\*)$;

 if (!-f $document\_root$fastcgi\_script\_name) {

 return 404;

 }

 fastcgi\_pass unix:/var/run/php/php7.4-fpm.sock;

 fastcgi\_param HTTP\_PROXY "";

 include snippets/fastcgi-php.conf;

 }

}

* Активировать сайт
ln -s /etc/nginx/sites-available/meeting /etc/nginx/sites-enabled/meeting
* Удалить старые логи, если такие есть
rm /var/www/html/app/meeting/logs/log.txt
* Перезапустить nginx
systemctl restart nginx
* Приложение будет доступно по ссылке
http://<адрес сервера>/