

Инструкция по установке ПО “апп321”

Оглавление

<u>1. Архитектура решения и технические требования</u>	<u>3</u>
<u>1.1. Описание архитектуры</u>	<u>3</u>
<u>1.2. Технические требования</u>	<u>3</u>
<u>1.2.1. Требования к серверу базы данных</u>	<u>3</u>
<u>1.2.2. Требования к серверу приложения</u>	<u>3</u>
<u>1.2.3. Требования к тонкому клиенту (браузер)</u>	<u>3</u>
<u>1.2.4. Требования для решения с использованием балансировщика</u>	<u>3</u>
<u>2. Поставка</u>	<u>4</u>
<u>3. Установка и настройка компонентов приложения</u>	<u>4</u>
<u>3.1. База данных</u>	<u>4</u>
<u>3.2. Приложение backend</u>	<u>4</u>
<u>3.2.1. Установка Java JRE</u>	<u>4</u>
<u>3.2.2. Распаковка приложения app321</u>	<u>4</u>
<u>3.2.3. Запуск приложения</u>	<u>1</u>

1. Архитектура решения и технические требования

1.1. Описание архитектуры

Приложение состоит из компонентов:

- Сервер приложения - сервис для обработки запросов пользователей и связи с сервером базы данных;
- Сервер базы данных - обеспечивает хранение данных;
- Тонкий клиент – работа в браузере.

С точки зрения безопасности, надежности, масштабирования приложения рекомендуется разделять сервер приложений и сервер базы данных.

Приложение поддерживает установку нескольких серверов приложений для распределения нагрузки через сервер-балансировщик.

1.2. Технические требования

1.2.1. Требования к серверу базы данных

Программное обеспечение:

- Программное обеспечение: PostgreSQL от версии 16 - рекомендуется устанавливать только для ОС на базе Linux;
- Инструмент для работы администратора с базами данных: pgAdmin, DBeaver или аналогичные.

Аппаратное обеспечение:

- Процессор: 2 CPU;
- Оперативная память: 16Gb RAM;
- Объем дисков: от 1 Гб;
- Канал связи: от 100 Mbit/sec, при кол-ве пользователей более 50 рекомендуется от 1000 Mbit/sec.

1.2.2. Требования к серверу приложения

Программное обеспечение:

- Linux;
- Windows.

Аппаратное обеспечение:

- Процессор: 4 CPU;
- Оперативная память: 8Gb RAM;
- Объем дисков: от 1 Гб;
- Канал связи: от 100 Mbit/sec, при кол-ве пользователей более 50 рекомендуется от 1000 Mbit/sec.

1.2.3. Требования к тонкому клиенту (браузер)

Программное обеспечение:

- GoogleChrome;
- FireFox;

1.2.4. Требования для решения с использованием балансировщика

Программное обеспечение:

- Nginx

2. Поставка

Будет получен архив app321-1.0-SNAPSHOT.zip с сервером приложения.

3. Установка и настройка компонентов приложения

3.1. База данных

Создать базу данных и пользователя.

3.2. Приложение backend

3.2.1. Установка Java JRE

Для корректной работы приложения необходимо иметь установленную Java SE версии 11 или позднее.

3.2.2. Распаковка приложения app321

Распаковать архив app321-1.0-SNAPSHOT.zip

В распакованной папке скопировать файл conf/application.conf.example в conf/application.conf и настроить параметры, указанные в этом файле:

```
# параметры подключения к БД
slick.dbs.default.profile="slick.jdbc.PostgresProfile$"
slick.dbs.default.db.driver="org.postgresql.Driver"
slick.dbs.default.db.url="jdbc:postgresql://localhost/app321"
slick.dbs.default.db.username=""
slick.dbs.default.db.password=""

# путь к каталогу для сохранения файлов
file.storage.path=""

# настройка подключения к SMTP-серверу для отправки почты
play.mailer {
  host = "example.com" # (обязательный параметр - либо указать
mock = yes)
  port = 25 # (по умолчанию 25)
  ssl = no # (по умолчанию no)
  tls = no # (по умолчанию no)
  tlsRequired = no # (по умолчанию no)
  user = null # (не обязательно)
  password = null # (не обязательно)
  debug = no # (по умолчанию no)
  timeout = null # (в миллисекундах, по умолчанию 60 сек)
  connectionTimeout = null # (в миллисекундах, по умолчанию 60
сек)
  mock = no # (по умолчанию no, только писать сообщения в лог
вместо отправки почты)
}
```

```

# имя отправителя в исходящей почте
mail.from = ${views.variables.appTitle}"
<noreply@app321.example.com>"

# URL-адрес этого сайта (подставляется в исходящей почте)
views.variables.baseUrl = "http://localhost:9000"

# секретный ключ приложения
# должен быть одинаковым на нодах (при использовании
балансировщика)
# может быть сгенерирован командой `sbt playGenerateSecret` или
`head -c 32 /dev/urandom | base64`.
play.http.secret.key="changeme"

# адреса, по которым данное приложение будет доступно
play.filters.hosts {
  allowed = ["localhost:9000", "localhost:9443", "127.0.0.1:9000"]
}

# установить демо-пользователей
liquibase.contexts = "demo"

```

Скопировать файл `conf/logback.xml.file.example` в `conf/logback.xml`

3.2.3. Запуск приложения

Для запуска приложения в качестве сервиса создайте файл `/etc/systemd/system/app321.service`

Содержимое файла (указать актуальный `/path/to/app321...`):

```

[Unit]
Description=Service for app321
After=network.target

[Service]
ExecStart=/bin/bash -c '/path/to/app321-1.0-SNAPSHOT/bin/app321
-Dconfig.file=/path/to/app321-1.0-SNAPSHOT/conf/application.conf
-Dhttp.port=9000'
Restart=on-failure
ExecStopPost=/bin/bash -c 'rm -rf
/path/to/app321-1.0-SNAPSHOT/RUNNING_PID'
ExecStartPre=/bin/bash -c 'rm -rf
/path/to/app321-1.0-SNAPSHOT/RUNNING_PID'

[Install]

```

WantedBy=multi-user.target

Для запуска сервиса используйте команду
`service app321 start`

Для остановки сервиса используйте команду
`service app321 stop`

3.2.4 Настройка сертификата

Чтобы обеспечить безопасное соединение, должен быть настроен сертификат. Если запросы от пользователей поступают на фронтенд сервер, такой как nginx, тогда необходимо указать настройки в этом сервере. Иначе добавьте настройки сертификата в `application.conf` приложения `app321`:

```
https.port=443
http.port=disabled
play.server.https.keyStore {
  path = "/path/to/app321-server.jks" # Путь до файла с хранилищем
  # приватного ключа и сертификата; если не указан, хранилище будет
  # создано автоматически
  password = "пароль хранилища" # по умолчанию пустой пароль
  # type - тип хранилища, по умолчанию JKS
  # algorithm - алгоритм хранилища, по умолчанию зависит от платформы
}
```