# БАРС.Бюджет – Кадры

Описание технической архитектуры

# Содержание

1	Уст	ановка	а и настройка	6		
		1.1.1	Установка СУБД PostgreSQL и создание БД приложения	6		
		1.1.2	Установка ПО «БАРС.Бюджет - Кадры»	8		
		1.1.3	Установка Supervisor	12		
		1.1.4	Установка Gunicorn	15		
		1.1.5	Установка Nginx	17		
		1.1.6	Установка RabbitMQ	19		
		1.1.7	Установка Redis	21		
	1.2	Запус	к Системы	21		
	1.3	Прове	ерка работоспособности Системы	22		
	1.4	.4 Создание резервной копии базы данных				
	1.5	Восст	ановление БД из резервной копии	24		
		1.5.1	Алгоритм установки обновления с возможностью отката	24		
		1.5.2	Установка обновления с возможностью отката	25		
2	Ава	арийнь	ые ситуации	29		
		2.1.1	Сообщение браузера об ошибке	29		
		2.1.2	Ошибка авторизации	29		
		2.1.3	Восстановление БД из резервной копии	29		

#### Аннотация

В данном документе приводится необходимая информация по установке и настройке компонентов ПО «БАРС.Бюджет - Кадры» на операционной системе Centos 7 х86\_64.

**Примечание.** Для других Linux систем команды схожи, но могут отличаться.

Перечень терминов и сокращений

Обозначение	Описание			
БД	База данных			
Репозиторий	Хранилище файлов, доступных для дальнейшего распространения по сети			
СУБД Система управления базой данных				

# 1 Установка и настройка

Установка компонентов ПО «БАРС.Бюджет - Кадры» и всех его составляющих выполняется с помощью команд в рабочей консоли (Рис. 1), которую можно открыть, например, с помощью нажатия клавиш Ctrl+Alt+F1:



Рис. 1. Рабочая консоль

# 1.1.1 Установка СУБД PostgreSQL и создание БД приложения

# Установка PostgreSQL

Для установки СУБД PostgreSQL 13 из репозитория необходимо в рабочей консоли выполнить следующую команду:

# # yum install postgresql13-server

**Примечание.** Если в репозиториях нет установочного пакета postgresql133-server, то СУБД PostgreSQL 13 следует установить из источника ниже:

http://wiki.postgresql.org/wiki/YUM\_Installation

# Настройка PostgreSQL

Для корректной работы модуля, необходимо в папке с установленной СУБД PostgreSQL в файле *postgresql.conf* найти и исправить параметр *datestyle*:

datestyle = 'iso, dmy'

Создание базы данных и пользователя для системы

Необходимо войти в консоль управления PostgreSQL:

# su postgres -c 'psql'

Создать пользователя с паролем:

create user bars\_web\_bb password 'bars\_web\_bb';

Создать базу данных и указать созданного пользователя ее владельцем:

create database bars\_web\_bb owner bars\_web\_bb;

Выйти из консоли PostgreSQL:

\q

Загрузить первоначальные данные в базу данных из резервной копии:

# su postgres

\$ pg\_restore --username postgres --dbname bars\_web\_bb web\_bb\_demo.backup

# 1.1.2 Установка ПО «БАРС.Бюджет - Кадры»

#### Установка необходимых пакетов

Для установки пакетов необходимо выполнить следующие команды:

# yum install python39 python39-devel python39-pip libxml2-devel libxslt-devel libjpegdevel freetype-devel ncurses-devel readline-devel openssl-devel openIdap-devel xz xzlibs xz-devel zlib-devel bzip2 bzip2-devel unzip sqlite sqlite-devel libffi-devel cacertificates nss nss-util openssl gcc gcc-c++ make passwd xmlsec1-devel xmlsec1openssl-devel libpq-devel git vim-enhanced wget telnet sudo postgresql13 postgresql13devel libtool-ltdl-devel

# python -m pip install --no-cache-dir --upgrade pip

# pip install virtualenv

В переменную РАТН необходимо добавить путь до бинарных файлов Postgres:

export PATH=\$PATH:/usr/pgsql-13/bin/

## Установка системы

Веб-сервер и Supervisor будет запускаться от пользователя *www-data*, если пользователя *www-data* нет в системе, то его необходимо создать:

# useradd www-data -- U -d /home/venv

Выставить права на его домашний каталог:

# chmod -R 774 /home/venv

Создать пользователя *celery*, от его имени будут запускаться фоновые процессы:

useradd celery -M -G www-data

Создать виртуальное окружение от пользователя *www-data* и, в качестве интерпретатора указать установленный Python3.9:

# su www-data -c 'virtualenv --setuptools --no-site-packages -p /usr/bin/python3.9 /home/venv/web\_bb'

Создать папку, где будут храниться конфигурации приложения, и присвоить ей права пользователя *www-data*.

Важно! Адрес папки должен быть строго: /var/www/web\_bb\_config.

# mkdir -p /var/www/web\_bb\_config

# chown www-data:www-data /var/www/web\_bb\_config

От имени пользователя www-data активировать виртуальное окружение командой:

# su www-data

\$ source /home/venv/web\_bb/bin/activate

Распаковать зависимости в отдельную папку:

(web\_bb)\$ mkdir /tmp/web\_bb\_requirements

(web\_bb)\$ cp web\_bb\_requirements.tar.gz /tmp/web\_bb\_requirements

(web\_bb)\$ cd /tmp/web\_bb\_requirements

(web\_bb)\$ tar -xvzf web\_bb\_requirements.tar.gz

Установить приложение с помощью pip:

(web\_bb)\$ pip install web\_bb.tar.gz --no-index --find-link /tmp/web\_bb\_requirements

После окончания установки приложения необходимо поправить конфигурационный файл, который находится по адресу: /var/www/web\_bb\_config/project.conf.example и переименовать его в project.conf. Если рабочее приложение уже было установлено ранее, необходимо перенести его project.conf в папку /var/www/web\_bb\_config/.

Далее необходимо выполнить миграции. Для этого необходимо определить переменные окружения:

# su www-data

\$ source /home/venv/web\_bb/bin/activate

(web\_bb)\$ export WEB\_BB\_CONF=/var/www/web\_bb\_config/project.conf

(web\_bb)\$ export DJANGO\_SETTINGS\_MODULE = web\_bb.settings

Запустить скрипт migrate.pyc:

(web\_bb)\$ python /home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/manage.pyc migrate

Если возникают проблемы с аутентификацией в БД, например, такие – FATAL: Ident authentication failed for user "bars\_web\_bb" – необходимо подкорректировать файл  $pg_hba.conf$ , обычно он находится здесь: /var/lib/pgsql/9.1/data/ (править этот файл может либо пользователь root, либо postgres).

В секцию local следует дописать:

# TYPE DATAE	BASE USER	ADDRESS	METHOD
# "local" is for U	nix domain socket coni	nections only	
local all	bars_web_bb	md5	

Для paбoты supervisor и самого приложения необходимо создать лог-файлы. Для этого вручную необходимо создать файлы логов и выставить им права, например, так:

# touch /var/log/web\_bb/{celery,main}.log

# chown www-data:www-data /var/log/web\_bb/{celery,main}.log

# chmod 774 /var/log/web\_bb/

Если потребуется настроить supervisor – готовый конфигурационный файл находится в /var/www/web\_bb\_config/supervisord.conf.example

# Структура проекта web\_bb

После установки приложения с помощью pip будет следующая структура:

/home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/ – здесь будет лежать приложение.

/var/log/web\_bb/ – сюда приложение будет писать логи.

/var/www/web\_bb\_config – сюда будут скопированы примеры конфигурационных файлов для supervisor, gunicorn и project.conf.example для самого приложения.

## Пробный запуск

Запуск приложения:

export WEB\_BB\_CONF=/var/www/web\_bb\_config/project.conf

(web\_bb)\$ python /home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/manage.pyc runserver 0.0.0.0:8000

Проверка работоспособности в браузере по адресу: http://aдрес-этого-сервера:8000

## Статические файлы

Для сбора статики необходимо выполнить команду:

export WEB\_BB\_CONF=/var/www/web\_bb\_config/project.conf

(web\_bb)\$ python /home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/manage.pyc collectstatic -noinput

Можно заранее определить папки с пользовательской статикой. Для этого, нужно дописать в *project.conf*:

[media]

ROOT = /var/www/web\_bb\_config/share/media

[downloads] ROOT = /var/www/web\_bb\_config/share/downloads [temporary] ROOT = /var/www/web\_bb\_config/share/tmp [static] ROOT = /var/www/web\_bb\_config/share/static

# 1.1.3 Установка Supervisor

Установить Supervisor:

# pip install supervisor

Создать файл конфигураций в supervisor:

# echo\_supervisord\_conf > /etc/supervisord.conf

В папке /etc/ появится файл конфигурации *supervisord.conf*, необходимо открыть его и изменить следующие строки:

chmod=0700	; socket file mode (default 0700)
chown=www-data:www-data	; socket file uid:gid owner

В нижней части файла конфигурации supervisord.conf следует добавить строки:

[include]

files = /etc/supervisor/conf.d/\*.conf

Создать папку *supervisor* и в ней *conf.d* – в этой папке будут храниться файлы конфигураций для запуска приложений:

# mkdir -p /etc/supervisor/conf.d

Скопировать в папку файл конфигурации для проекта *web\_bb.conf* или его создать:

# cp /var/www/web\_bb\_config/supervisord.conf.example
/etc/supervisor/conf.d/web\_bb.conf

Файл конфигурации проекта содержит настройки запуска проекта:

web\_bb.conf

[program:web\_bb]

#Переменные окружения для проекта.

environment=PATH="/home/venv/web\_bb/bin/",WEB\_BB\_CONF="/var/www/web\_bb\_co nfig/project.conf"

#команда запуска

command=/home/venv/web\_bb/bin/python manage.pyc -c /var/www/web\_bb\_config/gunicorn.conf.py

#Директория, в которой будет выполняться программа.

directory=/home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/

#Пользователь и группа, от имени который будет запускаться приложение

user=www-data

group=www-data

daemon=False

debug=False

autostart=true

autorestart=true

redirect\_stderr=True

stdout\_logfile=/var/log/web\_bb/main.log

[program:web\_bb\_celery]

environment=PATH="/home/venv/web\_bb/bin/",

WEB\_BB\_CONF="/var/www/web\_bb\_config/project.conf"

command=/home/venv/web\_bb/bin/python manage.pyc celery worker -l info

directory=/home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/

user=celery

group=www-data

numprocs=1

```
autostart=true
```

autorestart=true

startsecs=10

redirect\_stderr=true

stopwaitsecs = 600

stdout\_logfile=/var/log/web\_bb/celery.log

[program:pushme\_server]

command=python -m pushme.server -p 9999

user=www-data

group=www-data

autostart=true

autorestart=true

startsecs=10

```
redirect_stderr=true
```

stdout\_logfile=/var/log/web\_bb/pushme\_server.log

[program:pushme\_queue]

command=python -m pushme.queue -p 4000

user=www-data

group=www-data

autostart=true

autorestart=true

startsecs=10

redirect\_stderr=true

stdout\_logfile=/var/log/web\_bb/pushme\_queue.log

#### Автозагрузка Supervisor

Добавление supervisor в автозагрузку зависит от дистрибутива Linux. На сайте <u>supervisor</u> есть все необходимые конфигурационные файлы.

#### 1.1.4 Установка Gunicorn

Важно! В данном случае команды выполняются в виртуальном окружении.

# su www-data

\$ source /home/venv/web\_bb/bin/activate

(web\_bb)\$ pip install gunicorn

По умолчанию, после установки приложения, файл настройки для *gunicorn* будет находиться по адресу: /var/www/web\_bb\_config/gunicorn.conf.py

Файл gunicorn.conf.py содержит параметры запуска приложения:

#coding: utf-8

import os, multiprocessing

short\_name = "web\_bb"

run\_dir = "/var/tmp"

log\_dir = "/var/log/web\_bb"

bind = "unix:%s/%s.sock" % (run\_dir, short\_name)

proc\_name = "%s/%s" % (run\_dir, short\_name)

workers = multiprocessing.cpu\_count() \* 2 + 1

user = "www-data"

group = "www-data"

errorlog = "%s/gunicorn.log" % log\_dir

loglevel = "info"

pidfile = "%s/%s.pid" % (run\_dir, short\_name)

timeout = 3000

#### Запуск службы Supervisor

Для запуска службы Supervisor следует выполнить команду:

# /etc/init.d/supervisor start

или

# /usr/local/bin/supervisord

Необходимо проверить, что приложения запущены:

# supervisorctl status

Приложения должны оказаться в статусе RUNNING:

celery_web_bb	RUNNING
web_bb	RUNNING

Если приложения не запустились, то следует обратиться к разработчикам системы, приложив файлы *error.log* и *gunicorn.log* журнала ошибок из папки /var/www/web\_bb/log/.

# 1.1.5 Установка Nginx

#### Установка необходимых пакетов

В Centos Nginx можно установить из репозитория epel или из официального репозитория nginx. Установим из официального репозитория:

# wget http://nginx.org/packages/centos/6/noarch/RPMS/nginx-release-centos-6-0.el6.ngx.noarch.rpm

# rpm -ivh nginx-release-centos-6-0.el6.ngx.noarch.rpm

# yum install nginx

## Настройка Nginx

Копирование или создание файла web\_bb.nginx.conf в каталог /etc/nginx/conf.d:

# cp /var/www/web\_bb\_config/nginx.conf.example /etc/nginx/conf.d/web\_bb.nginx.conf

Ниже приведен пример виртуального хоста nginx:

Важно! Для корректной работы виртуального хоста nginx – в конфигурационном фале /etc/nginx/nginx.conf должна быть раскомментирована строка: include /etc/nginx/conf.d/\*.conf. В этой директории создаем файл web\_bb.nginx.conf со следующим содержимым

```
upstream web_bb_server {
```

```
server unix:/var/run/web_bb/web_bb.sock fail_timeout=0;
```

}

```
server {
```

```
listen *:80;
```

keepalive\_timeout 65;

access\_log /var/log/nginx/nginx\_access.log custom;

```
error_log /var/log/nginx/nginx_error.log;
```

```
location / {
```

```
if ( -e /var/www/html/servicework ) {
```

return 503;

```
}
```

proxy_redirect	off;				
proxy_set_header	Host \$host;				
proxy_set_header	X-Url-Scheme \$scheme;				
proxy_set_header	X-Host \$http_host;				
proxy_set_header	X-Real-IP \$remote_addr;				
proxy_set_header	X-Forwarded-For \$proxy_add_x_forwarded_for;				
proxy_set_header	Proxy-host \$proxy_host;				
client_max_body_siz	e 400m;				
client_body_buffer_size 128k;					
proxy_buffering	off;				

```
proxy_connect_timeout 10800;
    proxy_send_timeout
                            10800;
    proxy_read_timeout
                            10800;
    proxy_buffers
                         8 32k;
    add_header PR-B-TOTAL $request_time;
    proxy_pass http://web_bb_server;
  location /protected/downloads/ {
    alias /var/www/web_bb_config/share/downloads/;
    internal;
  }
  location /protected/media/ {
    alias /var/www/web_bb_config/share/media/;
    internal;
  }
  }
}
```

#### 1.1.6 Установка RabbitMQ

#### Установка необходимых пакетов

Необходимо установить пакет rabbitmq-server можно из репозитория epel (его нужно подключить) или скачать с официального сайта разработчиков. Для того чтобы установить пакет из репозитория epel необходимо выполнить следующие команды:

Установить репозиторий и пакет:

# su -c 'rpm -Uvh http://download.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm'

# yum install rabbitmq-server

Запустить сервер rabbitmq:

# /etc/init.d/rabbitmq-server start

Добавить сервер rabbitmq в автозагрузку:

# chkconfig rabbitmq-server on

## Настройка rabbitmq

Создать пользователя:

# rabbitmqctl add\_user web\_bb web\_bb

Создать хост:

# rabbitmqctl add\_vhost web\_bb

Настроить права на хост:

# rabbitmqctl set\_permissions -p web\_bb web\_bb ".\*" ".\*"

В файле настроек приложения *project.conf* необходимо указать хост и пользователя для подключения к RabbitMQ, а также включить фоновые процессы:

[waterfall]

WTF\_ASYNC = True

CELERY\_BROKER\_URL = amqp://web\_bb:web\_bb@localhost:5672/web\_bb

## Перезапуск проекта

Перезапустить приложение:

# supervisorctl restart celery\_web\_bb

# supervisorctl restart web\_bb

#### 1.1.7 Установка Redis

Установить Redis командой:

# yum install redis -y

Запустить Redis:

# systemctl start redis.service

# systemctl enable redis

Отредактировать project.conf, добавив в раздел [runtime] строчку:

[redis]

REDIS\_HOST = redis://0.0.0.0:16379/2

#### 1.2 Запуск Системы

Для начала работы необходимо выполнить следующую последовательность действий:

• запустить интернет-обозреватель двойным нажатием левой кнопки мыши по его ярлыку на рабочем столе или нажать на кнопку «Пуск» и в открывшемся меню выбрать пункт, соответствующий используемому интернет-обозревателю;

- в интернет-обозревателе в адресной строке ввести адрес: http://<aдpec\_cepвepa>;
- в окне идентификации ввести логин и пароль и нажать на кнопку «Вход» (Рис. 2);

ЛОГИН		
Пароль		
		Вход
	БАРС Груп © 2019	

Рис. 2. Кнопка «Вход»

• Откроется главное окно Модуля (Рис. 3):

Кадры Приказы Расчеты Зарплата Налоги Отчеть	ты Справочники Администрирование	Тестовое ЛПУ
БАРС.Бюджет Кадры		CerogH4: 09.11.2023 22:43
		A
		Добро пожаловать 🔅 引
		Ушакова Александра
Штатная Набор выплат Журни расстановка сотрудника Курни	ал невьходов Персонфицирова учет утворждени утакной расстановки	
		Рабочая дата: 🕮 🖉
		10.11.2023
		Рабочий период:
		C 01.01.2023
		По 31.12.2023 👜
		Рабочее учреждение: 🔳
		Тестовое ЛПУ ИН: 74020300 КПП: 74101001 1420
		Есть новые сообщения
		Уведомления:
		Нет уведомлений
		Загрузки: Удллить

Рис. 3. Главное окно Модуля

## 1.3 Проверка работоспособности Системы

Система считается работоспособной, если в результате действий пользователя, изложенных в <u>п.2.2</u>, на экране монитора отобразилось главное окно без выдачи пользователю сообщений о сбое в работе.

# 1.4 Создание резервной копии базы данных

Настройка резервного копирования

Для снятия резервной копии одной базы данных в PostgreSQL предусмотрена утилита *pg\_dump*. Она позволяет снимать копию всей базы или же каких-либо таблиц выборочно. Подробная справка по этой утилите доступна на официальном <u>сайте</u>. В общем случае, для снятия резервной копии команда будет выглядеть так:

pg\_dump -h 10.0.0.10 -U web\_bb -F c -f /home/bars/web\_bb.backup bars\_web\_bb

Формат команды:

pg\_dump [ОПЦИИ]... [ИМЯ БД]

#### Расшифровка:

-*h* 10.0.0.10 – указывает, что необходимо соединиться с сервером по адресу 10.0.0.10 $\cup$  -*U* web\_bb – хотим сделать это от имени пользователя web\_bb

-*F c* – будем использовать обычный формат  $pg\_dump$ , существуют следующие форматы – c|d|t|p (custom, directory, tar, plain text)

*-f /home/bars/web\_bb.backup* – куда положим нашу резервную копию.

*bars\_web\_bb* – имя нашей базы данных.

#### Автоматическое создание резервных копий

Обычно снятие резервных копий настраивают на ночное время. Для этого в \*nix системах используется планировщик задач *cron*. Задачи для выполнения описываются в файле *crontab*. Отредактировать его можно командой: crontab -e.

Файл будет выглядеть примерно так:

# m h dom mon dow command23 01 \* \* \* /home/backup/backup\_postgres

Таблица *crontab* состоит из 6 колонок, разделяемых пробелами или табуляторами. Первые пять колонок задают: время выполнения (Минута, Час, День, Месяц, День недели), в них может находиться число, список чисел, разделённых запятыми, диапазон чисел, разделённых тире или символ '\*'. Все остальные символы в строке интерпретируются, как выполняемая команда с её параметрами.

#### 1.5 Восстановление БД из резервной копии

#### 1.5.1 Алгоритм установки обновления с возможностью отката

Чтобы обновляться безопасно и иметь возможность откатиться назад, нужно действовать по следующему алгоритму:

- остановить приложение;
- снять дамп с БД;
- сделать копию приложения:
  - о 1-ый вариант: скопировать папку с виртуальным окружением;
  - о 2-ой вариант: достать пакет с предыдущей версией сборки.
- установить приложение в виртуальное окружение;
- провести миграции.

## Откат:

- остановить приложение.
- установить предыдущую версию приложения:
  - 1-ый вариант: удалить папку с виртуальным окружением и вернуть на место папку со старым окружением;
  - 2-ой вариант: удалить приложение из виртуального окружения и установить пакет с предыдущей версией.

## Если повреждена БД:

- остановить приложение;
- переименовать текущую базу (на всякий случай);
- создать новую чистую базу;
- развернуть в новую базу резервную копию;
- запустить приложение.

#### 1.5.2 Установка обновления с возможностью отката

Остановить приложение в Supervisor:

# supervisorctl web\_bb stop

# supervisorctl web\_bb\_celery stop

Снять резервную копию с базы данных утилитой *pg\_dump*. В общем случае команда будет выглядеть так:

\$ pg\_dump -U bars\_web\_bb -F c -f /home/bars/web\_bb.backup bars\_web\_bb

Сделать копию приложения:

1-ый вариант: скопировать папку с виртуальным окружением:

# cp -rp /home/venv/web\_bb /home/venv/web\_bb\_copy

2-ой вариант: достать пакет с предыдущей версией сборки.

Активировать виртуальное окружение:

# su www-data

\$ source /home/venv/web\_bb/bin/activate

В каталоге с архивом обновления, запустить команду:

(web\_bb)\$ export WEB\_BB\_CONF=/var/www/web\_bb\_conf/project.conf

(web\_bb)\$ pip install --upgrade web\_bb.tar.gz

Затем запустить миграции:

(web\_bb)\$ export WEB\_BB\_CONF=/var/www/web\_bb\_conf/project.conf

(web\_bb)\$ export DJANGO\_SETTINGS\_MODULE = web\_bb.settings

(web\_bb)\$ django\_admin.py migrate

Так же, можно запустить миграции явно:

(web\_bb)\$ export WEB\_BB\_CONF=/var/www/web\_bb\_conf/project.conf

(web\_bb)\$ python /home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/manage.pyc migrate

Собрать статику проекта:

(web\_bb)# python /home/venv/web\_bb/lib/python3.9/site-packages/web\_bb/manage.pyc collectstatic -noinput

Запустить приложение:

# supervisorctl web\_bb start

# supervisorctl web\_bb\_celery start

#### Откат

Если приложение не запустилось или произошли какие-то ошибки, то нужно остановить приложение в Supervisor:

# supervisorctl web\_bb stop

# supervisorctl web\_bb\_celery stop

Установить предыдущую версию приложения:

**1-ый вариант:** удалить папку с виртуальным окружением и вернуть на место папку со старым окружением:

# rm -rf /home/venv/web\_bb

# mv /home/venv/web\_bb\_copy /home/venv/web\_bb

**2-ой вариант:** удалить приложение *web\_bb* из виртуального окружения и установить пакет с предыдущей версией:

(web\_bb)\$ pip uninstall web\_bb

(web\_bb)\$ pip install web\_bb\_old\_version.tar.gz --upgrade

Запустить приложение:

# supervisorctl web\_bb start

# supervisorctl web\_bb\_celery start

## Если повреждена БД

Если после обновления/отката приложения корректность состояния БД была нарушена либо пропали какие-то данные, то следует восстановить БД из резервной копии. Для этого нужно остановить приложение в Supervisor.

# supervisorctl web\_bb stop
# supervisorctl web\_bb\_celery stop

Для резервного копирования необходимо переименовать старую базу данных, после чего создать новую чистую базу, в которую мы развернем резервную копию.

ALTER DATABASE bars\_web\_bb RENAME TO bars\_web\_bb\_old;

CREATE DATABASE bars\_web\_bb WITH OWNER bars\_web\_bb;

Воспользоваться утилитой *pg\_restore*. В общем случае команда будет выглядеть так:

\$ pg\_restore -F c -d bars\_web\_bb /home/bars/web\_bb.backup

Формат команды:

```
pg_restore [ОПЦИИ]... [ФАЙЛ]
```

Опции:

d-имя БД;

F – формат дампа (обычно определяется автоматически);

Запустить приложение:

# supervisorctl web\_bb start

# supervisorctl web\_bb\_celery start

# 2 Аварийные ситуации

# 2.1.1 Сообщение браузера об ошибке

При возникновении ошибок в браузере (не загружается страница авторизации, главная страница, отображается ошибка с кодом 403-407) необходимо проверить целостность подключения APM пользователя к сети internet.

При возникновении ошибок в браузере (отображается ошибка с кодом 500-510) необходимо попробовать войти в систему позже. Если ошибка возникает при повторных попытках авторизации в течение длительного времени, необходимо обратиться в службу технической поддержки.

# 2.1.2 Ошибка авторизации

При возникновении сообщения «Вход в систему невозможен» при попытке авторизации необходимо проверить правильность вводимых логина и пароля. Если правильность вводимых данных не вызывает сомнений, необходимо обратиться в службу технической поддержки.

## 2.1.3 Восстановление БД из резервной копии

При возникновении ошибок, влияющих на целостность данных, следует восстановить БД из резервной копии.

Лист регистрации изменений									
	Номера листов (страниц)				Всего		Входящий		
Изм.	изменен- ных	заменен- ных	новых	аннулиро- ванных	листов (страниц) в доку- менте	Номер доку- мента	номер сопроводи- тельного документа и дата	Под- пись	Дата