**Санкт‑Петербургское государственное унитарное предприятие**

**«Санкт‑Петербургский информационно‑аналитический центр»**

**(СПб ГУП «СПб ИАЦ»)**

ПРОГРАММА ДЛЯ ЭВМ

**«СТАТИСТИКА И АНАЛИТИКА ГОСУДАРСТВЕННЫХ   
И МУНИЦИПАЛЬНЫХ УСЛУГ»**

РУКОВОДСТВО АДМИНИСТРАТОРА

На 11 листах

Содержание

Перечень принятых сокращений и обозначений 3

1 Общие сведения о программе 4

1.1 Назначение программы 4

1.2 Функции программы 4

1.3 Минимальный состав технических средств 4

1.4 Минимальный состав программных средств 4

2 Структура программы 6

2.1 Сведения о структуре программы 6

3 Настройка программы 7

4 Проверка программы 8

4.1 Проверка технического состояния программы 8

4.1.1 Сервер базы данных 8

4.1.2 Сервер приложений 8

4.1.3 Прокси‑сервер 8

4.1.4 PHP FPM 9

4.2 Проверка установленного ПО 9

5 Дополнительные возможности 10

5.1 Процедура запуска/останова программы 10

5.1.1 Процедура запуска сервера базы данных 10

5.1.2 Процедура останова сервера базы данных 10

5.1.3 Процедура запуска сервера приложений 10

5.1.4 Процедура останова сервера приложений 10

5.1.5 Процедура запуска прокси‑сервера 11

5.1.6 Процедура останова прокси‑сервера 11

5.2 Ограничение доступа на время проведения регламентного обслуживания или устранения критических ошибок 11

### Перечень принятых сокращений и обозначений

В настоящем документе применены следующие сокращения (обозначения):

| **№ п/п** | **Сокращение (обозначение)** | **Значение сокращения (обозначения)** |
| --- | --- | --- |
| **1** | **2** | **3** |
|  | АРМ | Автоматизированное рабочее место |
|  | БД | База данных |
|  | ОЗУ | Оперативное запоминающее устройство |
|  | ОС | Операционная система |
|  | ПО | Программное обеспечение |
|  | СУБД | Система управления базами данных |
|  | ЦПУ | Центральное процессорное устройство |
|  | SSH | Secure Shell (англ.) – сетевой протокол прикладного уровня, позволяющий производить удалённое управление операционной системой |

# Общие сведения о программе

## 1.1 Назначение программы

Программа предназначена для централизованного сбора, анализа, хранения и предоставления статистических данных о предоставляемых органами власти услугах региона.

## 1.2 Функции программы

Основные функции программы:

просмотр сводной статистической информации по всем услугам с возможностью настройки параметров выбора необходимых пользователю статистических сведений, в том числе настройка периода отображаемых сведений;

внесение статистической информации через электронные формы;

подготовка и выгрузка аналитических отчетов;

предоставление в электронном виде статистических данных в государственную автоматизированную информационную систему «Управление».

## 1.3 Минимальный состав технических средств

Программа может функционировать на технических средствах, удовлетворяющих требованиям, перечисленным в таблице 1.

Таблица 11. Требования к техническим средствам

| **Назначение** | **Технические характеристики** |
| --- | --- |
| **1** | **2** |
| Сервер базы данных и приложений | Архитектура: x86‑64;  ЦПУ: не менее 4 ядер, частота не менее 2,4 ГГц;  ОЗУ: не менее 6 Гбайт;  дисковое пространство: не менее 12 Гбайт для установки базового ПО,  дополнительно выделенный том для размещения  данных и БД – не менее 96 Гбайт;  сетевой интерфейс: 1 Гбайт/c |
| Прокси‑сервер | Архитектура: x86‑64;  ЦПУ: не менее 1 ядра, частота не менее 1,6 ГГц;  ОЗУ: не менее 1 Гбайт;  дисковое пространство: не менее 16 Гбайт для установки базового ПО;  сетевой интерфейс: 1 Гбайт/c |
| АРМ пользователей | Процессор – не менее 2 ядер 1,5 ГГц или совершеннее;  оперативная память не менее 4096 Мбайт;  видеокарта и монитор с разрешающей способностью  не менее 1280\*1024;  жесткий диск объемом памяти не менее 120 Гбайт;  сетевая карта Ethernet 10/100 Мбит/с;  источник бесперебойного питания |

## 1.4 Минимальный состав программных средств

Требования к программным средствам приведены в таблице 2.

Таблица 22. Требования к программным средствам

| **Назначение** | **Общее ПО** |
| --- | --- |
| **1** | **2** |
| Сервер базы данных и приложений | ОС: Ubuntu 20.04 и более новые или Astra Linux 1.7.5;  СУБД: mariadb  Apache Tomcat: не ниже 8.5.72  PHP-фреймворк CodeIgniter 3.1  PHP-FPM 7.1  JAVA v.8  1С Битрикс |
| Прокси‑сервер | ОС: Ubuntu 16.04 и более новые;  Nginx 1.16 и более новые |
| АРМ пользователей | ОС: Microsoft Windows 7/8/8.1/10;  веб‑браузер: Mozilla Firefox 54 и более современные версии  (Google Chrome, Microsoft Edge, Яндекс Браузер) |

# Структура программы

## 2.1 Сведения о структуре программы

Программа построена на основе двухзвенной архитектуры, с выделенными уровнями сервера базы данных и приложений, а также рабочих мест пользователей, организованных с использованием «тонкого клиента».

Описание структуры и алгоритма программы приведены в документе «Статистика   
и аналитика государственных и муниципальных услуг». Описание программы».

# Настройка программы

Для функционирования программы требуется установить на сервер программные средства, приведенные в п. 1.4.

Для упрощения процедуры установки, перед ее началом требуется заполнить таблицу данными, которые пригодятся в процессе установки. Шаблон для заполнения приведен в таблице 3.

Таблица 33. Шаблон с данными, необходимыми в процессе установки

| **Роль** | **DNS‑имя сервера** | **IP‑адрес основного интерфейса** |
| --- | --- | --- |
| **1** | **2** | **3** |
| Прокси‑сервер | proxy.dns.suffix | w.x.y.z |
| Сервер приложений | prefix‑app.dns.suffix | w.x.y.z |
| Сервер базы данных | db.dns.suffix | w.x.y.z |

Установка программы включает следующие шаги:

1. Создать локального пользователя ansible с домашним каталогом и возможностью повышения привилегий в системе с помощью утилиты sudo до уровня пользователя root.
2. Для выполнения и унификации настройки с помощью ansible на сервере CI запустить роль create-users.
3. Установить ПО согласно документу «Статистика и аналитика государственных   
   и муниципальных услуг». Инструкция по развертыванию».
4. Проверить вывод запущенных задач на наличие ошибок, при необходимости внести изменения и перезапустить:

systemctl status tomcat.service

systemctl status nginx.

# Проверка программы

После выполнения установки и настройки программы проверка работоспособности осуществляется в два этапа.

## 4.1 Проверка технического состояния программы

Выполняется непосредственно на серверах. Требует повышения привилегий до уровня пользователя root.

### Сервер базы данных

Проверка программы на сервере баз данных включает следующие шаги:

1. Открыть интерактивный сеанс работы с сервером (консоль или ssh‑сессия).
2. Выполнить проверку наличия рабочих процессов и открытых сетевых сокетов:

# Выводит перечень процессов БД mariadb

# Ожидаемое значение в выводе: строка с идентификатором процесса и временем его активности

ps -aef | grep mariadb

# пример выполнения:

# + 58523 57704 0 12:39 pts/1 00:00:00 grep --color=auto mariadb

# выводит список прослушиваемых сетевых сокетов

# Ожидаемое значение: State=LISTEN, Local Address:Port=:::3306

ss -tln|egrep -i '3306|state'

# пример выполнения:

State Recv-Q Send-Q Local Address:Port Peer Address:Port

LISTEN 0 128 \*:3306 \*:\*

LISTEN 0 128 [::]:3306 [::]:\*

### Сервер приложений

Проверка программы на сервере приложений включает следующие шаги:

1. Открыть интерактивный сеанс работы с сервером (консоль или ssh‑сессия).
2. Выполнить проверку наличия рабочих процессов и открытых сетевых сокетов:

# Выводит перечень java-процессов

# Ожидаемое значение в выводе строка с идентификатором процесса и временем его активности

ps -aef | grep tomcat

# пример выполнения:

9676 9069 0 14:02 pts/1 00:00:00 grep --color=auto tomcat

tomcat 26746 1 12 Jun18 ? 7-21:15:27 /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.232.b09-0.el7\_7.x86\_64/jre//bin/java -Djava.util.logging.config.file=/opt/apache-tomcat-8.5.72/conf/logging.properties # выводит список прослушиваемых сетевых сокетов

# Ожидаемое значение: State=LISTEN, Local Address:Port=:::8087

ss -tln| grep 8087

State Recv-Q Send -Q Local Address:Port Peer Address:Port

LISTEN 0 100 :::8087 :::\*

### Прокси‑сервер

Проверка программы на прокси‑сервере включает следующие шаги:

1. Открыть интерактивный сеанс работы с сервером (консоль или ssh‑сессия).
2. Выполнить проверку наличия рабочих процессов и открытых сетевых сокетов:

# Выводит перечень процессов прокси-сервера

# Ожидаемое значение в выводе строка с идентификатором процесса и временем его активности

ps -aef | grep nginx

# пример выполнения:

root 2699 1 0 Jul21 ? 00:00:00 nginx: master process /usr/sbin/nginx -c /etc/nginx/nginx.conf

# выводит список прослушиваемых сетевых сокетов

# Ожидаемое значение: State=LISTEN, Local Address:Port=:::443

ss -tln|egrep -i '80|443|state'

# пример выполнения:

State Recv-Q Send-Q Local Address:Port Peer Address:Port

LISTEN 0 128 :::80 :::\*

LISTEN 0 128 :::443 :::\*

## 4.1.4 PHP FPM

Проверка программы php-fpm включает следующие шаги:

1. Открыть интерактивный сеанс работы с сервером (консоль или ssh‑сессия).

2. Выполнить проверку наличия рабочих процессов и открытых сетевых сокетов:

# Выводит перечень процессов php

# Ожидаемое значение в выводе: строка с идентификатором процесса и временем его активности

ps -aef | grep php-fpm

# пример выполнения:

# root 806 1 0 2023 ? 00:51:42 php-fpm: master process (/etc/php/7.1/fpm/php-fpm.conf)

root 807 1 0 2023 ? 01:00:25 php-fpm: master process (/etc/php/7.3/fpm/php-fpm.conf)

## 4.2 Проверка установленного ПО

Выполнить проверку веб‑интерфейса программы, для чего с рабочего места открыть в веб‑браузере адрес настроенного сервера приложений вида http://proxy.public.dns-name/.

В веб‑интерфейсе проверить основной функционал, описанный в документе «Статистика   
и аналитика государственных и муниципальных услуг». Руководство пользователя».

# Дополнительные возможности

Для выполнения операций необходим интерактивный сеанс работы с сервером (консоль или ssh‑сессия), в котором команды выполняются с привилегиями root, если не сказано иное.

## 5.1 Процедура запуска/останова программы

### Процедура запуска сервера базы данных

Процедура запуска сервера базы данных выполняется с привилегиями root, если не указано иное и включает следующее:

1. Включить сервер (виртуальную машину).
2. Проверить отсутствие ошибок (устранить, если обнаружены) запуска ОС и системных сервисов с помощью журналов: /var/log/boot.log, var/log/dmesg, /var/log/messages.
3. Проверить открытые сетевые порты баз данных:

# ss -tln|grep -i '3306|state'

1. Проверить журнал предупреждений базы данных на предмет наличия ошибок:

# less /var/log/mariadb/error.log

1. Проверить процессы требуемых сервисов базы данных:

$ systemctl status mariadb

Вывод должен содержать имя БД и статус «READY»

### Процедура останова сервера базы данных

Процедура останова сервера баз данных выполняется с привилегиями root, если не указано иное и включает следующее:

1. Проверить статус БД (выполняется с привилегиями root):

$ systemctl status mariadb

Вывод должен содержать имя БД и статус «READY»

1. Выполнить команду:

$ systemctl stop mariadb

1. Проверить список процессов пользователя mariadb:

# ps -u mariadb

В случае обнаружения процессов экземпляра БД выяснить причину их наличия.

1. Выполнить команду останова ОС:

# init 0

### Процедура запуска сервера приложений

Процедура запуска сервера приложений выполняется с привилегиями root, если не указано иное и включает следующее:

1. Включить сервер (виртуальную машину).
2. Проверить отсутствие ошибок (устранить, если обнаружены) запуска ОС и системных сервисов с помощью журналов: /var/log/messages, var/log/dmesg, var/log/syslog.
3. Проверить наличие процессов требуемых сервисов:

# ps –ef | grep tomcat

1. Проверить доступность приложения по сети:

http://prefix‑app.dns.suffix:8080/

1. Проверить журнал приложения на предмет наличия ошибок:

# less /opt/apache-tomcat-8.5.72/logs

### Процедура останова сервера приложений

Процедура останова сервера приложений выполняется с привилегиями root, если не указано иное и включает следующее:

1. Остановить работу tomcat:

# systemctl stop tomcat.service

1. Проверить наличие процессов останавливаемых сервисов:

# ps –ef | grep tomcat

В случае обнаружения процессов выяснить причину их наличия.

1. Для полного останова сервера выполнить команду останова ОС:

# init 0

### Процедура запуска прокси‑сервера

Процедура запуска прокси‑сервера выполняется с привилегиями root, если не указано иное и включает следующее:

1. Включить сервер (виртуальную машину).
2. Проверить отсутствие ошибок (устранить, если обнаружены) запуска ОС и системных сервисов с помощью журналов: /var/log/boot.log, var/log/dmesg, /var/log/messages, /var/log/syslog.
3. Проверить процессы требуемых сервисов:

# systemctl status nginx

1. Проверить открытые сетевые порты:

# ss -tln | egrep '80|443'

1. Проверить журнал на предмет наличия ошибок:

# less /var/log/nginx/error.log

Проверить доступность веб‑страницы http://<hostname>.

### Процедура останова прокси‑сервера

Процедура останова прокси-сервера выполняется с привилегиями root, если не указано иное и включает следующее:

1. Выполнить команду останова сервиса:

systemctl stop nginx

1. Проверить состояние останавливаемого сервиса:

systemctl status nginx

В случае обнаружения процессов выяснить причину их наличия.

1. Выполнить команду останова ОС:

# init 0

## 5.2 Ограничение доступа на время проведения регламентного обслуживания или устранения критических ошибок

Для ограничения доступа необходимо остановить службу tomcat на сервере приложений:

systemctl stop tomcat

По окончании выполнения требуемых работ необходимо восстановить доступ запустив службу tomcat на сервере приложений:

systemctl start tomcat