|  |  |  |
| --- | --- | --- |
| УТВЕРЖДАЮ  **От Заказчика**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  « » \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |  | УТВЕРЖДАЮ  **От Исполнителя**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  « » \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |

**ВЕБ-ПРИЛОЖЕНИЕ ПО ИЗМЕНЕНИЮ ГОЛОСА С ПОМОЩЬЮ НЕЙРОСЕТИ**

РУКОВОДСТВО ПРОГРАММИСТА

Содержание

[1 Условия применения программы 3](#_Toc135760005)

[1.1 Требования к техническим средствам 3](#_Toc135760006)

[1.2 Требования к общему программному обеспечению 3](#_Toc135760007)

[2 Среда разработки 4](#_Toc135760008)

[3 Характеристика программы 5](#_Toc135760009)

[4 Обращение к программе 6](#_Toc135760010)

[4.1 Управление проектом 6](#_Toc135760011)

[4.2 Содержание проекта 6](#_Toc135760012)

[5 Входные и выходные данные 7](#_Toc135760013)

[5.1 Внесение изменений в инсталляцию 7](#_Toc135760014)

[5.2 Процедура автоматизированного внесения изменений 7](#_Toc135760015)

[6 Сообщения 9](#_Toc135760016)

[7 ПЕРЕЧЕНЬ СОКРАЩЕНИЙ 10](#_Toc135760017)

# Условия применения программы

## Требования к техническим средствам

Для работы программы необходим один сервер с конфигурацией не ниже:

* процессор AMD A8-9600 (3100 MHz); оперативная память не менее 8 Гб;
* свободное пространство на жестком диске не менее 120 Гб;
* операционная система – Windows 2019 Server / Linux Serve;
  + постоянное подключение к сети интернет скорость не менее 10 Мбит/c.

## Требования к общему программному обеспечению

Требования к общему программному обеспечению, необходимому для работы приложения, представлены в таблице 2.

|  |  |
| --- | --- |
| Наименование | Кол-во |
| 1. ОС Debian (Ubuntu) Server/ Windows Server | 1 |
| 2. MS SQL Server (Microsoft SQL Server); | 1 |
| 3. HTML5/Web Assembly: | 1 |

# Среда разработки

Программа проекта была написана на языке программирования Python с использованием фреймворка Flask. Python — это высокоуровневый язык программирования с простым синтаксисом, который обеспечивает эффективное создание кода. Flask - микрофреймворк для разработки веб-приложений на Python, предоставляющий минималистичный и гибкий подход к разработке.

Для разработки была использована среда разработки Visual Studio Code (VS Code). VS Code предоставляет множество полезных функций, таких как подсветка синтаксиса, автодополнение, отладка и интеграция с системами контроля версий. Он является бесплатным и открытым исходным кодом редактором кода, специально разработанным для работы с различными языками программирования, включая Python.

# Характеристика программы

Система рассчитана на круглосуточный режим работы в течение 365 дней в году.

Сервер находиться под постоянным наблюдением специальных программ для мониторинга, как и сам веб-интерфейс приложения.

# Обращение к программе

## Управление проектом

Для того, чтобы запустить проект, программисту необходимо:

1. Авторизоваться в терминале сервера под ролью root,

2. Перейти в папку проекта с помощью команды cd /var/www/projects/rus-export,

3. Запустить сервер приложения: script/server.

Проект должен запуститься на 80-ом порту, после вывода следующих строчек:

bash-3.2# script/server

=> Booting Mongrel

=> Rails 2.3.4 application starting on http://0.0.0.0:80

=> Call with -d to detach

=> Ctrl-C to shutdown server

Входящие запросы будут приниматься по любым интернет-адресам, которые адресуют пользователя на данный сервер.

## Содержание проекта

Проект состоит из следующих файлов

Program.bat – файл для изменения

Program.py – код программы

# Входные и выходные данные

Аудиозапись речи в формате WAV;

Данные о голосах персонажей переозвучки.

## Внесение изменений в инсталляцию

Внесение изменений в работающую версию будет принято  после перезапуска сервера. Перезапуск сервера может быть осуществлен как с остановкой HTTP-сервера и прерыванием соединений, так и без них.

## Процедура автоматизированного внесения изменений

Для автоматизации  внесения изменений для активной разработки, в поставке сайта содержатся настройки утилиты Capistrano. Для автоматизации обновления при постоянных изменениях ее можно использовать без перенастройки со следующим окружением:

* nginx,
* mod\_passenger,
* ssh-ключ для выкладывающего пользователя,
* система контроля версий Git.

Для выкладывания необходимо использовать команду cap deploy. Основные настройки выкладывания расположены  в файле ~/config/deploy/staging.rb.

Описание настроек:

* + Расположение  репозитория Git:

set :repository, "svn+ssh://"

* + Путь расположения на сервере:

set :deploy\_to, "/var/www/#{application}"

* + Имя пользователя, от лица которого Capistrano будет действовать на сервере. Этот пользователь должен иметь возможность получить исходные коды из системы контроля версий и быть доступным тому, кто запускает команду cap deploy.

set :user, "deployer"

* + Имена серверов для программного окружения и  СУБД.

server "[rus-export.dv-t.net](http://forseti.matrixtelecom.net/)", :app, :web

server "[rus-export.dv-t.net](http://forseti.matrixtelecom.net/)", :db, :primary => true

# Сообщения

*В ходе выполнения программы, программисту выдаются сообщения. Здесь приводится описание их содержания и действий, которые необходимо предпринять по этим сообщениям.*

**Возможные ошибки, которые могут появиться при заходе на веб-страницу единого портала:**

dispatch.fcgi

505 Rails application failed to start

**Решение:** мягкая перезагрузка http-сервера

**Возможные ошибки в журнале ошибок (файл /var/log/messages в файловой системе сервера):**

[warn] mod\_fcgid: Read data error, fastcgi server has close connection

[error] [client xxx.xxx.xxx.xxx] Premature end of script headers: dispatch.fcgi

**Решение:** мягкая перезагрузка http-сервера

# ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

|  |  |
| --- | --- |
| ППО | Прикладное программное обеспечение |
| СУБД | Система Управления Базами Данных |