Федеральное государственное автономное образовательное учреждение высшего образования «Московский политехнический университет»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| УТВЕРЖДАЮ | |  | УТВЕРЖДАЮ | |
| Д.Г. Демидов (декан, Московский политехнический университет) | |  | Е.В. Шукалова (старший преподаватель кафедры Инфокогнитивных технологий, Московский политехнический университет) | |
| Личная  подпись | Расшифровка подписи |  | Личная  подпись | Расшифровка подписи |
| Печать | |  | Печать | |
| Дата | |  | Дата | |

Автоматизированная информационная система

Система профессиональной подготовки школьников

АИС «Пора»

**РУКОВОДСТВО ПРОГРАММИСТА**

На 18 листах

Действует с 09.12.2021

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| СОГЛАСОВАНО | |  |  | |
| М.В. Даньшина (заместитель декана, Московский политехнический университет) | |  |  | |
| Личная  подпись | Расшифровка подписи |  |  |  |
| Печать | |  |  | |
| Дата | |  |  | |

**СОДЕРЖАНИЕ**

[1 ОБЩИЕ ПОЛОЖЕНИЯ 3](#_Toc90633180)

[1.1 Полное наименование системы и ее условное обозначение 3](#_Toc90633181)

[1.2 Тема разработки 3](#_Toc90633182)

[2 НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ СИСТЕМЫ 4](#_Toc90633183)

[2.1 Назначение системы 4](#_Toc90633184)

[2.2 Цели создания системы 4](#_Toc90633185)

[2.3 Технические условия применения системы 4](#_Toc90633186)

[3 ХАРАКТЕРИСТИКА ПРОГРАММЫ 7](#_Toc90633187)

[3.1 Требования к режимам функционирования системы 7](#_Toc90633188)

[3.2 Требования к показателям назначения 8](#_Toc90633189)

[3.3 Семантика и синтаксис языка 8](#_Toc90633190)

[4 ОБРАЩЕНИЕ К ПРОГРАММЕ 9](#_Toc90633191)

[4.1 Запуск в полноценной контейнеризованной среде 9](#_Toc90633192)

[4.2 Запуск в контейнеризованной среде разработки 9](#_Toc90633193)

[4.3 Запуск модулей программы в неконтейнеризованном   
окружении 10](#_Toc90633194)

[5 ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ 12](#_Toc90633195)

[5.1 Структура представления входных данных 12](#_Toc90633196)

[5.2 Структура представления выходных данных 13](#_Toc90633197)

[6 СООБЩЕНИЯ 15](#_Toc90633198)

[6.1 Формат сообщений 15](#_Toc90633199)

[6.2 Расшифровка сообщений 16](#_Toc90633200)

# ОБЩИЕ ПОЛОЖЕНИЯ

## Полное наименование системы и ее условное обозначение

Полное наименование системы: Система профессиональной подготовки школьников «Пора».

Краткое наименование системы: АИС «Пора».

## Тема разработки

Темой разработки является автоматизация следующих процессов:

* поиск образовательных курсов для школьников (осуществляется школьником);
* поиск онлайн-олимпиад для школьников (осуществляется школьником);
* продвижение курсов для школьников от высших учебных заведений (осуществляется ответственным за профориентационную работу на факультете);
* привлечение абитуриентов в высшее учебное заведение (осуществляется сотрудником приёмной комиссии);
* поиск потенциальных кандидатур стажеров в компанию (осуществляется HR-специалистом);
* формирование репутации компании как работодателя (осуществляется PR-специалистом).

# НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ СИСТЕМЫ

## Назначение системы

АИС «Пора» предназначена для информирования школьников о проведении образовательных курсов высшими учебными заведениями, а также для проведения компаниями олимпиадных мероприятий для школьников в формате онлайн, которое включает в себя следующие процессы:

* внесение и хранение данных об олимпиадных заданиях представителями компаний;
* внесение и хранение данных об ответах школьников на олимпиадные задания;
* внесение и хранение данных о проверке представителями компаний ответов школьников;
* автоматическое создание сертификата об успешном прохождении олимпиадного задания школьником.

## Цели создания системы

Основными целями создания АИС «Пора» являются:

* повышение эффективности поиска школьниками информации о проведении образовательных курсов высшими учебными заведениями России через агрегацию этих данных;
* повышение лояльности школьников к потенциальным будущим компаниям-работодателям через выполнение олимпиадных заданий от последних;
* повышение лояльности школьников к высшим учебным заведениям через предоставление информации о проведении ими образовательных курсов.

## Технические условия применения системы

В составе комплекса технического обеспечения должны присутствовать следующие технические средства:

* серверы БД;
* веб-сервер;
* ПК пользователей;
* ПК администраторов;
* мобильные устройства пользователей;
* мобильные устройства администраторов.

Серверы БД должны быть объединены в отказоустойчивый кластер. Серверы БД и веб-сервер должны быть объединены одной локальной сетью с пропускной способностью не менее 30 Мбит.

Требования к техническим характеристикам серверов БД:

* процессор – 2 х Intel Xeon 3 ГГц;
* объем оперативной памяти – 16 Гб;
* дисковая подсистема – 4 х 146 Гб;
* сетевой адаптер – 30 Мбит/с.

Требования к техническим характеристикам веб-сервера:

* процессор – 2 х Intel Xeon 3 ГГц;
* объем оперативной памяти – 16 Гб;
* дисковая подсистема – 4 х 146 Гб;
* сетевой адаптер – 30 Мбит/с.

Требования к техническим характеристикам ПК пользователя и администратора:

* процессор – Intel Pentium 1.5 ГГц;
* объем оперативной памяти – 256 Мб;
* видеокарта – 256 Мб;
* дисковая подсистема – 40 Гб;
* сетевой адаптер – 30 Мбит/с;
* монитор – разрешение не менее 300x400 пикселей;
* клавиатура;
* мышь или тачпад.

Требование к техническим характеристикам мобильных устройств – операционная система Android версии 8.1 и выше или операционная система iOS версии 11 и выше.

# ХАРАКТЕРИСТИКА ПРОГРАММЫ

## Требования к режимам функционирования системы

Для АИС «Пора» определены следующие режимы функционирования:

* нормальный режим функционирования;
* аварийный режим функционирования.

Основным режимом функционирования АИС является нормальный режим.

В нормальном режиме функционирования системы:

* клиентское программное обеспечение и технические средства пользователей и администратора системы должны обеспечивать возможность круглосуточного функционирования с перерывами на обслуживание;
* серверное программное обеспечение и технические средства серверов должны обеспечивать возможность круглосуточного функционирования с перерывами на обслуживание;
* должно исправно работать оборудование, составляющее комплекс технических средств;
* должно исправно функционировать системное, базовое и прикладное программное обеспечение системы.

Для обеспечения нормального режима функционирования системы необходимо выполнять требования и выдерживать условия эксплуатации программного обеспечения и комплекса технических средств системы, указанные в соответствующих технических документах (техническая документация, инструкции по эксплуатации).

Аварийный режим функционирования системы характеризуется отказом одного или нескольких компонентов программного и (или) технического обеспечения.

В случае перехода системы в аварийный режим необходимо:

* завершить работу всех систем с сохранением данных;
* выполнить резервное копирование БД.

После этого необходимо выполнить комплекс мероприятий по устранению причины перехода системы в аварийный режим.

## Требования к показателям назначения

Система обеспечивает возможность одновременной работы 1000 пользователей при среднем времени реакции интерфейса на действие пользователя не более 5 секунд.

Система предусматривает возможность масштабирования по производительности и объему обрабатываемой информации без модификации ее программного обеспечения путем модернизации используемого комплекса технических средств. Возможности масштабирования обеспечиваются средствами используемого базового программного обеспечения.

## Семантика и синтаксис языка

Архитектура АИС «Пора» состоит из подсистем-модулей на основе языка программирования Python, фреймворка Django и системы управления базами данных PostgreSQL. Адресация и доступ к модулям регулируются веб-сервером Nginx.

Для построения клиент-серверного взаимодействия использована технология Apollo GraphQL.

Интерфейсная часть системы разработана с использованием свободно распространяемого фронтенд-фреймворка Vue.js.

Для контейнеризации микросервисов, их запуска и масштабирования использованы технологии Docker и Docker Compose.

# ОБРАЩЕНИЕ К ПРОГРАММЕ

## Запуск в полноценной контейнеризованной среде

Перед запуском программы в полноценной контейнеризованной среде необходимо установить следующие утилиты для интерфейса командной строки:

* Docker;
* Docker Compose.

Первый запуск программы в полноценной контейнеризованной среде требует стабильного интернет-подключения для установки зависимостей проекта.

Для запуска программы необходимо от имени администратора системы перейти в корневую директорию проекта и выполнить следующую команду из интерфейса командной строки: «docker-compose -f docker-compose-prod.yml up --build -d».

После завершения загрузки зависимостей и запуска контейнеров для программных модулей система станет доступна на порте, который будет указан в консоли.

Для создания пользователя для отладки с доступом к административной панели необходимо при запущенной программе перейти в корневую директорию проекта и выполнить следующую команду из интерфейса командной строки: «docker-compose run backend python manage.py createsuperuser».

## Запуск в контейнеризованной среде разработки

Перед запуском программы в контейнеризованной среде разработки необходимо установить следующие утилиты для интерфейса командной строки:

* Docker;
* Docker Compose.

Первый запуск программы в контейнеризованной среде разработки требует стабильного интернет-подключения для установки зависимостей проекта.

Для запуска программы необходимо от имени администратора системы перейти в корневую директорию проекта и выполнить следующую команду из интерфейса командной строки: «docker-compose up --build».

После завершения загрузки зависимостей и запуска контейнеров для программных модулей система станет доступна на порте, который будет указан в консоли.

Для создания пользователя для отладки с доступом к административной панели необходимо при запущенной программе перейти в корневую директорию проекта и выполнить следующую команду из интерфейса командной строки: «docker-compose run backend python manage.py createsuperuser».

## Запуск модулей программы в неконтейнеризованном окружении

### Запуск модуля визуального интерфейса системы

Перед запуском модуля серверной части необходимо установить программную платформу Node.js не ниже 16 версии.

Для запуска веб-сервера разработки интерфейса необходимо перейти из корневой директории проекта в папку frontend и выполнить следующую последовательность команд из интерфейса командной строки:

* «npm install -g vue-cli» (выполняется при первом запуске);
* «npm install»;
* «npm run serve».

После завершения загрузки зависимостей и запуска модуля, интерфейс будет доступен по адресу «http://127.0.0.1:3000/» в случае, если порт 3000 не будет занят иным локальным приложением.

### Запуск модуля серверной части системы

Перед запуском модуля серверной части необходимо установить оболочку языка Python версии 3.

Для запуска модуля серверной части необходимо перейти из корневой директории проекта в папку backend и выполнить следующую команду из интерфейса командной строки: «python manage.py runserver».

После завершения загрузки зависимостей и запуска модуля серверная часть, а также интерфейс административной панели будут доступны по адресам «http://127.0.0.1:8000/» и «http://127.0.0.1:8000/admin» соответственно, если порт 8000 не будет занят иным локальным приложением.

# ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ

## Структура представления входных данных

Входные данные от школьника представлены следующим образом:

* фамилия (текстовое поле);
* имя (текстовое поле);
* отчество (текстовое поле);
* дата рождения (поле с типом «дата», представление хранится в формате ДД.ММ.ГГГГ);
* город проживания (текстовое поле);
* email адрес (текстовое поле с типом «email»);
* интересующие школьные предметы (список полей с типом «чекбокс»).

Входные данные от представителя высшего учебного заведения представлены следующим образом:

* фамилия (текстовое поле);
* имя (текстовое поле);
* отчество (текстовое поле);
* email адрес (текстовое поле с типом «email»);
* название высшего учебного заведения (текстовое поле);
* город высшего учебного заведения (текстовое поле);
* описание высшего учебного заведения (текстовое поле);
* название курса (текстовое поле);
* связанные с курсом школьные предметы (список полей с типом «чекбокс»);
* описание курса (текстовое поле);
* длительность курса (поле-раскрывающийся список);
* формат проведения курса (поле-раскрывающийся список);
* даты проведения курса (два поля с типом «дата», представление хранится в формате ДД.ММ.ГГГГ);
* ограничение по количеству слушателей курса (числовое поле).

Входные данные от представителя компании-работодателя представлены следующим образом:

* фамилия (текстовое поле);
* имя (текстовое поле);
* отчество (текстовое поле);
* email адрес (текстовое поле с типом «email»);
* название компании (текстовое поле);
* описание компании и ее деятельности (текстовое поле);
* название олимпиады (текстовое поле);
* связанные с олимпиадой школьные предметы (список полей с типом «чекбокс»);
* описание олимпиады (текстовое поле);
* время на прохождение олимпиады в минутах (числовое поле);
* задания олимпиады (текстовое поле);
* конечная дата проведения олимпиады (поле с типом «дата», представление хранится в формате ДД.ММ.ГГГГ);
* конечная дата оглашения результатов олимпиады (поле с типом «дата», представление хранится в формате ДД.ММ.ГГГГ).

## Структура представления выходных данных

Выходные данные для школьников представлены следующим образом:

* сертификаты об успешном прохождении олимпиадных заданий, доступные для скачивания в формате PDF;
* список всех пройденных курсов в формате отображаемой на экране таблицы;
* список всех решенных олимпиадных заданий в формате отображаемой на экране таблицы.

Выходные данные для представителей высших учебных заведений представлены в виде отображаемого на экране в виде таблицы списка всех школьников, прошедших курсы.

Выходные данные для представителей компаний-работодателей представлены в виде отображаемого на экране в виде таблицы списка всех школьников, прошедших олимпиадные задания, а также в виде доступных для скачивания в формате PDF сертификатов школьников об успешном прохождении олимпиадных заданий.

# СООБЩЕНИЯ

## Формат сообщений

Так как архитектура АИС ImGames реализована на основе языка программирования Python, фреймворка Django, сообщения, выдаваемые программой, будут отсылаться к частям кода, написанного именно на этом языке программирования с использованием вышеприведённого фреймворка.

Сообщения, содержащие системную информацию о работе сервера, позволяют программисту ориентироваться в программном коде, информируют его о неточностях и ошибках. Программа АИС выдаёт сообщения следующих степеней важности:

* «ERROR»;
* «CRITICAL»;
* «WARNING»;
* «INFO».

В журнале консоли сообщения отображаются в формате   
«%(name)-12s %(levelname)-8s %(message)s», где:

* «%(name)» – имя пакета, которое выдает сообщение журнала;
* «%(levelname)» – степень важности сообщения;
* «%(message)» – содержание сообщения;
* «-12s» и «-8s» – виды интервалов между частями сообщения.

Сообщения в этом формате будут выглядеть следующим образом: «django.request WARNING Not Found: /».

Если степень важности сообщения определена как «ERROR» или «CRITICAL», программист, используя информацию об источнике ошибки и её содержании, должен перейти в компонент программы, на который ссылается сообщение, и исправить часть кода, которая указана в сообщении.

Если степень важности сообщения определена как «WARNING», система выявила часть программы, которая потенциально может вызвать неполадки в работе АИС, но которые автоматически решаются. Программисту, используя информацию об источнике предупреждения и его содержании, следует перейти в компонент программы, на который ссылается сообщение, и доработать описанную часть кода.

Если степень важности сообщения определена как «INFO», система информирует о важных этапах работы программы, например, таких как запуск или остановка.

## Расшифровка сообщений

### Группа сообщений «django.request»

Данная группа обрабатывает все сообщения, вызванные HTTP-запросами, и регистрирует исключения для определенных кодов состояния. Все коды ошибок HTTP 5\*\* будут вызывать сообщения «ERROR». Аналогичным образом, коды HTTP 4\*\* будут отображаться в виде «WARNING».

### Группа сообщений «django.server»

Данная группа будет регистрировать сообщения, связанные с обработкой запросов при запуске сервера, точно так же, как группа django.request, за исключением того, что все другие сообщения регистрируются на уровне «INFO».

### Группа сообщений «django.template»

Группа сообщений django.template обрабатывает ошибки, связанные с отображением шаблонов.

### Группа сообщений «django.db.backends»

Данная группа охватывает любые сообщения, связанные с взаимодействием кода с базой данных. Все операторы SQL уровня приложения будут отображаться в регистрируемых сообщениях.

### Группа сообщений «django.security»

Django предоставляет обработчики ошибок, связанных с безопасностью. В частности, эта группа будет охватывать любые сообщения, возникающие в результате ошибки «SuspiciousOperation». Django определяет подклассы для каждого типа ошибок безопасности, таких как «DisallowedHost» или «InvalidSessionKey».

**СОСТАВИЛИ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Наименование организации, предприятия** | **Должность исполнителя** | **Фамилия, имя, отчество** | **Подпись** | **Дата** |
| Московский политехнический университет | Студент | Беляева Дарья Владиславовна |  | 09.12.2021 |

**СОГЛАСОВАНО**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Наименование организации, предприятия** | **Должность исполнителя** | **Фамилия, имя, отчество** | **Подпись** | **Дата** |
| Московский политехнический университет | Заместитель декана | Даньшина Марина Владимировна |  | 09.12.2021 |