

**РАСПРЕДЕЛЕННАЯ СИСТЕМА ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ ДЛЯ
ОБРАЗОВАТЕЛЬНОЙ ОРГАНИЗАЦИИ**

Техническое задание

Титульный лист

Листов 10

1 Введение

Распределенная система представления информации для образовательной организации – это система, позволяющая редактировать и представлять различными способами информацию об образовательной организации.

2 Основания для разработки

Основанием является заказ на разработку распределенной системы представления информации для образовательной организации от Государственного бюджетного общеобразовательного учреждения средней общеобразовательной школы №80 с углубленным изучением английского языка Петроградского района Санкт-Петербурга.

3 Назначение разработки

3.1 Функциональное назначение

Система предоставляет возможности создания расписания занятий для учеников и работников образовательной организации и внесения изменений в созданные расписания. Помимо этого система визуализирует полученные расписания, а также предоставляет возможность добавлять расписания в личный календарь.

3.2 Эксплуатационное назначение

Система позволяет облегчить с помощью частичной автоматизации процессы составления, представления и доставки информации о расписаниях занятий работников и учеников образовательной организации.

4 Требования к программе или программному изделию

4.1 Требования к функциональным характеристикам

Система состоит из двух частей: клиентской и серверной, между которыми должно быть налажено взаимодействие.

4.1.1 Требования к серверной части

Серверная часть должна представлять собой несколько взаимодействующих между собой микросервисов, собранных в виде Docker-контейнеров.

Должно быть организовано хранение информации о расписаниях в базе данных.

Также должно быть организовано взаимодействие с базой данных для осуществления представления информации о расписаниях.

Должен быть реализован интерфейс для получения информации о расписаниях, представленной в виде календарей iCal.

Должна быть осуществлена возможность вывода представленной информации.

4.1.2 Требования к клиентской части

На клиентской части требуется реализовать два компонента: приложение для создания расписаний и внесения изменений в расписания и интерфейс для подключения персонифицированного календаря.

4.1.3 Требования к взаимодействию клиентской и серверной части

Взаимодействие между клиентской и серверной частями должно осуществляться с помощью HTTP-запросов. Обмен данными о расписаниях между клиентом и сервером осуществляется через обмен сообщениями в формате JSON.

4.2 Требования к надёжности

4.2.1 Требования к обеспечению устойчивого функционирования системы

Пользователю, взаимодействующему с системой через веб-браузер должен быть предоставлен непрерывный доступ к веб-сервису, расположенному по определенному адресу. Веб-сервис не должен непредвиденно прерывать свою работу.

4.2.2 Время восстановления после отказа

В случае отказа работы серверной части и последующей недоступности веб-сервиса, время восстановления не должно превышать одни сутки.

4.2.3 Отказы из-за некорректных действий пользователя

После запуска системы на сервере отказ вследствие некорректных действий пользователя должен быть исключен .

4.3 Условия эксплуатации

4.3.1 Климатические условия эксплуатация

Требования к климатическим условиям эксплуатации не предъявляются.

4.3.2 Требования к видам обслуживания

Обслуживание не требуется.

4.3.3 Требования к численности и квалификации персонала

Для использования системы достаточно одного человека, обладающего достаточными квалификациями для составления расписания занятий в образовательной организации.

4.4 Требования к составу и параметрам технических средств

4.4.1 Требования к серверу

Сервер должен быть совместим с любой из операционных систем, поддерживающей работу Docker.

4.4.2 Требования к клиенту

Компьютер клиента должен иметь доступ к сети, в которой работает сервер, а также на нем должен быть установлен любой веб-браузер.

4.5 Требования к информационной и программной совместимости

4.5.1 Требования к исходным кодам и языкам программирования

Исходные коды серверной составляющей должны быть написаны на языке программирования Go.

5 Требования к программной документации

Состав программной документации:

1. Руководство по установке и настройке системы.
2. Руководство пользователя.

6 Технико-экономические показатели

6.1 Ориентировочная экономическая эффективность

В рамках данной работы расчет экономической эффективности не предусмотрен. Использование разрабатываемой системы сократит время, затрачиваемое на составление, представление и доставку информации о расписаниях в образовательной организации.

6.2 Предполагаемая потребность

Предполагаемая потребность обуславливается трудоемкостью процессов составления, представления и доставки информации о расписаниях в образовательной организации.

7 Стадии и этапы разработки

1. Определение данных, необходимых для работы системы.
2. Проектирование базы данных.
3. Разработка модели и архитектуры распределенной системы.
4. Разработка микросервиса, отвечающего за создание расписаний и внесения изменений в расписания.
5. Разработка микросервиса, отвечающего за представление расписаний в виде календаря iCal.
6. Разработка микросервиса, отвечающего за представление расписаний в графическом виде.
7. Контейнеризация каждого микросервиса.
8. Организация микросервисов, представленных в виде Docker-контейнеров, в систему с помощью docker-compose.

8 Порядок контроля и приемки

8.1 Виды испытаний

Проводится проверка корректного выполнения системой заложенных в нее функций, функциональное тестирование системы.

Проводится визуальная проверка интерфейса веб-сервиса.

8.2 Общие требования к приёмке системы

Прием системы будет утвержден при: корректной работе системы при различных входных данных и предоставлении документации к продукту.