

ӘОЖ 004.75:004.7:004.4

РАЗРАБОТКА АРХИТЕКТУРЫ ВЕБ-ОРИЕНТИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ

Султангазиев Е.Р.

магистрант, К. Құлажанов атындағы Қазақ технология және бизнес
университеті, Астана қ., Қазақстан Республикасы

Ұзаққызы Нұргүл

PhD, К. Құлажанов атындағы Қазақ технология және бизнес университеті,
Астана қ., Қазақстан Республикасы

В статье представлен процесс создания архитектуры информационной системы, основанной на веб-технологиях, для выполнения распределенных вычислений. Рассмотрены ключевые аспекты проектирования, включая выбор подходящих инструментов и технологий, а также обеспечение гибкости и производительности системы. Предложенная модель ориентирована на решение задач обработки данных в распределенной среде с использованием доступных веб-решений.

Ключевые слова: распределенные вычисления, веб-технологии, архитектура, информационная система, производительность

Введение

В условиях роста объемов данных и необходимости их оперативной обработки традиционные подходы к вычислениям уступают место распределенным системам. Такие системы позволяют разделять задачи между несколькими узлами, что значительно повышает скорость и эффективность работы. Веб-технологии, благодаря своей универсальности и широкому распространению, открывают новые возможности для создания подобных решений. Целью данной статьи является разработка архитектуры веб-ориентированной системы, способной выполнять распределенные вычисления, сохраняя при этом простоту использования и доступность для пользователей. Задачи исследования включают анализ требований к системе, выбор подходящих технологий и создание модели архитектуры.

Материал и методы исследования

Исследование началось с изучения современных подходов к построению распределенных систем. Были рассмотрены такие технологии, как HTTP-запросы для обмена данными, WebSocket для двусторонней связи и JSON как формат передачи информации. В качестве основного инструмента для серверной части выбран фреймворк Flask (на основе Python) благодаря его легкости и гибкости. Для клиентской части использованы стандартные веб-технологии: HTML для структуры, CSS для оформления и JavaScript для динамического взаимодействия.

Методы исследования включали проектирование системы методом "сверху вниз", при котором сначала определялись общие требования, а затем детализировались компоненты. Также применялся сравнительный анализ различных архитектурных подходов, таких как монолитная структура и микросервисы. В результате был сделан выбор в пользу гибридной модели, сочетающей простоту монолитного подхода с возможностью распределения задач.

Результаты и обсуждение

Разработанная архитектура включает три ключевых уровня: клиентский интерфейс, сервер обработки запросов и слой распределенных узлов. Клиентский интерфейс представляет собой веб-страницу, доступную через браузер. Пользователь может задавать параметры вычислений (например, объем данных или тип задачи) через удобную форму. Сервер, реализованный на Flask, принимает запросы, разбивает задачу на подзадачи и распределяет их между доступными узлами. Узлы представляют собой отдельные серверы или виртуальные машины, которые выполняют вычисления и возвращают результаты на главный сервер.

Для связи между компонентами используется протокол WebSocket, обеспечивающий передачу данных в реальном времени. Например, если задача требует обработки массива чисел, сервер делит массив на части, отправляет их узлам, а затем собирает результаты в единое целое. Такой подход позволяет сократить время выполнения сложных операций, таких как сортировка или вычисление математических функций.

Преимущества предложенной архитектуры заключаются в ее доступности (нет необходимости устанавливать специальное ПО) и масштабируемости (можно подключать новые узлы по мере роста нагрузки). Однако есть и ограничения: производительность системы зависит от скорости сети, а отказ одного узла может замедлить выполнение задачи. Для решения этой проблемы предлагается внедрить механизм резервирования узлов, который будет автоматически переназначать задачи в случае сбоя.

Выводы

Разработанная архитектура веб-ориентированной информационной системы демонстрирует возможность использования веб-технологий для выполнения распределенных вычислений. Система отличается простотой развертывания и гибкостью, что делает ее пригодной для образовательных и исследовательских целей. В дальнейшем планируется доработка системы путем добавления механизмов автоматического масштабирования и повышения отказоустойчивости. Полученные результаты могут быть использованы для создания прототипа системы, который будет протестирован на реальных задачах обработки данных.

Список использованной литературы

1. Кук Д. Основы веб-разработки. – М.: Эксмо, 2020. – 320 с.
2. Смит Дж. Распределенные вычисления: теория и практика. – СПб.: БХВ-Петербург, 2018. – 480 с.
3. О'Рейли Т. Веб-технологии: будущее интернета. – Киев: Технопресс, 2019. – 256 с.
4. Название (English): Development of the Architecture of a Web-Oriented Information System for Distributed Computing

ТАРАТЫЛҒАН ЕСЕПТЕУЛЕР ҮШІН ВЕБ-НЕГІЗДЕЛГЕН АҚПАРАТТЫҚ ЖҮЙЕ АРХИТЕКТУРАСЫН ДАЙЫНДАУ

Сұлтангазиев Е.Р., Ұзаққызы Н.

Мақалада таратылған есептеулерді орындауға арналған веб-технологияларға негізделген ақпараттық жүйе архитектурасын әзірлеу үдерісі ұсынылады. Жүйені жобалаудың негізгі аспектілері, соның ішінде тиісті құралдар мен технологияларды таңдау, сондай-ақ жүйенің икемділігі мен өнімділігін қамтамасыз ету мәселелері қарастырылған. Ұсынылған модель таралған ортада деректерді өңдеу міндеттерін шешуге бағытталған және қолжетімді веб-шешімдерді қолдануға негізделген.

Кілт сөздері: таратылған есептеулер, веб-технологиялар, архитектура, ақпараттық жүйе, өнімділік

DEVELOPMENT OF A WEB-BASED INFORMATION SYSTEM ARCHITECTURE FOR DISTRIBUTED COMPUTING

Sultangaziyev E.R, Uzaqqyzy N.

This article presents the process of developing an architecture for an information system based on web technologies, aimed at performing distributed computing tasks. Key aspects of system design are discussed, including the selection of appropriate tools and technologies, as well as ensuring system flexibility and performance. The proposed model is focused on solving data processing tasks in a distributed environment using available web-based solutions.

Keywords: distributed computing, web technologies, architecture, information system, performance.

REFERENCES

1. Cook, D. *Fundamentals of Web Development*. Moscow: Eksmo, 2020. – 320 p.
2. Smith, J. *Distributed Computing: Theory and Practice*. St. Petersburg: BHV-Petersburg, 2018. – 480 p.
3. O'Reilly, T. *Web Technologies: The Future of the Internet*. Kyiv: Tekhnopress, 2019. – 256 p.
4. *Development of the Architecture of a Web-Oriented Information System for Distributed Computing*.