

*М.П. Хоронко, М.А. Медунецкий, Е.И. Нехведович, 2 курс*

*Научный руководитель – М.В. Стержанов, к.т.н., доцент*

*Белорусский государственный университет информатики и радиоэлектроники*

Node.js – программная платформа, позволяющая языку JavaScript становиться языком общего назначения. Платформа Node.js основана на разработанном компанией Google движке V8 Engine (специальная программа, транслирующая JavaScript в машинный код, распространяется по лицензии BSD). Node.js добавляет возможность JavaScript взаимодействовать с устройствами ввода–вывода через свой API (написанный на C++), подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript–кода. Особенности движка являются компиляция исходного кода JavaScript непосредственно в машинный код, эффективная система управления памятью за счет специальным образом организованной процедуры «сборки мусора» и позволяющая избежать утечки памяти, а также адаптивная оптимизация кода во время компиляции.

Технология Node.js ориентирована на приложения с высокой интенсивностью ввода–вывода и небольшое количество вычислений. Node.js применяется преимущественно на сервере, выполняя роль веб–сервера. Есть возможность разрабатывать на Node.js и оконные приложения и даже программировать микроконтроллеры. В основе Node.js лежит событийно–ориентированное и асинхронное программирование с неблокирующим вводом/выводом.

Основная идея Node.js заключается в использовании неблокирующего событийно–ориентированного ввода/вывода, чтобы оставаться легковесным и эффективным при обращении с приложениями, обрабатывающими большие объемы данных в реальном времени и функционирующими на распределенных устройствах.

Node.js блистает в приложениях реального времени, так как в технологическом стеке многих выдающихся компаний, непосредственно объемы данных в реальном времени и функционирующими на распределенных устройствах.

В состав Node.js входит собственный установщик пакетов npm [1, с. 398], позволяющий получить множество библиотек и программных платформ, созданных различными разработчиками для Node.js. Данная платформа была выбрана для реализации серверной части т.к. удобна для разработки, содержит необходимые библиотеки для взаимодействия с документно–ориентированной базой данных MongoDB, изначально может обрабатывать объекты JSON, которые используются в веб–приложении.

Для удобства разработчиков Node.js предлагает различные платформы (например, Connect, Express, Geddy), предоставляющие широко используемые программные модули и набор функций для создания приложений.

Express.js – это одна из наиболее популярных программных платформ для Node.js. Express.js позволяет быстро разработать каркас сервера веб–приложений, широко применяется при создании прототипа веб–приложения. Предназначена для создания одностраничных и многостраничных веб–приложений. Ключевыми возможностями Express.js являются наличие гибкой системы маршрутизации запросов, перенаправления, а также собственная технология обработки ошибок [2, с. 548]. Содержит множество функций, доступных в виде плагинов. Express.js является частью стека технологий MEAN [3], вместе с базой данных MongoDB и Angular.js.

Платформа Node.js в основном ориентирована на работу с NoSQL базами данных, например, Redis, MongoDB, CouchDB и многими другими. Одной из наиболее популярных документных СУБД NoSQL–типа является MongoDB, поскольку в MongoDB данные хранятся в виде документов формата BSON (двоичный JSON). Для работы с MongoDB (как и для других СУБД NoSQL–типа) существуют различные модули, которые необходимо установить дополнительно из соответствующего источника, например, указать название модуля mongodb в файле package.json. Он поз-

воляет вставлять в текст скрипта стандартные команды для работы с MongoDB: создание, удаление, обновление, поиск и т.п. Обмен данными осуществляется через TCP. Однако Node.js поддерживает работу не только с СУБД NoSQL– типа, но и с реляционными базами данных. Для них также существует ряд модулей, которые устанавливаются дополнительно. Например, для СУБД MySQL имеется модуль `mysql`, который также прописывается в файле `package.json`.

Таким образом, Node.js является привлекательной технологией разработки, тк позволяет группе программистов использовать JavaScript для написания кода на стороне клиента и на стороне сервера. Node.js действительно хорош для создания быстрых масштабируемых сетевых приложений, поскольку позволяет одновременно обрабатывать огромное количество соединений с большой пропускной способностью, что равноценно высокой масштабируемости.

#### **Список использованных источников**

1. Пауэрс Ш. Изучаем Node.js. Издательство: Питер, 2014 г., ISBN: 978–1449323073 (англ.), 978–5–496–00356–8 (русск.).
2. Кантелон М., М. Хартер, Т. Головайчук “Node.js в действии” СПб.: Питер, 2014.
3. Node.js. [Электронный ресурс]. Режим доступа: URL: <https://nodejs.org/en> .Яз. англ.