



УТВЕРЖДАЮ
Директор по персоналу ООО «Рубиус»
(по Доверенности №05 от 01.01.2022)
Ольга Сергеевна Мальцева
«01» января 2022 г.

ПРОГРАММА ПОВЫШЕНИЯ КВАЛИФИКАЦИИ Node.js для Опытных разработчиков (Node.js for Experienced Developers)

Пояснительная записка

- **Актуальность программы.** Данная среда является актуальной, так как в ней возможно реализовать код JavaScript вне браузера, которая позволяет писать серверный код для веб-страниц и веб-приложений, а также для программ командной строки. С помощью Node.js реализуется парадигма «JavaScript для всего». Она предполагает использование одного языка программирования для разработки веб-приложений вместо применения разных языков для работы над фронтендом и бэкендом.
- **Специфика** данной программы обучения состоит в том, что вместо традиционной модели параллелизма на основе потоков автор выбрал событийно-ориентированный подход. Это такая парадигма программирования, в которой выполнение программ определяется событиями.
- **Направленность программы:** естественно-научная.
- Данная программа повышения квалификации создает условия для **непрерывного образования** разработчиков посредством применения и совершенствования их навыков, знаний и умений на практике, в своём рабочем пространстве.
- **Новизна программы** состоит в том, что все теоретические знания, полученные на курсе, закрепляются на практических примерах. Курс ведут опытные преподаватели, профессионалы своего дела. Все они успешно сочетают преподавание с практической работой.

- Данная программа **адресована** для опытных разработчиков, уверенно применяющих JavaScript на практике.
- Данная программа **рассчитана** на 62 академических часа (34 часа аудиторных занятий, 25 часов самостоятельной работы, 3 часа контрольного занятия). Литературные ресурсы указаны ниже в списке учебной литературы.
- Данная программа **рассчитана** на людей, имеющих v\o, ср.p\o.
- Программа состоит из десяти модулей. Все вместе они дают целостное содержание, которое необходимо для профессионального становления разработчиков в данной сфере.
- В результате освоения программы студенты получают сертификаты, подтверждающие прохождение данной программы.

Цель программы - сформировать у слушателей знания и навыки, необходимые для повышения квалификации в сфере Node.js. Благодаря этой программе слушатели расширят круг своих компетенций, освоят новые методики, научатся разрабатывать бэкенд любой сложности на Node.js и глубже окунутся в специфику данной профессии, в результате чего смогут применять полученный опыт в реальных проектах.

Данная программа нацелена на формирование следующих **профессиональных компетенций**:

- умение осваивать методики использования программных средств для решения практических задач;
- умение разрабатывать и клиентскую, и серверную часть кода;
- умение самостоятельно создавать веб-серверные приложения, которые работают в режиме реального времени.

По итогам освоения программы слушатели смогут:

- Глубоко освоить понимание архитектуры и дизайна Node.js
- Создавать веб-серверные приложения
- Использовать инструменты разработчика для написания, контроля качества, тестирования и отладки кода
- Развёртывать проект вручную и в автоматизированных системах
- Использовать популярные модули Node.js
- Работать с менеджером пакетов npm.

Организационные формы учебной и познавательной деятельности

Для решения указанных задач особое внимание уделяется *организационным формам* учебного процесса, включающим инновационные социальные формы работы:

- аудиторные занятия под руководством тренера;
- обязательная самостоятельная работа по заданию преподавателя (домашняя работа, презентации, просмотр и анализ видеороликов, проведение исследования по Case Study и тд.);
- индивидуальная самостоятельная работа с Интернет-ресурсами.

Методы и принципы обучения

Методы обучения, реализуемые в рабочей программе, реализуемые в рамках индивидуального подхода к каждому студенту группы:

- Метод проблемно-ориентированного обучения
- Метод тренингов
- Метод кейсов

Данная рабочая программа реализуется на основе следующих **дидактических принципов** обучения:

- принцип мотивации – поддержание тренером мотивации обучения на высоком уровне, принимая за основу потребности учащихся.
- принцип личностно ориентированной направленности обучения – равновесное и равноправное взаимодействие всех участников учебного процесса, направленное на достижение общей цели;
- принцип профессиональной направленности – введение в содержание обучения профессионально значимого материала;
- принцип создания положительного отношения к учению – развитие сознательного отношения и мотивации к изучению программирования на протяжении всего периода обучения в центре и после окончания обучения;
- принцип системности.

Учебный план

Тема	Количество часов
Вводная лекция. Регламент и цели курса	2,5 часа
Модуль 1. Введение в Node.js. <ol style="list-style-type: none">1. Схема работы на платформе Node.js2. Отличие серверного кода на JavaScript от обычного3. Подготовка основы для проекта и написание первого приложения с интерфейсом командной строки	4 часа
Модуль 2. Библиотека Express.js <ol style="list-style-type: none">1. Библиотеки Node.js для веб-разработки. Express.js2. Развертывание приложений на сервере3. Сравнение реляционных и объектно-ориентированных БД	3 часа
Модуль 3. Введение в базы данных <ol style="list-style-type: none">1. Знакомство с Docker2. Mongoose3. Клиент для MongoDB	3 часа
Модуль 4. TypeScript <ol style="list-style-type: none">1. Система типов TypeScript2. Перенос и интеграция JS-проекта3. Организация проекта и кодовой базы. Разделение приложения на слои	3 часа
Модуль 5. Nest.js <ol style="list-style-type: none">1. Погружение в Nest.js: инверсия зависимостей, провайдеры2. Валидация и обработка ошибок3. Тестирование. Библиотека JEST. Unit-тесты, интеграционные и приемочные тесты	3 часа

Модуль 6. Express.js 1. Обработка запросов клиента 2. Express.js. 3. MVC.	4 часа
Модуль 7. Middleware. Валидация. Загрузка файлов 1. Зачем и как валидировать данные. 2. Загрузка файлов с помощью multer 3. Middleware static	3 часа
Модуль 8. Ограничение доступа. Авторизация 1. JSON Web Tokens. 2. Практика создания приватных маршрутов 3. Отличие авторизации от аутентификации	2,5 часа
Модуль 9. Node.js на Prod 1. Отладка и профилирование Node.js 2. Управление процессами с Node.js 3. Node.js в Docker	4 часа
Модуль 10. Оформление резюме 1. Как составить резюме, пройти собеседование и получить работу.	2 часа
Контрольное занятие	3 часа
Самостоятельная работа	25 часов
Итого	62 часа

Учебно-тематический план

№ п/п	Тема	Количество часов		
		Всего	Теория	Практика

1	Вводная лекция. Регламент и цели курса	4	2	2
2	Введение в Node.js	5	4	1
3	Библиотека Express.js	5	3	2
4	Введение в базы данных	5	3	2
5	TypeScript	7	4	3
6	Nest.js	6	3	3
7	Express.js	5	3	2
8	Middleware. Валидация. Загрузка файлов	7	4	3
9	Ограничение доступа. Авторизация	6	4	2
10	Node.js на Prod	7	4	3
11	Оформление резюме	2	1	1
	Контрольное занятие	3	1	2
	ВСЕГО	62	36	26

Текущий и итоговый контроль (аттестация)

Для эффективного управления и контроля за качеством усвоения учащимися учебного материала организуется текущий контроль, который в себя включает следующие элементы:

- *домашние работы* (под домашними работами понимаются все виды работ, проводимые непосредственно в ходе самостоятельной работы и имеющие целью проверку состояния знаний учащихся; баллы, полученные за эти виды работ, включаются в качестве составной части итогового балла за курс (макс 100 баллов).

- *краткие опросы* (проводятся при проверке пройденного на занятии материала в конце занятия, а также в начале следующего)
- *блиц-опрос* (помогает организовать контроль усвоения текущего материала и закрепление изученного материала, как отдельный блок занятия)

Итоговая проверка знаний осуществляется по средствам выполненных домашних работ.

Шкала оценивания представлена ниже:

<i>Домашние работы №</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>	<i>7</i>	<i>8</i>	<i>9</i>	<i>10</i>	<i>Итого</i>
Максимальный балл	10	10	10	10	10	10	10	10	10	10	100

Итоговый балл: 100 баллов максимум

Результаты домашних работ являются доказательством успешного\неуспешного прохождения курса\модуля и являются основанием к выдаче сертификата\диплома.

Список учебной литературы

1. Браун Э. Изучаем JS. Руководство по созданию современных веб-сайтов: Учебное пособие / Э. Браун. - СПб.: Лань П, 2017. - 368 с.
2. Райлих Н., Т.Дж. Головайчук. Node.js в действии. 2-е издание. / Н. Райлих, Т.Дж. Головайчук. - М.: ДМК, 2018. - 432 с.
3. Шелли Пауэрс. Изучаем Node. Переходим на сторону сервера / Шелли Пауэрс. - М.: LSI, 2016. - 272 с.
4. Дронов В.А., Прохоренко Н.А. JS и Node.js для веб-разработчиков. / В.А. Дронов, Н.А. Прохоренко. - БХВ, 2022. - 768 с.
5. Файн Я., Моисеев А. TypeScript быстро / Я. Файн, А. Моисеев. – Питер, 2021. - 528 с.
6. Хэррон Д. Node.js Разработка серверных веб-приложений на JS. / Д. Хэррон - ДМК.: Пресс, 2022. - 200 с.
7. Кинг С.С. Промежуточное программное обеспечение. Передача данных. / С.С. Кинг – 1992. – 58 с.

8. Касьяро М. Node.js / К.Д. Лециевский, М. Касьяро; Пер. с нем. М.Н. Терехина. - М.: СмартБук, 2019. - 128 с.
9. Грег Лим. Начало Node.js / М.А. Плаксин. - М.: БИНОМ. ЛЗ, 2013. - 167 с.
10. Шерер Дж. Руки вверх JavaScript / Дж. Шерер. Пер. с нем. М.Н. Терехина. - М.: СмартБук, 2020. - 567 с.
11. Адам Д. Скотт. JavaScript везде создание кроссплатформенных приложений/ Адам Д. Скотт. - СПб.: Лань, 2020. - 176 с.