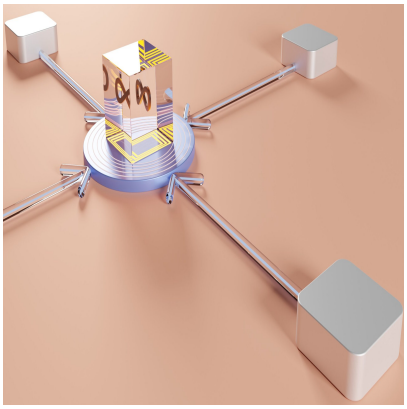

5 Часто задаваемых вопросов и ответов на интервью DevOps

Описание

DevOps становится все более популярным способом создания и развертывания приложений, что привело к росту спроса на специалистов DevOps. В этой статье будут рассмотрены некоторые из наиболее часто задаваемых вопросов интервью DevOps и ответы на них. Использование этих вопросов в качестве основных для изучения может увеличить ваши шансы получить работу, или, если вы являетесь рекрутером, они могут помочь вам в процессе найма.

Часто задаваемые вопросы и ответы на интервью DevOps

Что такое DevOps?

DevOps – это краткая форма разработки и эксплуатации. Эта практика разработки программного обеспечения объединяет команды разработки и эксплуатации для автоматизации каждого этапа проекта. Команда DevOps фокусируется на управлении ресурсами, коммуникации и командной работе. DevOps – это подход, направленный на улучшение работы на протяжении всего жизненного цикла разработки программного обеспечения. Процесс DevOps изображается как бесконечный цикл, включающий этапы планирования, кодирования, сборки, тестирования, выпуска, развертывания, эксплуатации и мониторинга. DevOps влияет на жизненный цикл приложения на всех этапах его планирования, разработки, доставки и эксплуатации. Каждая фаза зависит от других, и фазы не являются специфичными для роли. В настоящей культуре DevOps каждая роль втой или иной степени задействована на каждом этапе.

Каковы различные этапы методологии DevOps

- **Непрерывное развитие.** Это этап планирования и кодирования.
- **Непрерывная интеграция.** Это этап, на котором команда разработчиков интегрирует различные компоненты и гарантирует, что код не сломается.
- **Непрерывное тестирование.** Это этап предварительного и автоматизированного тестирования.
- **Непрерывное развертывание.** Этот этап обеспечивает беспроблемную разработку, которая не влияет на производительность всего приложения.
- **Непрерывный мониторинг.** Этот этап обеспечивает общую эффективность приложения.
- **Постоянная обратная связь.** Анализируются проблемы и производительность, о которых сообщают конечные пользователи.
- **Непрерывная работа.** Позволяет разработчикам создавать более совершенные версии продукта и быстро обнаруживать проблемы.

В чем преимущества DevOps?

Команды, использующие подход DevOps, как правило, быстрее выполняют свои проекты. Таким образом, избегаются ошибки в общении, что способствует быстрому внедрению улучшений и других необходимых изменений. Подход DevOps поощряет более тесное сотрудничество между командами разработки и эксплуатации, где цели всех из них более согласованы. Вы можете обобщить некоторые

преимущества следующим образом:

- Улучшенная коммуникация
- Командная работа и сплоченность
- Улучшенное сотрудничество
- Более быстрая доставка качественных продуктов
- Снижение сложности и быстрое решение проблем
- Большая масштабируемость и гибкость для добавления нового кода
- Более стабильные операционные среды
- Большая автоматизация
- Снижение затрат и использование ресурсов
- Безопасность с помощью интегрированных и автоматизированных инструментов

Назовите некоторые инструменты DevOps

- Инструменты непрерывной интеграции, такие как Jenkins
- Контроль версий, такой как Git
- Инструменты для создания, такие как Maven
- Инструменты управления конфигурацией, такие как Chef
- Контейнерные платформы, такие как Docker

Назовите некоторые ключевые показатели эффективности DevOps

- Частота развертывания
- Изменение громкости
- Время развертывания
- Показатели неудачного развертывания
- Определение времени
- Частота устранения дефектов

Что такое CI / CD и каковы его преимущества?

Непрерывная интеграция (CI)

CI сочетает в себе инструменты и методы, которые упрощают разработку

высококачественного и быстрого программного обеспечения. Таким образом, устраняются узкие места и устраняются технические проблемы на этапах запуска.

Непрерывная доставка (CD)

Эта практика автоматизирует внесение изменений в код, гарантируя, что код может быть быстро дополнен в процессе производства и что приложения работают правильно.

Непрерывное развертывание (CD)

В отличие от непрерывной доставки, целью этой практики является отсутствие вмешательства человека при развертывании программного обеспечения в рабочей среде. Публикация нового или измененного кода в рабочей среде автоматизирована.

CI / CD реализует DevOps на протяжении всего жизненного цикла приложения. Эти методы помогают повысить производительность на различных этапах процесса разработки программного обеспечения за счет их оптимизации и автоматизации.

Что такое архитектура микросервисов и чем она отличается от традиционной архитектуры?

Традиционный метод написания программного обеспечения является монолитным. Это означает, что все функции приложения собраны в одном блоке. В случае ошибок или необходимости обновления анализируется код, управляющий всем продуктом. Это приводит к остановкам или даже сбоям и представляет большую опасность для сохранения кода. Для клиентов это означает потери и головную боль. В архитектуре микросервисов функции приложения представляют собой независимо разработанный и интегрированный сервис. Компоненты являются отдельными, но работают вместе для выполнения взаимосвязанных задач. Таким образом, каждая независимая служба работает, не мешая другим, что также применяется в случаях сбоев. Если функция выходит из строя, ее можно настроить без ущерба для функционирования всего приложения. Появление микросервисов делает DevOps еще более гибкими и эффективными, ускоряя сквозную разработку и делая приложения еще более масштабируемыми.

Разница между DevOps и Agile

DevOps – это культура, которая способствует сотрудничеству между всеми членами команды, участвующими в разработке и обслуживании программного обеспечения. Agile – это методология разработки, разработанная для того, чтобы сделать команды продуктивными и продвигать релизы на основе меняющихся потребностей.

Что такое инфраструктура как код?

Инфраструктура как код – это подход к определению вычислительной и сетевой инфраструктуры, который использует методы управления исходным кодом и рассматривается как любая программная система. Этот код может храниться в системе управления версиями, чтобы обеспечить слышимость и воспроизводимость конструкции, при условии соблюдения практики тестирования и полной дисциплины непрерывной доставки. Инфраструктура как код основана на нескольких практиках:

- Используйте файлы определений
- Самодокументируемые системы и процессы
- Версия все элементы
- Непрерывное тестирование систем и процессов
- Небольшие изменения вместо пакетов
- Постоянно поддерживайте доступность сервисов

Какова роль AWS в DevOps?

AWS предлагает набор гибких сервисов, которые помогают компаниям надежно и быстро разрабатывать приложения. Сервисы AWS упрощают развертывание кода, подготовку и упрощение инфраструктуры, автоматизацию процессов выпуска программного обеспечения и мониторинг производительности инфраструктуры.

В чем польза SSH?

SSH (безопасная оболочка) – это протокол, который обеспечивает безопасное соединение между различными компьютерами. SSH служит различным целям, таким как предоставление; надежная аутентификация, надежное шифрование,

поддержание целостности соединения, надежное соединение и безопасность.

Что такое конвейер DevOps?

Конвейер DevOps – это процесс, который включает в себя несколько шагов перед выпуском нового программного обеспечения. Его можно применять на всех этапах проекта, от интеграции и тестирования до доставки и развертывания. В целом, он проходит следующие этапы и команды:

- Видение продукта: определяет функции продукта;
- Команда разработчиков: развивается итеративно и постепенно;
- Область деятельности: внедряет и поддерживает стабильные среды;
- Мониторинг и обратная связь: стремитесь создавать ценность и использовать ее заказчиком.

Что такое управление конфигурацией?

Это системный инженерный процесс установления и поддержания согласованности производительности системы / продукта, а также функциональных и физических атрибутов на протяжении всего его жизненного цикла.

Что такое непрерывное тестирование?

Это предварительное и автоматическое тестирование приложения по мере разработки.

Что такое Git?

Это бесплатный инструмент с открытым исходным кодом для управления исходным кодом. Этот инструмент предназначен для работы с небольшими и крупными проектами.

Упомяните некоторые основные команды Git

- Git clone

```
git clone <https://name-of-the-repository-link>
```

Копировать

- Ветка Git

```
git branch <branch-name>
```

Копировать

- Проверка Git

```
git checkout <name-of-your-branch>
```

Копировать

- Git добавить

```
git add <file>
```

Копировать

- Git commit

```
git commit -m "commit message"
```

- Git push

```
git push <remote> <branch-name>
```

Копировать

- Мерзавец тянет

```
git pull <remote>
```

Копировать

Объясните разницу между Git merge и Git rebase

Команда Git merge позволяет разработчикам объединять различные ветви Git, в то время как журналы коммитов соответствующих ветвей остаются нетронутыми. Команда Git rebase позволяет разработчикам интегрировать изменения из одной ветви в другую, а журналы изменяются после завершения процесса.

Объясните разницу между Git fetch и Git pull

Git fetch сообщает git для извлечения / проверки последних метаданных из оригинала. Однако он не выполняет никакой передачи файлов, а просто проверяет. Git pull проверяет и переносит изменения из удаленного репозитория в локальный проект.

Что такое конфликт слияния?

Конфликты слияния возникают при объединении двух ветвей с конкурирующими коммит. Конфликты слияния обычно возникают, когда разные разработчики изменяют одну и ту же строку кода или файл.

Что такое контроль версий и каковы его различные типы?

Контроль версий (управление версиями) – это процесс отслеживания изменений в программном коде и управления ими. Это различные типы контроля версий;

- **Централизованный** – иметь единственную “центральную” копию исходного кода на сервере, и все изменения фиксируются в ‘центральной’ копии. Subversion (SVN) и Perforce являются примерами централизованного контроля версий.
- **Распределенный (DVCS)** – нет центрального сервера для хранения всех файлов проекта. Git и Mercurial являются примерами DVC.

Каковы преимущества использования контроля версий?

- Помогает разработчикам сохранять эффективность и гибкость

- Предоставляет полный журнал всех изменений в проекте / файле
- Улучшение отслеживаемости
- Помогает в управлении слиянием и ветвлением
- Позволяет сотрудничать

Что такое ветвление?

Ветвление относится к созданию отдельной / новой версии основного репозитория. Ветки позволяют вам работать над различными частями проекта, не влияя на основную ветку.

Что такое стратегии ветвления?

Стратегии слияния – это методологии, которые команды разработчиков программного обеспечения используют при написании, объединении и развертывании кода. Такой подход гарантирует, что ошибки и ошибки будут исправлены на этапе ветки, прежде чем они будут объединены с основным проектом.

Объясните несколько типов тестов, используемых в DevOps?

Модульные тесты

Они тестируют конкретный фрагмент кода изолированно. Модульные тесты не должны подключаться к базе данных, использовать файловую систему, взаимодействовать с внешними системами или с системными компонентами. Это позволяет им работать быстро, и вы получаете быструю обратную связь, если изменения нарушили существующую функциональность.

Тесты компонентов

Они тестируют большие группы функций, чтобы выявить подобные проблемы. Как правило, они работают медленнее, требуют более сложной конфигурации и большего количества операций ввода-вывода, подключаемых к базе данных, файловой системе или другим системам.

Тесты развертывания

Такие тесты проверяют, что развертывание сработало, другими словами, что ваше приложение правильно установлено, настроено, может получать доступ ко всем необходимым службам и отвечает. Помимо этого, существуют также тесты на эксплуатацию и пригодность к использованию. Чтобы узнать больше о тестировании, ознакомьтесь со статьей о типах тестирования приложений.

Что такое автоматическое тестирование и как автоматизировать тестирование в DevOps?

Автоматическое тестирование – это метод автоматического просмотра и проверки кода. Тестирование автоматизации проводится для проверки соблюдения стандартов качества функциональности (бизнес-логики), стиля кода и пользовательского интерфейса. Тестирование автоматизации выполняется с использованием таких инструментов, как Selenium и Katalon.

Что такое автоматизация сборки и какие существуют инструменты автоматизации сборки?

Средства автоматизации сборки определяют, как должно быть построено программное обеспечение (какие шаги необходимо выполнить и в каком порядке) и какие требуются зависимости (какое другое программное обеспечение должно присутствовать для успешной сборки). Некоторые инструменты лучше подходят для проектов на определенных языках программирования, таких как Maven и Ant от Apache, которые, хотя технология может использоваться с другими языками, в основном используются в проектах Java. Другие, такие как Хадсон или Дженкинс, могут использоваться более широко в различных проектах.

Что такое Jenkinsfile?

Jenkinsfile – это текстовый файл, содержащий определение конвейера Дженкинса. Файл Jenkins проверяется в репозитории системы управления версиями. Некоторые из его применений: он разрешает контрольный журнал для конвейера, позволяет просматривать код и выполнять итерации в конвейере и предоставляет единый

источник истины в конвейере.

Объясните архитектуру в Jenkins

У Дженкинса есть архитектура master-slave. Это означает, что на одного “хозяина” работает много “подчиненных”. Таким образом, распределенная сборка Jenkins может запускать идентичные тесты в разных средах, а результаты собираются и объединяются на главном узле для мониторинга.

Каковы лучшие плагины Jenkins?

- Плагин Blue Ocean для Jenkins
- Плагин почтовой программы для Jenkins
- Плагин Git для Jenkins
- Плагин Jira для Jenkins
- Плагин Docker для Jenkins
- Плагин интеграции Maven для Jenkins
- Плагин Kubernetes для Jenkins
- Плагин SonarQube для Jenkins
- Плагин Amazon EC2 для Jenkins

Что такое Selenium IDE?

Selenium IDE – это инструмент, который записывает взаимодействия с браузером для тестовых случаев. Вы можете устанавливать точки останова, а также проверять и приостанавливать переменные при возникновении ошибок с помощью интегрированной среды разработки Selenium.

Каковы различные компоненты Selenium?

- Selenium IDE
- Selenium RC
- Selenium WebDriver
- Selenium Grid

Какие существуют различные типы тестирования в

Selenium?

- Сквозное тестирование
- Регрессионное тестирование
- Тестирование системы
- Тестирование производительности
- Тестирование совместимости

Каковы различные исключения в Selenium?

- Исключение ElementNotSelectableException
- Исключение NoAlertPresentException
- Исключение InvalidSelectorException
- Исключение NoSuchSessionException
- Исключение StaleElementReferenceException
- Исключение NoSuchWindowException
- Исключение NoSuchFrameException
- Исключение NoSuchElementException
- Исключение времени ожидания
- Исключение ElementNotVisibleException

Что такое driver.close() и driver.quit() в WebDriver?

`driver.close()`Команда закрывает окно браузера при фокусировке. Если открыт только один браузер, он закрывает весь сеанс браузера.

`driver.quit()`Команда закрывает весь сеанс браузера со всплывающими окнами, вкладками и окнами браузера.

Что такое Dockerfile?

Dockerfile – это текстовый документ со всеми командами, которые пользователь может использовать в командной строке для сборки образа.

Различия между образами docker и контейнером docker

Образы Docker – это доступные только для чтения шаблоны, созданные с использованием исходных кодов, инструментов, библиотек, внешних зависимостей и других файлов, необходимых для успешной работы любого приложения в любой операционной системе или платформе. Контейнер docker – это окно, в котором запускаются шаблоны изображений docker.

Объясните архитектуру Docker

Docker использует архитектуру клиент-сервер и включает в себя клиент Docker, хост Docker и реестр Docker.

Клиент Docker использует REST API и команды для связи с сервером (демон Docker).

Клиент Docker использует CLI для выполнения этих команд

```
docker build
```

```
docker pull
```

```
docker run
```

Копировать

- **Docker Host** предоставляет среду для выполнения и запуска приложений. Состоит из демона docker, хранилища, образов, контейнеров и сетей.
- **Реестр Docker** управляет и хранит образы Docker.

Облачные платформы, поддерживающие docker

Платформы, поддерживающие хостинг docker, следующие;

- Kamatera
- Запуск Google Cloud
- Хостинг A2

- StackPath
- Sloppy.io
- Amazon ECS
- Microsoft Azure
- HostPresto
- Vultr

В чем разница между контейнерами и виртуальными машинами?

Как контейнер, так и виртуальная машина работают с виртуализацией и изоляцией сред для содействия обработке независимо от приложений. Тем не менее, контейнеры создают изолированные среды, в которых разные приложения могут выполняться одновременно, поскольку разделение производится на уровне доступных ресурсов, таких как память и обработка. С другой стороны, виртуальная машина позволяет физической машине размещать другие устройства с различными операционными системами, жесткими дисками и аппаратным обеспечением – независимо от исходных.

В чем основное различие между Docker и Kubernetes?

Docker – один из наиболее часто используемых контейнерных инструментов DevOps. Основное внимание уделяется переносимости приложений, размещению их в технологических контейнерах и упаковочных системах и перемещению на разные платформы. В то же время Kubernetes – это инструмент оркестровки контейнеров, который может группировать контейнеры по логической классификации. Он может быть развернут на нескольких компьютерах посредством автоматического распространения.

Что означает виртуализация платформы и каковы ее преимущества?

Виртуализация платформы означает моделирование всей компьютерной системы для одновременного запуска нескольких экземпляров операционных систем на

одной физической машине. В этой конфигурации монитор виртуальной машины (VMM), или гипервизор, имеет полный контроль над аппаратными ресурсами физической машины. Гостевые операционные системы работают на виртуальных машинах, которыми управляет VMM. Виртуализация среды включает в себя моделирование одной или нескольких виртуальных машин и сетевых подключений между ними. Виртуализация может помочь сократить время, затрачиваемое на развертывание, и связанные с этим риски несколькими способами. Использование виртуальных машин при развертывании оказывает огромную помощь в достижении эффективного управления конфигурацией по вертикали и горизонтали в ваших системах. В частности, использование виртуализации дает следующие преимущества:

- Быстрая реакция на меняющиеся требования
- Консолидация
- Стандартизация
- Простота обслуживания базовых показателей

Что такое Nagios?

Nagios – это инструмент, который контролирует всю ИТ-инфраструктуру, чтобы гарантировать, что бизнес-процессы, системы, службы и приложения функционируют должным образом.

Как работает Nagios?

- **Мониторинг**. ИТ-персонал настраивает Nagios для мониторинга сетевых протоколов, системных показателей, приложений, сетевой инфраструктуры, служб и серверов.
- **Оповещение**. Nagios отправляет оповещения при сбое критически важных компонентов. Оповещения могут быть отправлены с помощью SMS, пользовательского кода или электронной почты.
- **Ответ**. ИТ-команда принимает предупреждения и действует.
- **Отчеты**. Создаются отчеты, содержащие хронологическую запись уведомлений, сбоев, ответов на предупреждения и событий для последующего просмотра.
- **Техническое обслуживание**. Запланированные простои не позволяют отправлять оповещения во время обслуживания.

Упомяните некоторые плагины в Nagios

- **Официальные плагины Nagios** – разработаны и поддерживаются официальной командой
- **Пользовательские плагины** – написаны частными лицами / компаниями в соответствии с индивидуальными потребностями
- **Плагины сообщества** – разработаны членами сообщества Nagios

Объясните виртуализацию с помощью Nagios

Nagios предлагает возможность отслеживать различные показатели на различных платформах виртуализации. Nagios может запускаться с различных платформ виртуализации, таких как Xen, Amazon EC2, VMware и Microsoft Virtual PC.

Как Nagios помогает в мониторинге?

Nagios использует такие агенты, как NRPE, check_mk или SNMP, для сбора статистики на вашем сервере и отправки предупреждения, если показатели превышают заданный порог.

Какие номера портов использует Nagios для целей мониторинга?

SNMP, порты 161 и 162

Что такое сетевой анализатор Nagios?

Nagios network analyzer – это программное обеспечение для анализа данных sflow, которое позволяет пользователям активно устранять аномальные поведения, сбои и угрозы безопасности, прежде чем прерывать бизнес-операции. Этот инструмент коммерческого уровня обеспечивает расширенное представление о сетевом трафике и ИТ-инфраструктуре.

Переменные, влияющие на наследование и рекурсию в Nagios

- **Имя** – шаблон “имя”, который может наследовать переменные / свойства объекта.
- **Использовать –** укажите имя объекта шаблона, от которого вы хотите наследовать переменные / свойства.
- **Регистрация** указывает, должно ли имя объекта шаблона быть ‘зарегистрировано’ в Nagios.

Что такое Puppet?

Puppet – это инструмент с открытым исходным кодом для управления конфигурацией и развертывания программного обеспечения.

Что такое кукольный манифест?

Кукольный манифест – это файл, содержащий язык конфигурации кукольного, который описывает, как должны быть настроены ресурсы.

Что такое кукольный код?

Кукольный код – это декларативный код, означающий, что вы описываете его, используя желаемое состояние вашей системы, а не шаги, необходимые для его достижения.

Что такое шеф-повар?

Chef – это инструмент управления конфигурацией, который преобразует инфраструктуру в код. Этот инструмент фокусируется на написании кода, а не на ручном процессе.

Как работает шеф-повар?

Работа шеф-повара начинается, когда системные администраторы или разработчики определяют задачи, которые необходимо автоматизировать. Затем кулинарные книги и рецепты (небольшие программы, написанные на языке, специфичном для конкретной предметной области) собираются и тестируются с использованием различных инструментов, таких как Test Kitchen, ChefSpec и Foodcritic.

Если кулинарные книги и рецепты работают должным образом, они затем развертываются на сервере Chef с помощью инструментов командной строки knife и chef. Процесс развертывания chef включает сервер Chef, рабочие станции и узлы. Вы можете выбрать из более чем 3000 шаблонов кулинарных книг и настроить их в соответствии с вашими потребностями. Затем шеф-повар позаботится об остальной работе.

Расскажите нам о своем опыте управления проектами DevOps

Навыки DevOps – это не просто технические ноу-хау, они требуют человеческих и управленческих навыков, которые часто встречаются у менеджеров, директоров программ и руководителей проектов. Важно знать, как управлять изменениями и поддерживать их, предоставлять шлюз для облегчения переноса новых версий

кода на операционную сторону, оценивать и отслеживать различные платформы и инструменты для измерения производительности, а также анализировать и делиться результатами для улучшения будущих поставок и т.д.

Подойдите к этому вопросу с методологической точки зрения и объясните свой путь в решении сложных проблем. Как вы намерены добиваться улучшения функционирования инфраструктуры? Как вы планируете сотрудничать с разработчиками front, middle и back-office? Какими инструментами мониторинга вы владеете? Кроме того, подумайте о том, чтобы поделиться своим опытом разрешения конфликтов внутри вашей команды.

Краткие сведения

По данным Glassdoor, средняя зарплата инженера DevOps в США составляет 104 281 доллар в год. Несмотря на то, что DevOps – это новая профессия, сейчас мы видим, что все больше компаний нанимают на эту должность и публикуют различные курсы, чтобы обучить профессионалов необходимым навыкам. Выше приведены некоторые вопросы, с которыми вы, вероятно, столкнетесь на собеседовании DevOps.

Дата Создания
14.02.2023