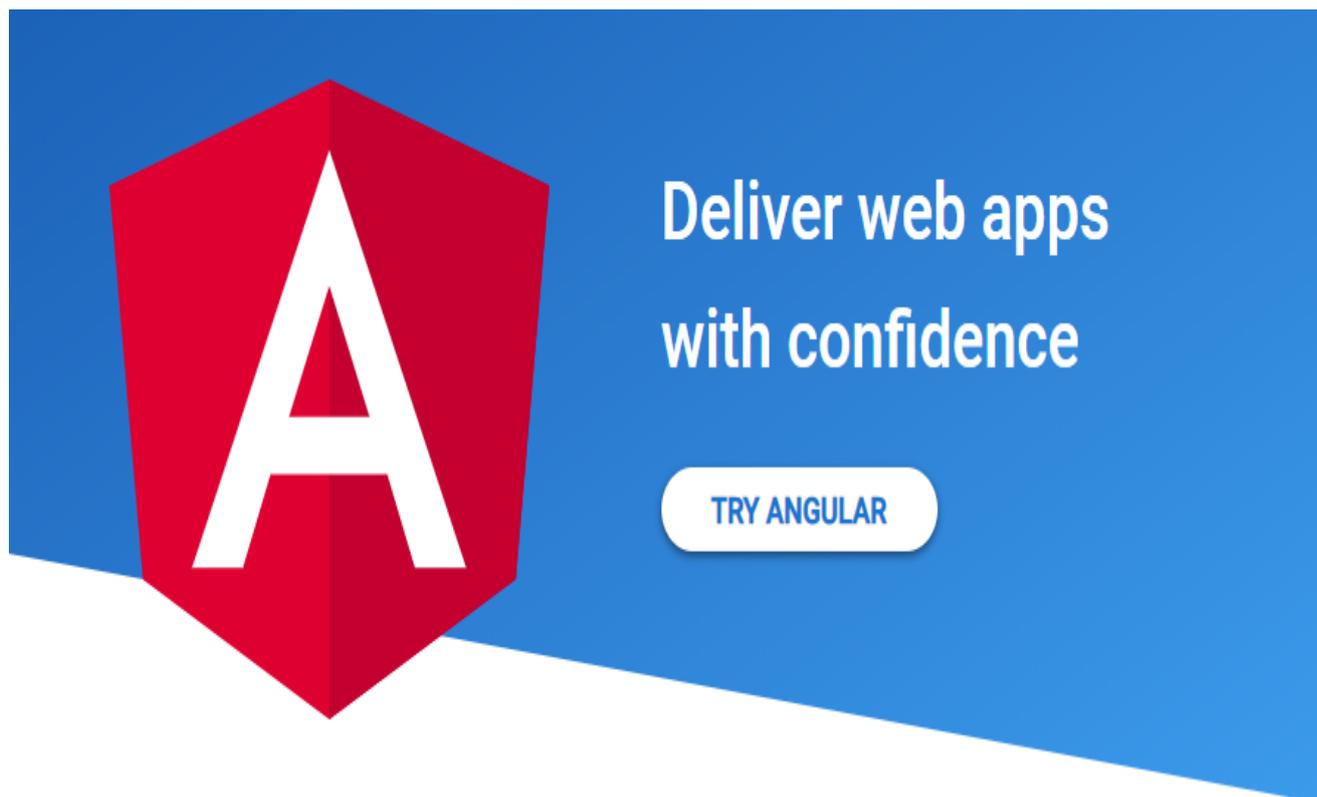


Angular против Vue: Какой фреймворк использовать в 2023 году

Описание

Angular и Vue – одни из самых популярных фреймворков для построения пользовательских интерфейсов. Поскольку многие разработчики и компании используют эти фреймворки в своих приложениях, часто возникают споры о том, какой фреймворк имеет больше преимуществ по сравнению с другим. Однако важно понимать, что оба этих фреймворка – это всего лишь инструменты, и поэтому следует подумать о том, что именно вы хотите создать, а затем подобрать для этого подходящий инструмент. Поэтому давайте рассмотрим оба этих фреймворка, чтобы понять, какой из них лучше выбрать.

Введение в Angular



Angular – это веб-фреймворк с открытым исходным кодом на основе TypeScript, разработанный и поддерживаемый компанией Google и используемый для построения пользовательских интерфейсов в мобильных и настольных веб-приложениях с помощью JavaScript или TypeScript. Angular был выпущен в 2016 году как преемник Angular.js и с тех пор стал вторым по популярности фронтенд-фреймворком; по данным опроса разработчиков StackOverflow, 23 % профессиональных разработчиков используют Angular, в то время как только 10 % разработчиков, обучающихся коду, используют его. Angular предназначен для создания одностраничных приложений (SPA).

В традиционных веб-приложениях каждый раз, когда пользователь взаимодействует с веб-страницей, на сервер отправляется запрос, который возвращает новую страницу, и, таким образом, каждое взаимодействие пользователя требует полной перезагрузки страницы. Однако в одностраничных приложениях все веб-приложение работает на одной странице, содержимое которой динамически обновляется по мере взаимодействия пользователя со страницей. Это позволяет создавать более быстрые сайты с лучшим пользовательским опытом. По словам представителей команды Angular в Google, Angular предоставляет разработчикам все необходимые инструменты для создания

мощных масштабируемых приложений. Кроме того, особое внимание уделяется инструментам, снижающим усталость от принятия решений и повышающим производительность разработчиков, например, мощному интерфейсу командной строки Angular.



Кроме того, фреймворк с самого начала ориентирован на применение лучших практик и имеет множество встроенных инструментов, позволяющих создавать масштабируемые и поддерживаемые приложения без необходимости установки дополнительных библиотек, например, библиотеки маршрутизации. Кроме того, у него отличная документация и сообщество, а новые версии выходят регулярно. Тем не менее, Angular – это фреймворк, который долгое время вызывал много негатива со стороны разработчиков, жаловавшихся в Интернете на то, что фреймворк многословен и очень медленно выпускает новые интересные функции, способные произвести революцию в веб-разработке.

Это можно объяснить тем, что Angular ориентирован на надежность и стабильность, а не на быстрое внедрение новых блестящих функций. Однако стоит отметить, что этот фреймворк очень востребован и является отличным выбором для изучения, если вы рассчитываете получить работу в области front-end разработки. Многие компании предпочитают выбирать надежное и стабильное решение, а не то, что кажется интересным в данный момент. Например, Netflix использует Angular, и вы можете быть уверены, что это происходит по очень веской причине.

Введение в Vue.js



Vue.js, известный как просто Vue, – это прогрессивный JavaScript-фреймворк с открытым исходным кодом и возможностью постепенного внедрения для построения пользовательских интерфейсов. Vue, публичный релиз которого состоялся в феврале 2014 года, был разработан Эваном Ю, который был частью команды Google, работавшей с Angular. По словам Эвана, мотивом для создания Vue послужило то, что ему не нравились дизайнерские решения, которые Angular навязывает пользователям, и он чувствовал, что Angular слишком тяжел и имеет слишком много функций, которые им не нужны. Поэтому он решил создать действительно легкий фреймворк, который обладал бы теми возможностями, которые ему нравились в Angular.

В отличие от Angular, который поставляется с готовыми инструментами, которые разработчики используют при создании фреймворка, Vue использует другой подход. Vue состоит из ядра, включающего только привязку данных и компоненты, которые предлагают очень минимальный набор функций. Однако Vue предлагает официально поддерживаемые адаптируемые части, такие как интерфейс

командной строки, решение для маршрутизации, инструментарий для сборки и т.д., которые вы можете добавить в свое приложение.

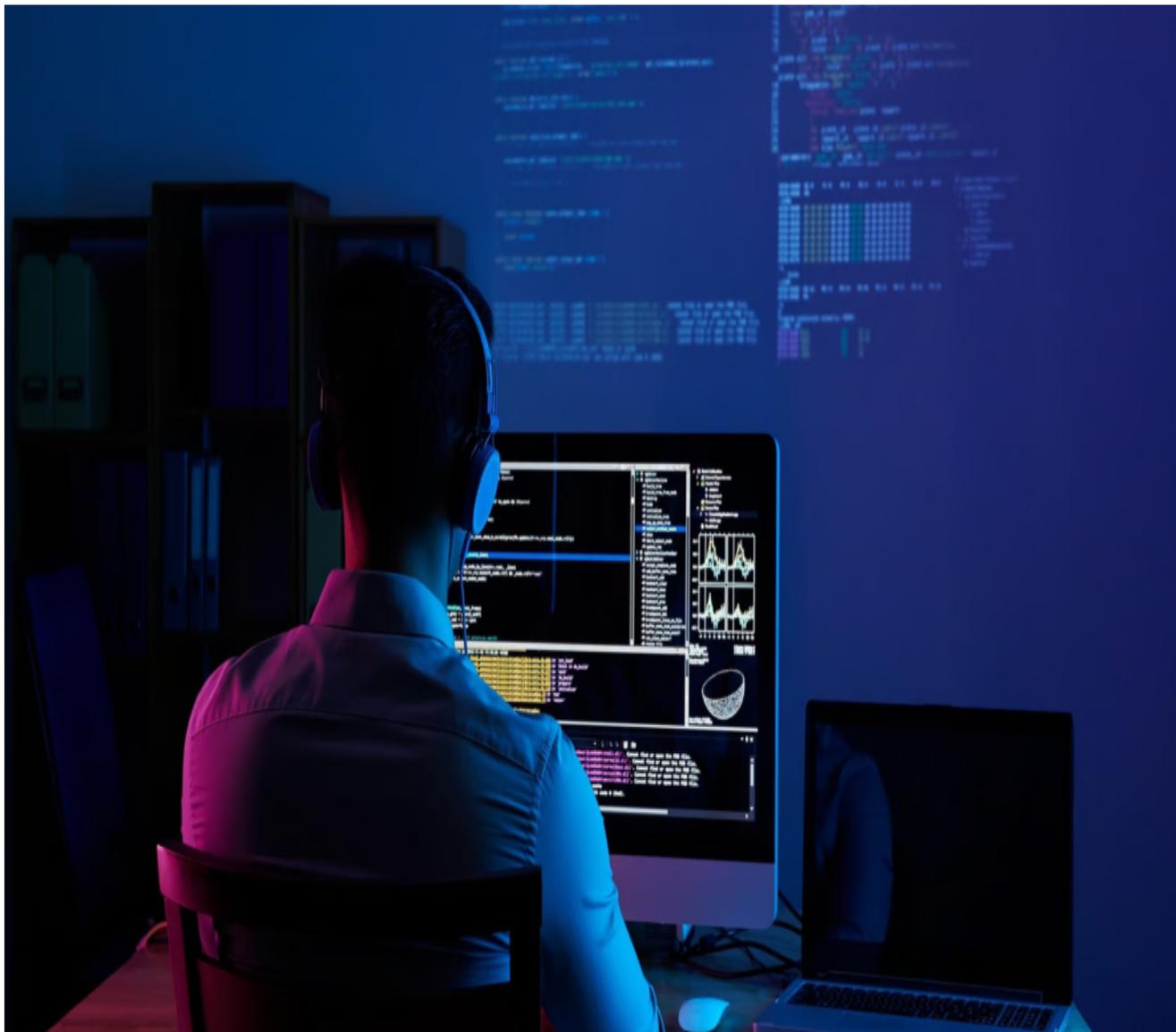
Это позволяет начать работу только с базовыми функциями, а затем добавлять необходимые дополнительные элементы фреймворка по мере роста сложности приложения. Поскольку Vue не принимает за вас решения о том, какие функции использовать, Vue является очень универсальным и гибким фреймворком, который легко масштабировать и использовать для создания высокопроизводительных приложений. При использовании Vue нет необходимости в ручной оптимизации кода, поскольку он оптимизирует его за вас. Несмотря на то, что Vue является более новым фреймворком и не был разработан крупной технологической компанией, как Angular, Vue постепенно завоевывает место надежного и любимого фреймворка для веб-разработки.

Более того, по данным опроса разработчиков StackOverflow в 2022 году, больше разработчиков, изучающих код, изучают Vue по сравнению с Angular. Кроме того, больше разработчиков заинтересованы в разработке с использованием Vue по сравнению с Angular, а Vue занимает более высокое место в списке самых любимых фреймворков по сравнению с Angular. Несмотря на то что Angular и Vue могут быть совершенно разными по подходу к разработке пользовательских интерфейсов, эти два фреймворка имеют ряд общих черт. Давайте рассмотрим некоторые из этих сходств.

Angular против Vue: Сходства

К числу общих черт Angular и Vue относятся:

Компонентно-ориентированная архитектура



Компонентная архитектура – это модель проектирования программного обеспечения, при которой программное обеспечение разбивается на более мелкие, многократно используемые и самодостаточные строительные блоки, называемые компонентами. Каждый компонент выполняет определенную функцию и не зависит от других компонентов. Вместо того чтобы создавать программное обеспечение как единое целое, разработчики создают компоненты, которые затем объединяются в более сложное приложение. Эти компоненты могут быть повторно использованы в любой точке приложения.

И Angular, и Vue при построении пользовательских интерфейсов придерживаются компонентной архитектуры. При использовании этих фреймворков пользовательские интерфейсы разбиваются на независимые многократно

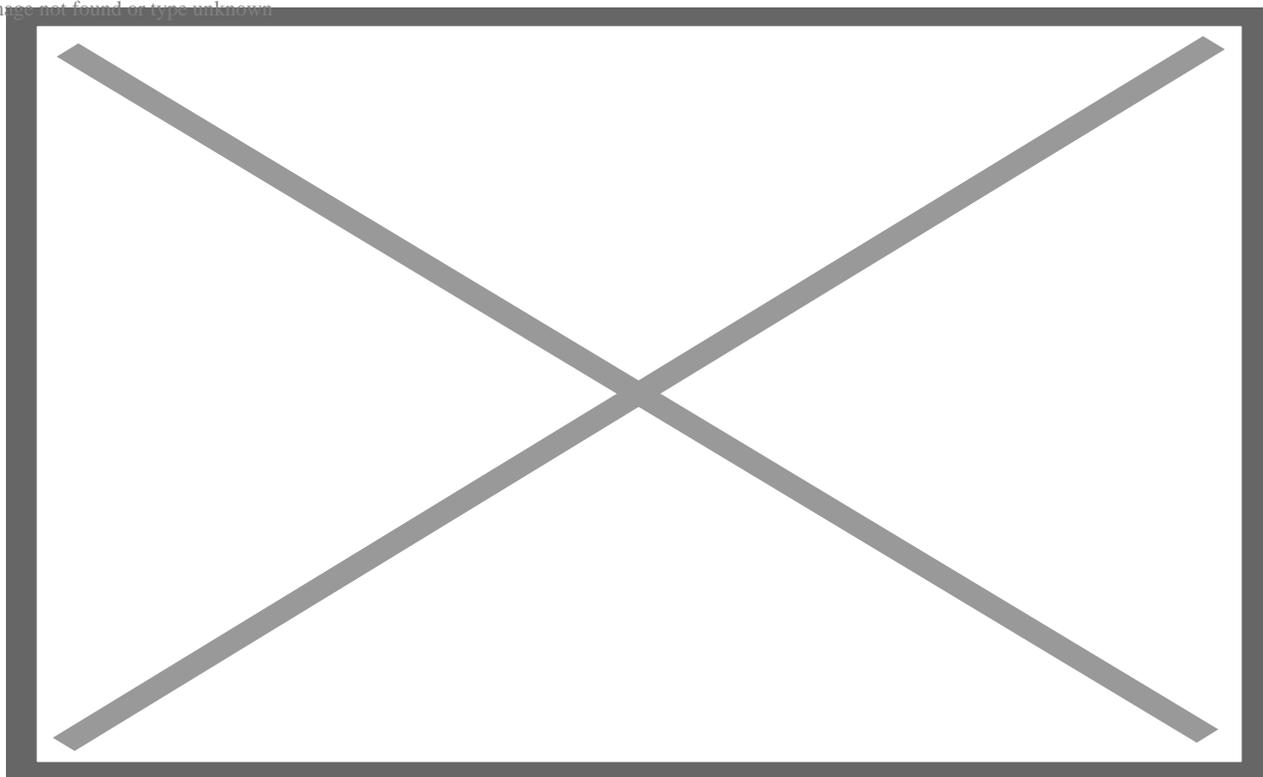
используемые компоненты, которые можно создавать по отдельности, а затем соединять вместе для создания функционального пользовательского интерфейса приложения.

Реактивное связывание данных

Реактивное связывание данных подразумевает привязку данных к представлению. Таким образом, модель данных автоматически синхронизируется с представлениями. Таким образом, все изменения, внесенные в модель данных, отражаются в представлении в режиме реального времени, а все изменения, внесенные в представление, отражаются в модели данных без необходимости вручную манипулировать объектной моделью документа. И Vue, и Angular поддерживают реактивную привязку данных.

Синтаксис

Image not found or type unknown



Несмотря на различия в синтаксисе, поскольку Vue и Angular – совершенно разные фреймворки, оба этих фреймворка объединяет декларативный стиль программирования. При использовании обоих фреймворков вместо того, чтобы описывать шаги, которые необходимо выполнить для достижения определенной

цели, что характерно для императивного стиля программирования, достаточно описать, какая цель должна быть достигнута. Это называется декларативным программированием.

Сообщество

Оба фреймворка имеют большие сообщества разработчиков, которые создают свои продукты с использованием этих фреймворков. Эти сообщества отвечают за совершенствование фреймворков, предлагая добавлять, удалять или настраивать функции. Сообщества, стоящие за этими фреймворками, также отвечают за сопровождение фреймворков, предоставляют отзывы о них, развивают и улучшают их, а также оказывают поддержку другим пользователям. Рассмотрев сходства между этими двумя фреймворками, рассмотрим теперь их различия.

Angular против Vue: Различия

К числу различий между этими двумя фреймворками относятся:

Язык

Angular – это фреймворк, основанный на TypeScript, и поэтому разработчики должны использовать TypeScript при работе с ним. Vue.js, напротив, использует JavaScript, но при этом предлагает официальную и первоклассную поддержку TypeScript. Разработчики могут свободно использовать JavaScript или TypeScript при работе с Vue. С этой точки зрения, хотя освоение Angular может занять больше времени, поскольку сначала придется изучить TypeScript, это лучшая альтернатива, поскольку TypeScript позволяет значительно сократить количество ошибок в приложении. Кроме того, TypeScript выявляет ошибки в процессе разработки, что позволяет избежать дорогостоящих ошибок в производстве. Н

Кривая обучения

Одна из причин, по которой многие разработчики любят фреймворк Vue, заключается в том, что его легко изучать и использовать. По словам Эвана Ю, создателя Vue, Vue был разработан с учетом принципа доступности, чтобы разработчики, имеющие лишь базовое представление о HTML, CSS и JavaScript, могли легко освоить фреймворк. По его словам, Vue является одним из самых

простых в освоении фреймворков пользовательского интерфейса. Angular, напротив, гораздо сложнее в освоении, и вам потребуется гораздо больше времени, чтобы понять и использовать его. Если вы просто выбираете фреймворк, исходя из того, насколько легко его изучить, то выбирайте Vue.

Производительность

Если говорить о производительности, то, поскольку Vue – очень легкий фреймворк, он обеспечивает лучшую производительность и более высокую скорость работы приложений по сравнению с Angular. Производительность Vue также можно объяснить использованием виртуальной объектной модели документа (DOM). Манипулирование реальной DOM является очень затратным процессом с точки зрения производительности. Используя виртуальную DOM, Vue уменьшает количество фактических манипуляций с реальной DOM. В Vue реальная DOM отображается только для тех компонентов, которые были изменены, а не для всей страницы. Это позволяет ускорить рендеринг и, следовательно, повысить производительность Vue.

Экосистема и инструментарий



Когда речь идет о фреймворках, под экосистемой подразумеваются инструменты и библиотеки, доступные для фреймворка. В отличие от Vue.js, Angular имеет развитую экосистему с мощными встроенными инструментами, которые облегчают создание, тестирование и развертывание приложений Angular. Экосистема Vue не такая мощная, как у Angular, но Vue поставляется с официально поддерживаемыми инструментами, которые могут быть интегрированы в приложение Vue в соответствии с потребностями разработчика. Кроме того, Vue.js имеет большой набор сторонних библиотек, плагинов и дополнений, которые можно использовать для настройки приложений или оптимизации процесса разработки.

Гибкость

Как уже отмечалось ранее, одной из причин разработки Vue является то, что Angular очень своеобразен и принимает многие решения за разработчика, предоставляя ему структуру, лучшие практики и рекомендации, с которыми он должен работать. В этом отношении Vue предлагает большую гибкость и менее категоричен по сравнению с Angular. Vue поддерживает компоненты, которые разработчики могут интегрировать и использовать для настройки своих приложений по своему усмотрению на любом этапе их разработки. Если гибкость является важным фактором, то Vue – это очевидный выбор, поскольку он легко адаптируется к любому проекту. Однако стоит отметить, что многие крупные предприятия ценят четко определенную структуру Angular, поскольку она позволяет создавать более стабильные приложения.

Масштабируемость

Если вы создаете крупномасштабное приложение и, соответственно, заинтересованы в фреймворке с отличной поддержкой масштабируемости, то для вас очевидным выбором будет Angular. В отличие от Vue, Angular создан с учетом требований масштабируемости, и этот фреймворк предназначен для того, чтобы команды могли легко масштабироваться. Angular предоставляет встроенные инструменты и библиотеки, поддерживающие масштабирование приложений. Это объясняет, почему многие масштабные приложения, такие как Netflix, построены на Angular. Несмотря на то что Vue.js также поддерживает масштабирование, масштабировать с помощью Vue гораздо сложнее, и при использовании этого фреймворка могут потребоваться сложные стратегии масштабирования.

Скорость разработки

Если бы вы разрабатывали одно и то же приложение с использованием Angular и Vue, то разработка приложения с использованием Vue заняла бы гораздо меньше времени. Angular требует значительно больше времени для разработки приложений, поскольку это очень обширный фреймворк, а также очень многословный. И наоборот, Vue предлагает значительно меньшее время разработки, поскольку писать и читать код проще, а фреймворк не требует такого количества шаблонного кода, как Angular. Vue также использует шаблоны, написанные на стандартном HTML, и это может быть быстрее, поскольку

разработчики лучше знакомы с HTML, чем дополнительный синтаксис написания шаблонов в Angular, специфичный для Angular.

Angular против Vue.js: краткое описание

В таблице ниже приведены основные различия между Angular и Vue.js:

	Angular	Vue.js
Язык	TypeScript является основным и рекомендуемым языком для создания приложений Angular.	Имеет официальную поддержку как JavaScript, так и TypeScript.
Кривая обучения	Крутая кривая обучения и требует много времени и усилий для освоения и использования.	Значительно проще в освоении и использовании
Производительность	Отличная производительность и высокая скорость работы, особенно при работе со сложными приложениями.	Надежная экосистема с большим количеством мощных встроенных инструментов для разработки.
Экосистема и инструментарий	Надежная экосистема с мощными встроенными инструментами.	Менее надежная экосистема и меньшее количество встроенных инструментов, но множество сторонних дополнений.
Гибкость	Очень ограниченная гибкость	Обеспечивает большую гибкость
Масштабируемость	Легко масштабируется и имеет встроенную поддержку масштабирования.	Масштабирование гораздо сложнее и требует сложных стратегий.

Скорость разработки

Скорость разработки
очень низкая.

Обеспечивает высокую
скорость разработки.

Компании, использующие Angular или Vue

Хороший способ оценить, насколько хорош тот или иной фреймворк и насколько он может помочь вам в трудоустройстве, – это посмотреть на компании, использующие эти фреймворки. Среди крупных компаний, использующих Angular, можно назвать следующие:

1. Netflix
2. Microsoft
3. Google
4. Paypal
5. Deutsche Bank
6. Tesla
7. Forbes
8. Фрилансер
9. Wise
10. Santander

Среди крупных компаний, использующих Vue.js, можно назвать следующие:

1. Adobe
2. Gitlab
3. Behance
4. Dribbble
5. Euronews
6. Trivago
7. Nintendo
8. Euronews
9. Xiaomi
10. Alibaba

Angular и Vue.js используются очень крупными компаниями, что говорит о том, что оба этих фреймворка являются отличными инструментами для изучения с точки зрения трудоустройства. Не стоит беспокоиться о том, что в ближайшие годы ни один из этих фреймворков не будет востребован разработчиками.

Заключение

Как уже было сказано выше, Vue и Angular – это инструменты, которые можно использовать для построения пользовательского интерфейса приложений. Решение о том, какой инструмент выбрать, должно зависеть от того, чего вы хотите добиться и что ищете во фреймворке. Если вы ищете гибкий фреймворк, простой в освоении и внедрении и предлагающий высокопроизводительные приложения, то Vue.js – это очевидный выбор. Если же вы создаете сложное крупномасштабное приложение и поэтому нуждаетесь в стабильном приложении, которое можно быстро и легко масштабировать, то Angular должен стать вашим основным инструментом.

Дата Создания

08.08.2023