Differences Between API and Microservices? Here's all you need to know.

You may have heard of API and microservices. They are used extensively to build useful and important software applications. Some people may be confused about whether [API and microservices](#) are the same or different. We explore these two cutting-edge technologies.

Introduction to Microservices

Microservices is a technique to develop applications as a group of independent units. Microservices architecture is an improvement over the monolithic applications of the past. The latter is a single unit. On the other hand, in microservices architecture, there are a number of independent units each having a function or multiple functions. All components are self-contained with a separate UI (User Interface) and database if applicable. Microservices are a type of distributed computing architecture.

Advantages of Microservices

- Changes made to one component don't affect the other components.
- Different components can be made in the technology that is most suitable and productive. Separate components can be developed using different technologies.
- When a component goes down it doesn't affect the other components. The latter continues to work as usual.
- Developers building a component don't have to wait for any other component to be completed first. So, the development effort happens in parallel.
- Components can be more easily upgraded to new technology. Upgrades are difficult and time-consuming in the case of applications built using monolithic architecture.
- There is a faster time to market. As a result, your business commences early translating into increased revenue as well as profits.
- Implementing microservices significantly enhances security. If a particular component is compromised the vulnerability doesn't migrate to other components. On the other, in the

case of monolithic architecture, the different parts are interconnected and interdependent. If a hacker is able to penetrate one part, he/she will find it easy to attack other parts of the whole application.

- Superior flexibility. If developers want to develop a particular component in a different way, they have the full freedom to do so. This feature is difficult in monolithic applications because the different parts are both interconnected and interdependent.
- A programmer doesn't need knowledge of the overall architecture in order to comprehend a single part of the application.
- Components can be added or deleted easily as per requirements and preferences. Scalability is easy and convenient in software implementing microservices architecture.

Examples of Microservices

Many industry majors including [Amazon](), Netflix, and PayPal have adopted microservices architecture. In the past, they were using monolithic applications. However, when companies grow and expand monolithic applications become a liability because of the drawbacks of the monolithic architecture. The advantage of microservices architecture is that applications developed using this technology can be scaled up or down with ease.

Introduction to APIs

The full form of API is Application Programming Interface. APIs enable applications to communicate seamlessly with each other. The former also makes it possible for disparate applications to exchange information. Finally, APIs specify the manner of interaction between applications. To your knowledge, a microservice can use an API to communicate with another microservice. In technical terminology, an API is a set of protocols as well as methods that outline how to separate applications and share and modify data between themselves. Thanks to APIs applications having different architecture and features can communicate efficiently between themselves in a defined format.

APIs by themselves have no value. They expose the functionality of an application that can be used by other applications without knowing the internals of the functions. For your information APIs can be either public or private. There are many other types of APIs existing in the market

today. In microservices architecture components communicate with each other, obtain/send data and use functionality thanks to APIs.

Advantages of APIs

Maintains the Privacy of Data
When an API is invoked, it doesn't expose the internals of the desired functionality and information. Programmers don't need to know how a function works. They only need the desired response.

Ensures Superior Customer Experience
APIs aid clients in conveniently interacting with enterprises. This enables the latter to have a better comprehension of the client's pain points as well as interests. Organizations can use this data to deliver better-quality products and/or services.

No Need to Develop Software from Scratch
Sometimes software with the desired functionality is already existing in the market. Developers need not build all software components from scratch. Don't make the fundamental and frequent mistake of attempting to reinvent the wheel. Software API calls obtain the desired functionality. Some APIs are even free. So, you save on the software project expenses. Some APIs come at a price but are generally cheaper than having to build the software module on your own. You don't have to understand how the software works to benefit from the software API.

Improves Collaboration.
By leveraging private APIs, the quality of communication and collaboration in teams is elevated. This results in ramping up synergy and achieving optimum teamwork as well as performance.

Ramps up Innovation
If you have to survive as well as thrive in the extremely competitive ecosystem it is imperative to have a high degree of innovation. The latter is achieved by utilizing the power and other significant capabilities of APIs. A tip is to use API and microservices together for synergy as well as stellar results.

Examples of APIs

Suppose you want to book a flight ticket from a travel website. The latter shows the flight times and ticket fares of different airlines. How is that possible? Thanks to APIs the travel website gets all the desired information without having to get access to the different airline's private databases and you needn't navigate away from the travel website. The different airline's databases remain private and yet you get the desired information.

A popular example of API is websites requiring social login. There is an API that transmits your user id and password to social networking websites. The website which requires a login is not allowed to see your credentials. Yet you can log in and not leave that website.

Some common everyday usage of API includes downloading software, online orders of food/beverages via a delivery service, and withdrawing money from an ATM. The ATM entertains debit/credit cards from different banks. You get money from the bank without the ATM software knowing your account number, balance, and other confidential details. You don't have to go to your bank to withdraw cash.

When you order food from a food delivery platform you don't have to directly contact the restaurant. You also don't have to directly pay the restaurant for your order. The API will place the order, process the transaction, and enable you to track the delivery staff.

What are the Differences between API and Microservices?

- APIs take longer time to develop than microservices
- APIs are generally larger in size than microservices
- Their respective architectures are different.
- APIs are a type of interface while Microservices consist of a group of multiple and independent components.
- APIs deliver useful functionality or data while microservices are the building blocks of a particular application.
- APIs enable communication between components while microservices contain components providing services.
- There are 2 kinds of microservices. They are stateless microservices and stateful microservices. There are more than 2 kinds of API. They are REST (Representational State Transfer), SOAP (Simple Object Access Protocol), remote, public, private, partner, and others.

- In the case of microservices, each unit includes business logic, a database, a dedicated data access layer, and relevant APIs. While in the case of API, the fundamental parts are functions, tools, protocol, and standard formats.

While API and microservices are distinct technologies they are popularly used together for superior results. To use them you must have clarity of each technology. Then you are well-positioned to leverage the power of API and microservices. Remember that grasping the concepts of API and microservices requires an investment of time and effort. Investing in implementing API and microservices is sure to yield manifold returns.

Conclusion

Now that you are aware of the differences between API and microservices include them in your software to avail of considerable benefits. API and microservices together work to enhance the performance, productivity, and profits of any business. Start using API and microservices to be competitive in the market. If you have in-house expertise and experience in developing API and microservices then it is a bonus. You then don't have to hire an external provider to develop API and microservices. However, if you lack the capability, save time, money, and effort by using the services of a professional outsourcing firm. Firstly, evaluate whether their team is sound in the concepts of API and microservices. Prepare a list of relevant questions about API and microservices. Ask them about their past projects involving the usage of API and microservices. Only if you are entirely satisfied avail of the services of that firm.

About Us

Focaloid is a stellar [software development products and services company catering to clients in the US & UK](#). We have the necessary expertise and experience in building APIs and microservices. For your information, Focaloid has a team of talented and seasoned professionals with extensive expertise in API and microservices. Over time we have established a mature outsourcing model which is productive, efficient as well as effective. You can safely rely on us to deliver outstanding software solutions and services within stipulated deadlines. Contact us at your earliest to know how we can help you. We are committed to superior client satisfaction and mutual growth.

FOCALOID
TECHNOLOGIES

Differences Between API and Microservices?
Here's all you need to know.