Why we need APIs

A shared code known as an API links two or more components of an application. An API is actually just a set of instructions that are understood by both the client and the server. But why do we even need to discuss APIs? One explanation is that APIs are necessary for creating reliable apps.

Here are some examples of why APIs are important in today's technology landscape:

Integrating with third-party services:

Many applications need to interact with external services to provide additional functionality or data to their users. For example, a weather app might use an API to retrieve up-to-date weather information from a weather service, or a social media app might use an API to post updates to a user's profile on another platform. By integrating with external services via APIs, developers can create more robust and feature-rich applications.

Creating platform ecosystems:

APIs are also essential for creating platform ecosystems, where multiple applications can work together to provide a seamless experience for users. For example, the Google Maps API allows developers to integrate maps and location-based data into their own applications, which in turn can be integrated with other applications to provide a more comprehensive solution.

Enabling mobile app development:

APIs are especially important for mobile app development, where developers need to access data and functionality from multiple sources in order to create a compelling user experience. For example, a mobile banking app might use APIs to access account information from a bank's backend system, while also integrating with payment processing systems to enable transactions.

Accelerating the development of systems and software:

Instead of creating large systems, APIs let developers code and deliver functionality as microservices. By segmenting this functionality, developers can actually accelerate the creation of both software and system development by removing dependencies and lowering the regulatory burden of things like code reviews and testing.

APIs also speed up development by enabling frontend and backend teams to collaborate in parallel. In other words, while a backend developer works on the system's backend components, which users and customers do not see, a frontend developer can focus on building the system's interface.

Reducing operating expenses:

APIs are also used by developers to cut operating expenses. APIs can assist with a variety of functions that formerly might have been handled by humans, such as retrieving reports, sending emails, and

abstracting data from one system to share with another. There are a plethora of additional ways that APIs can lower running costs. Examples include scheduling employees' workdays, automating the start-up and shutdown of industrial equipment, and lowering the number of users who require software licenses.

Lowering the cost of software development:

The ability to build reusable components is one of the main ways APIs may lower software development expenses. For instance, a backend developer could build a system that provides details on clients, such as names, email addresses, and recent purchases of goods. Then, other programmers from different parts of the company can utilize APIs to access that data and track payments for finance and accounts payable, assist customer service in solving issues more quickly, or even provide suggestions for marketing campaigns.In addition to lowering architectural complexity and the effort needed to modify systems, APIs can also assist lower software development costs in other ways.

Enhancing system and software testing:

By enabling quality engineering teams to segregate tests for frontend components, or the portions of software customers see, from tests for backend components, or the parts of software users don't see, APIs can aid in improving software and system testing. Automation testing can be used to evaluate the health, quality, and performance of APIs, and it can be incorporated into the pipeline for CI and CD.

Enhancing corporate governance and security:

APIs can be used to enhance organizational security. For instance, APIs are frequently used to provide Single Sign-On, which enables users to log in to numerous services with just one username and password. This reduces the likelihood of having a sticky note collection of usernames and passwords, which poses a serious security concern.

APIs frequently play a significant role in corporate governance as well. APIs can be used to automate and enforce business regulations and policies, such as requesting authorization before paying employee expenses.

Enabling mobile applications:

In order to provide consumers with critical information, many mobile applications rely on APIs. For instance, if you book a flight using your smartphone and choose your seat, APIs can transmit the seat you choose so that the flight attendants are aware of your location when you board.

Reducing outages and non-performing systems:

Finally, APIs can assist lessen system failures and outages. For instance, a business might utilize an API to swiftly pinpoint a specific issue with a production line and even suggest a solution, which enables maintenance staff to address the issue and get the system back online more quickly.