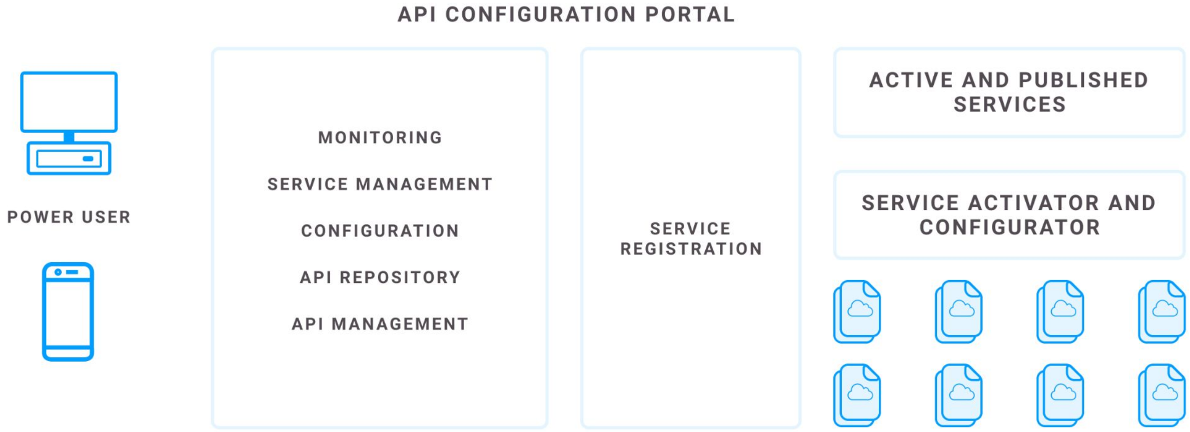
CASE STUDY

API Framework

Adding a custom API Framework layer to a complex legacy system created a solution that is now one of the three core systems our client has, and on which everything else rests



Challenge

Our client serves millions of customers per month.

Over the years they have worked with multiple vendors who have built apps on top of their legacy code system, which has become extremely complex and unstable over time.

All the services in the client's portfolio were developed by different vendors/teams who had used various architectures and patterns.

Every service had its own login and monitoring, with many of those services being mutually dependent. Over time, the resources needed to maintain the code skyrocketed.

Furthermore, the vulnerability and instability of the system made it virtually impossible to implement new services, since adding a new app would endanger the functioning of the system as a whole.

Full Stack: .NET

7 Months

Solution

Our solution was based on introducing a new layer to the system, API Framework, as a configuration portal to be used for configuring, managing, and integrating different services/apps.

Within the portal, we implemented shared authentication/authorization, tracking, and monitoring.

Since our client hosts personal user data, we needed to increase overall security, so we made sure that external API access points were implemented with the highest level of protection.

Once we created and configured the API Framework, we separated various services/apps and standardized their architecture.

Benefits

The solution we developed has become one of the three core systems our client has: the core system, the business process modelling system and now the API framework.

The API Framework, functioning as a central hub, enables our client to turn off specific parts of the system (i.e. services and apps), while the rest of them remain available to the customers.

We have set up development templates for future vendors, both for developing new apps, and for refactoring existing modules, ensuring proper architecture, app security, and services built on top of the API framework in the future.

Since our client's internal IT team was not familiar with the API framework, we prepared a comprehensive set of documents with instructions and explanations.

Additionally, we held a series of workshops for their developers, enabling them to take full ownership of the solution.