# Skill management

For the 1st year of my developer position I've made **a number of important decision**s which affected the architecture designed **before**. After a year we as a development team and product owner decided to go for an update from .NET Framework 4.7 to .NET Core (first to 3, then to 6). We understood that with this update we **also have to boost the performance** of our application not just by newer technology, but also with a proper architecture. In spite my contribution to the project and the absence of the architect I volunteered to take these responsibilities and the team agreed.

As I've said we had some performance issues, so my first task was fix them. For that I've decided to host some test services in a private network that were monitoring our services. As we discovered later, one of the issues was waterfall retries on high load, so the solution I've chosen was a Circut breaker pattern.

*Services were hosted in a private network and messages about retries were posted to Kaffka and then processed by this service.*

Also there were some issues with querries to database.

The next step was to upgrade the architecture for a simplier management and development. At the beginning the project was developed with the SOA approach and I've updated it to a more microservices-like look.

And one more thing I've updated was the deployment of the application.

Initially the app was provided for the client with "Environment copy" model, so it was hosted on a separate server for each client and it was expensive. I've decided to move common modules and rarely used data to a separate scalable services (support, common data, rarely used data, reports) and the core logic was hosted on the same servers but as a multitennant app. As a result requirements for a copy lessened as the cost.

# Corporate mailing

Initialy it was a small project where the customer needed only an application for a corporate documents exchange, so I just had to develop a simple architecture of client app and and API, but later the custome asked me to help with their internal services architecture. As the customer did't have the documentation for their projects, most difficult task was the discovery of them, cause the goal was to design the architecture of services for private and public networks.

The app itself allows companies and departments to specify thir own workflow of documents signing and any other type of processing with a flexible configuration.

Public network

* Client app
* Secure Document Transmission - this service handles the secure transmission of documents between clients and the internal system. It should include encryption, secure file transfer protocols, and data validation.
* Customer Support and Communication (Public Network):

Private

* Documents storage and management
* Reporting and analytics
* Workflow automation
* Signing service
* Private modules - one customer asked to develop a specific reporting service with a more extend functionality than basic (charts, documents export and some other)

# Other

Внешний гейтвей обязательно ддос защита

Вебсокеты, дешевле htpp, но много возни + отваливаются, + 1 сервер не может много сокетов держать

Клиентские очереди сообщений

Хэлсчеки для микросов

CQRS Command–query separation , если бальшая нагрузка на отдельный тип действий

Репликации

Отказоустойчивость

Ретраи

Сперва ориентируюсь на бизнес

Потом цены, сроки, ресурсы

Paul

Costs

Вьюхи

безопасность

редабилить

реюзабилити

латенси

GDPR

System compliency

SADA

proxy + Firewall + gateway

Go Lynx

Get guru