

# Microservices

Proof of expertise

|



# Foreword

In order to “digitally transform” your business, your processes (internally, with employees and contractors) need to reflect a new way of doing business and an embrace of the tech stack.

Externally, you need to make sure your systems are easy and intuitive to use for an end-user. No one likes to land on a website or app and think it was built in 1994.

One way to think about upgrading your processes and systems is through microservices.

Many global names, forward-thinkers such as PayPal, Netflix, LinkedIn, Uber, and the greatest store in the world- Amazon, have turned to new modern practices, employing microservices and cloud technologies and adopting agile methodologies.

Ever since our foundation in 2007, we have worked on numerous projects for clients for different industries. All of them were looking for a partner who has in-depth knowledge of digital transformation and the latest technology trends. This document will address several case studies and real-life examples of microservices implementation.

## Driving factors for the microservices architecture market:

**Digital transformation**

**Customer-oriented businesses**

**Proliferation of connected devices**



# Designing microservice architecture for leading German automotive company

Our customer provides the best products in his business area (automotive industry), and with that, the best software is needed to be ahead of the competition and define new trends. Previously they had developed a very large portal for customers with many advanced features.

Features were very attractive to users, but due to the size of the code and large development team, it wasn't easy to manage such a huge system where all different features are connected. It was challenging to separate features by teams and to work on the same codebase.

Also, there was much information retrieved from different systems many times, which caused some difficulties in performance. Due to all these issues, it was decided to redefine architecture and create a new customer portal.

Microservices architecture approach was chosen and biggest challenge for customer was lack of expertise and engineering capacity to design and implement microservices based application.

Our customer is a leading German manufacturer of luxury cars that sets global trends in automotive industry development. Their broad portfolio of vehicles includes all categories of personal vehicles and trucks. With customers and sales offices in all parts of the world, they have a global presence.



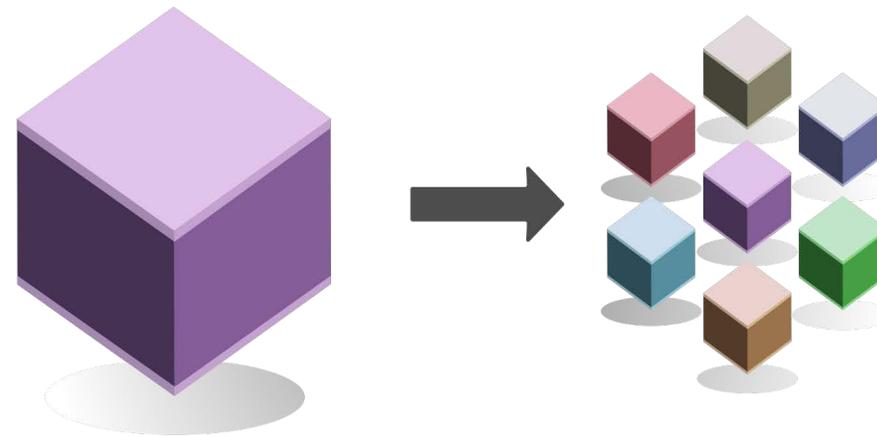
# Responsibilities

I

Serengeti provided development and maintenance service for new features for the existing customer portal. The existing customer portal was a monolith application developed by another company.

After we analyzed the existing portal and got familiar with business logic, we designed microservices architecture for the new customer portal. After that, the team started to work on developing a new portal based on microservice architecture.

All existing features were split into small applications and developed from scratch for the new platform.



# Solution

|

The first step was to form a new team. For the first two months, two Serengeti ambassadors were at the client's offices, studying the current situation of projects, work mode, and business issues.

During the first two weeks, ambassadors were reading documentation and getting familiar with the project. Afterward, they joined the scrum team and started to work on the first tasks: fixing bugs and developing new features for the existing portal.

After the onsite period, most of the work was done from Serengeti office in Zagreb, with the arrival of Serengeti team lead every three weeks to scrum meetings at the client's office. Serengeti's team lead needed to hear new stories directly from the product owner to transfer all required information to the team.

Returning to Zagreb was the most challenging part for ambassadors since they were not in the room next to their colleagues. It was essential to keep successful communication. They managed to do it by having regular meetings and very intense communication together with strict and transparent processes on the Jira board.

One year later project for portal migration to new microservices architecture started. Serengeti engineers used their knowledge of the system they were working on and knowledge of microservice architecture to design and prepare architecture for a large number of small applications. The idea was to split the existing large monolith into many small applications but keep the same features in the new system.

All applications were packed into Docker images. Images were uploaded to the cloud provider registry, and a container cluster was created. Docker containers are then deployed to the cloud. Containers communicated and worked together.

Industry-recommended standards and tools are used to organize the team and manage the project - Scrum & Kanban, Jira, Confluence, GitLab, unit testing, CI / CD, etc.

# Result

Within few months from the start of cooperation, Serengeti developers were completely into the project and worked equally with other team members.

## **The use of modern microservices technologies made the application scalable and flexible.**

With help from Serengeti, the customer implemented a proven and reliable microservices-based application on time. The team tracked applications usage and, depending on that, scaled needed memory for every app and feature. During development, different cloud providers were used depending on client contracts and decisions. For developers, it was easy to deploy and configure applications for other cloud providers. Usage of Docker also made those changes easy.

After the successful first phase of co-operation, the Serengeti team has expanded. Our team was in charge of the complete development of different applications. Backend developers worked on the backend part of the application in charge of requests and data retrieval, while frontend developers worked on the user interface.

All applications developed by our team followed similar architecture, and it was not difficult for developers to switch between different applications. Successful co-operation and team cohesion were proven in this co-operation, and Serengeti established a long-term partnership with the customer.



# Migration to Microservices Helps Digital Enterprise Transformation

|

Our client is a strong and well-known international financial group, a large banking group from Austria.

They offer their customers a full range of banking and other financial services through the following entities: real estate savings bank, leasing, voluntary pension fund management, pension insurance, investing, consulting, factoring, and bonuses. Its subsidiaries in Central and Eastern Europe cover de facto the entire region.

More than 16 million customers are serviced through 2.200 business outlets in corporate and private customer segments and investment banking.

The client had a vision of transforming the bank into a digital enterprise by improving the effectiveness and performance of products, services, and processes. The goal was to customize the bank according to the needs of new generations of users and new business ecosystems.

|



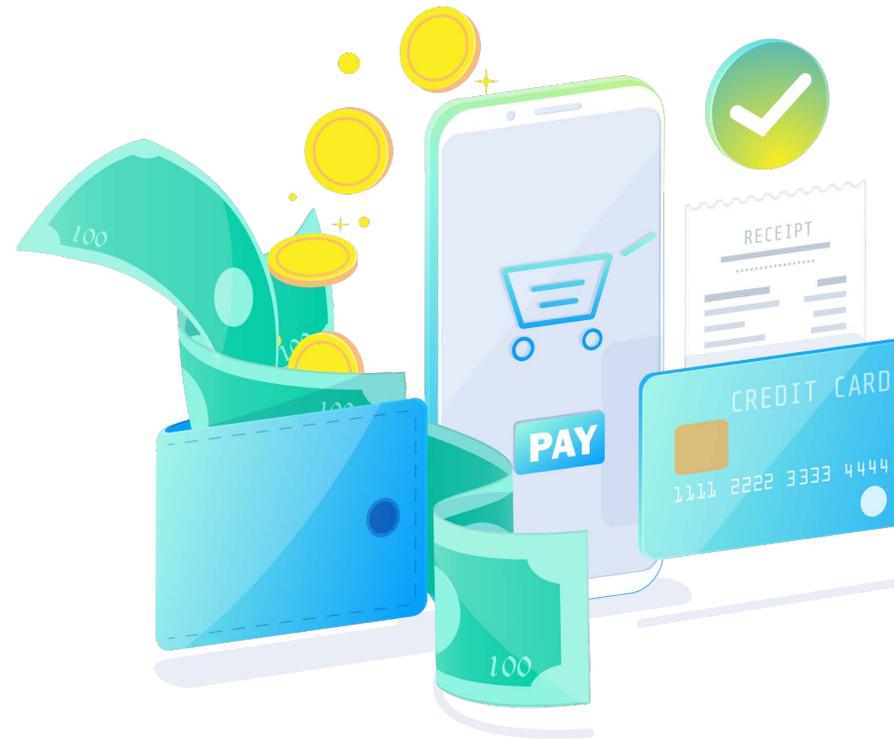
# Responsibilities

|

The client had a vision of transforming the bank into a digital enterprise by improving the effectiveness and performance of products, services, and processes. The goal was to customize the bank according to the needs of new generations of users and new business ecosystems. Apart from increasing flexibility, it is essential to enable the integration of the bank with the new EU directives in short periods. Reaching this speed and agility level is not possible with legacy IT systems and monolith architectures. Introducing modern technologies and tools, new flexible models, design principles, and agile methodologies to development is critical in supporting the goals mentioned above.

Implementing modern technologies like microservices requires specific know-how and engineering capacity.

Our client needed support from a partner that already has experience in microservice architectures to help with design and implementation.



# Solution

I

Serengeti provided microservice architecture design services and implementation services.

We developed a microservice platform for the needs of the client's mobile applications and the incorporation of various EU directives into the banking system integration.

Serengeti assigned its experts to existing client teams according to the proven Team Extension Model to accomplish these goals. The Team Extension Engagement Model is a step-by-step process that ensures fast and efficient onboarding and reaching the external team's optimum productivity level very fast. Together, Serengeti's team and the internal team worked on the overall platform architecture design and implementation.

Microservice architecture has proven to be the best solution because it enables continuous delivery and deployment of large-scale complex applications. It also allows the organization to upgrade the technologies it uses easily. Scaling applications is much easier than scaling large monolithic applications because microservices can be scaled individually.

Another advantage of this approach is that different teams can independently develop and deploy all microservices without coordinating with other teams. We have solved business problems like fast delivery and reuse of standard parts by using modern cloud technologies.

The OpenShift ecosystem delivers various technologies, from monitoring container metrics and application performance inside a container to supporting different Java and other technologies and software environments. Some of the services offered by the OpenShift platform to the applications running on that platform are high availability/load balancing, single sign-on, a unique system for collecting, accessing, and managing application logs, a CI/CD system for creating and installing applications, and service discovery. All these services were used in the overall system design.



# Result

|

By making sound design choices and implementing the best practices of microservice architecture, many long-term benefits were achieved, such as improved communication across teams, improved collaboration and communication efficiency, independent implementation of new features, independent scaling, failure, and resource isolation, easier maintenance, and potential heterogeneity since developers were free to pick the language and stack best suited for their service.

**Serengeti provided the client with a team of experts with broad experience in the banking industry and microservice architectures. This allowed the client to get access to the right skillset and know-how fast. It also enabled the Serengeti team to quickly reach optimum team productivity, with little engagement needed from the client's key technical people.**



# MPSI | eXact in microservices

Mercury Processing Services International (MPSI), a company established in 2009, provides payment solutions to clients from the financial and banking sectors across Central & Eastern Europe, the Middle East, and North Africa.

Located in Croatia and Slovenia, the foundations of MPSI rely on complementary strengths of the two strongest cards businesses, Privredna Banka Zagreb and Banka Koper, and their transition from local companies into a fully international organization.

Its specialties are Payment Processing Services, Issuing Solutions, Acquiring Solutions, Payment Acceptance, and Value Added Services.

MPSI uses a wide range of technology when developing its services, and, as an enterprise organization, it systemizes departments to maximize the efficiency of the system itself.

Software development is organized in the following departments: architecture, development, build management, application management, testing, data warehouse, etc.



# Responsibilities

|

An internally developed back-office application within MPSI, eXact, provides functionalities to support Merchant and Card Payment Processing activities.

The main features of eXact include:

- Web portal integrating acquiring, issuing, clearing and settlement authorization, authentication, mobile and fraud case management functionality
- Online reporting
- Generated output batch interfaces available in the eXact web portal, available for manual or automatic download
- A set of common input/output interfaces across markets (institutions)
- Modules/batches separated for issuing and acquiring transaction processing web services provision (supporting various technologies; that may or may not be Client specific)
- Provision of file stores (not content management; that may or may not be Client specific)



# Solution

|

eXact was designed as a monolithic application. MPSI gained new partners, and their transactions needed to be processed. Scalability and maintenance of the current system started being an issue, so it was decided to transition from monolithic to microservice architecture. MPSI found a reliable partner in Serengeti to support them in this challenging project.

Experienced team members from Serengeti combined with architects from MPSI communicate daily, aiming to design and develop eXact in a microservice architecture, following state-of-the-art techniques and best practices. Depending on the type and need of a microservice, both relational and NoSQL databases were used.

Microservices were developed in .NET Core, containerized in Docker, deployed on RedHat's OpenShift platform (Kubernetes, Istio), and are communicating with each other via async event-based communication based on IBM MQ and Kafka.

To support MPSI during this important project in the best possible way, the Serengeti team needed very competent, experienced, and research-oriented engineers. Our developers were quickly ready to jump in and start working on the product with the quality of developed and documented microservices and the level of details in written documentation.

The project is still ongoing. So far, the Serengeti team has developed over 20 microservices.

Our team completed, among other things, the StandIn module on the acquiring side. The focus is now shifting to the issuing side, where new challenges await. This is a long-term project in which architectural design is crucial, so everything has to be (and is done) in an exact and well-thought-out way.

|



# Closing Words

Simply put, microservices architecture is when a developer works on software as a series of independent, small, modular services.

Each service runs a particular process and communicates with a lightweight mechanism (often an HTTP resource API) to serve an end business goal. You can think of it as a software architectural pattern-breaking interfaces down into APIs designed with one particular purpose. When building a database, a software team might be tasked with capturing first name, last name, phone number, etc. They could create each user field out as a microservice; then, if someone else internal to the organization needed just phone numbers, they could access that - as opposed to call out for all data.

This approach contrasts with a monolithic architectural style, where everything is built as a single unit. Monolithic approaches to software worked for generations, but as more applications deploy to the cloud, microservices have gained in popularity among developers. Change cycles are often tied together so in monolithic architecture, one small change requires an entirely new complete unit.

When talking about microservices, maybe it's the best way to observe simple examples such as this one with the flight-booking application.

People search for flights many times before they book one. Indeed. So the load in the search module is multiple times higher than the load in the reservation module. When you implement microservices, you can configure multiple instances of the search service with multiple instances of the reservation service. Based on load, specific microservices can be increased or decreased, making the overall application perform much better.

## What you need is expertise

And many have the expertise, but you need to find the right fit and someone (or a team) that understands the interplay of all the technical factors driving growth right now. That's a common reason companies will do outsourced software development - because they want to find a team with diverse expertise and backgrounds to drive their business forward too.

Let us know if you need help with implementation of microservices. We would be happy to help you.

## Microservices characteristics:

High Code Quality

Small and Lightweight

Message-Based

Stateless

Single Business Capability

Independent Team

Independently Deployable

Automated Build and Release

Event-Driven



# Who are we?

|

An international nearshoring and offshoring software development and consulting company. With our unique Team Extension Model, we help companies with each stage of the development lifecycle, not only with development but also with consulting services. Technical consulting in different domains like Architecture Design, Automated Testing, DevOps and SCRUM was recognized in 300+ projects across different industries such as finance, logistics, hospitality, industrial manufacturing, healthcare, energy, and retail.

Serengeti d.o.o.  
Ulica Pile I. br. 1  
HR – 10000 Zagreb, Croatia  
sales@serengetitech.com  
serengetitech.com

