

Markus Hartung

Hartmark IT AB

Lead developer

Email: mail@hartmark.se

Senior .NET engineer with strong experience building scalable platforms and modernizing legacy systems into cloud-native architectures.

He has extensive experience in modern architectural patterns and microservice-based solutions, mainly within the banking and finance sector.

Markus is also skilled in automated testing, with solid experience in Test Driven Development (TDD), helping ensure high quality and robust solutions.

Throughout his career, he has worked with a wide range of technologies and tools, and is quick to adapt to new systems and challenges.

He is technically curious and enjoys exploring new areas, including how AI can be applied in real-world development workflows and systems.

Personality-wise, Markus is driven, responsible, and unpretentious.

He works well both independently and in teams, and is often the go-to person for solving complex technical problems.

Key skills

- .NET / C#
- Azure & cloud-based solutions
- Microservices & modern architectures
- REST APIs & system integrations
- Test Driven Development (TDD) & automated testing
- DevOps, CI/CD & Azure DevOps
- AI-assisted development (e.g. Claude, Copilot, local LLMs)
- Full-stack development (React)

Experience

Skandikon October 2024 - April 2026

Skandikon specializes in pension administration, providing digital platforms for employers and individuals to manage and track pension-related data.

Markus works as a full-stack developer in a team responsible for both the partner portal and individual user portals. The partner portal supports employers in managing payroll and sick leave reporting, while the individual portals provide end users with insight into their pension savings.

The team also handles integrations with external systems, including the Swedish Social Insurance Agency (Leka/Lefi), managing critical data exchange related to sick leave and reporting.

The work involves both maintaining legacy systems and driving the transition toward a more modern, scalable architecture, improving performance, user experience, and integration capabilities.

In addition to development, Markus is part of the architecture group, where he helps guide the technical direction of the platform. He provides hands-on leadership, supports other developers, and promotes best practices around clean code, sustainable architecture, and long-term maintainability.

He has also been responsible for driving test strategies, coaching the team in Test-Driven Development (TDD) and building effective automated testing to ensure high-quality, reliable systems.

Markus has also explored practical use of AI in development, including building an MCP endpoint to support agent-based workflows for retrieving and structuring data from underlying systems. He has worked with concepts such as RAG and agentic flows to enable more efficient data access and automation.

.NET / .NET 6+, C#, React, REST, Microservices, Microsoft SQL Server, Azure DevOps, CI/CD, TDD, DevSecOps, Git, AI (RAG, Agentic flows)

Handelsbanken April 2021 - June 2024

At Handelsbanken, I worked as a full-stack developer driving both feature development and modernization of a large advisory support system, including migration of legacy components onto a newer technical platform. I was heavily involved in system-critical integrations with external systems using REST and IBM MQ, ensuring stable and reliable communication across services.

I also contributed to improving engineering practices by supporting initiatives around test automation and DevOps streamlining, with a strong focus on quality and delivery speed. As part of a two-year project, I played a key role in delivering a document signing solution that enabled business systems to initiate and manage customer signing workflows end-to-end.

Security and DevSecOps were a continuous focus area where I actively pushed for better practices and awareness across teams.

In addition, I developed new microservices for expanding functionality and led migrations of existing capabilities into the new architecture.

On top of the formal role, I acted as an unofficial team architect, taking responsibility for architectural consistency and making sure the external vendor delivering the signing solution didn’t break anything during updates. I designed and enforced a smoke test process that effectively served as a go/no-go gate before production deployments.

.NET framework, .Net 5+, git, rest, Microsoft SQL Server, TDD, React, Azure DevOps, SAFE, Rest, Microservices, JSON, DevSecOps

KPA Jan 2019 - April 2021

At KPA, the continued assignment involves working with the same system as before, but now KPA is focusing more on a technical transition where many old systems are being decommissioned and rewritten into new, modern solutions. A significant part of the effort has been migrating applications to Git and .NET Core, as well as developing new applications using modern web solutions with React. Markus has been instrumental and leading in implementing test automation for REST services and creating pipelines for builds and releases in Azure DevOps.

.NET, .Net core, git, rest, Microsoft SQL Server, TDD, React, Azure DevOps, SAFE

SPP Sept 2018 - Dec 2018

At SPP, a provider of private occupational pension solutions, I worked in a team of eight developers within the “Business & Brokers” domain. The team was responsible for a distributed system built on multiple microservices, supporting payroll reporting and salary exchange services used by corporate finance departments and external brokers managing large enterprise clients.

My work focused on developing new features on the platform, fixing defects, and gradually decommissioning and modernizing legacy components as part of ongoing system evolution. This included working across both backend and frontend parts of the system in a complex, service-oriented architecture.

During my time at SPP, I also gained exposure to Lumera, a leading policy administration system widely used in the Nordic life and pension insurance market, which broadened my understanding of large-scale pension administration ecosystems.

NET, ASP.NET, Microsoft SQL Server, WCF, WebForms, Web API, REST, JSON, XML, Angular, React, stored procedures, Scrum, and SAFE

KPA Sep 2016 - Sept 2018

KPP is the system used for Swedish municipalities to report the salaries and receive invoices for pension payments. KPP is available in two versions, a mainframe platform and a modern .net platform. The mainframe is being phased out. Version two is being developed and maintained by Markus and 80 other persons split into 8 teams. Most of the municipalities migrated to the new system and when the missing features are in place the rest will migrate around fall of 2018. Markus' primary focus has been responsible for writing code for calculating the premiums so that pensioners receive the right pension payments.

Furthermore, Markus have also worked to modernize the code to migrate from a monolith to a more modern service oriented system. This allows for gradually modernizing of code so that it becomes more maintainable and new features are easier implemented in the future.

.NET, ASP.NET, C#, .NET Core, Entity Framework, flat files, Microsoft SQL Server, TDD, Scrum, Store Procedures, WCF, XML, Git, TFS, Octopus deploy, WebForms, WinForms, WebAPI, Rest, Swagger

Paternity leave June 2015 - Aug 2016

Scania August 2013 - June 2015

The customer needed a senior software developer who could take responsibility for introducing TDD and help automate testing and implement a CI process.

Markus was part of a large project with six development teams working on a solution to collect and analyze data from the Scania trucks and manage data subscriptions. The system is developed on the .NET platform and Markus worked a lot with the integration layer used the WCF technology. The solution is large and complex, and the product of Markus team consists of 3 solutions and a total of about 300 projects in Visual Studio, which is shared between 3 development team. Markus had the role of senior developer and was responsible for unit tests and build servers. The project went from manually testing everything to fully automated tests.

.NET, ASP.NET, C#, Microsoft Enterprise Library, Microsoft SQL Server, Microsoft Visual Studio, Model-View-Presenter, Preforce, .net Core, ReSharper, Rhino Mocks, SOAP, Store Procedures, TDD, WCF, WCF Blue, XML, Selenium, CodedUI, WebForms, WinForms

Forsmark, Vattenfall February 2013 - July 2013

The customer needed new features and improvements in its proprietary process control system for managing plant modifications.

Markus added new features to the existing process control system. This system was used at Forsmark nuclear facility to monitor renewal processes of the facility. The application is a client-server application which consists of a Windows client designed in WPF and a server part with web services in WCF and an Oracle database for persistency.

.NET, Auto Mapper, C#, Devart dotConnect for Oracle, Entity Framework, Expression Blend 4, Infragistics NetAdvantage WPF, Microsoft Enterprise Library, Microsoft Visual Studio, MOQ, MVVM Light Toolkit, Oracle, TDD, WCF, WPF, XAML, XAMLPad, XML, Resharper

Paternity leave in April 2012 - Jan 2013

Sirius International in March 2011 - Apr 2012

Sirius International is a reinsurance company and they needed a developer to help implement needed functions for the application so they could phase out their old underwriter system.

Markus worked with the development of a risk-simulation system for Sirius, one of the world's largest reinsurance companies. The system had a SOA-based architecture and consisted of a client developed in C# and WPF, and a service layer that was developed in WCF. The project was an ongoing project which Markus joined after they had been working with it for almost a year. He was one of the developers in a team of 15 developers, 4 testers and 2 requirement specialists.

.NET, C#, Fluent NHibernate, LINQ, Microsoft SQL Server, Microsoft Visual Studio, Resharper, Scrum, TDD, WCF, WPF

Mobyson January 2011 - March 2011

The customer needed a developer to maintain and develop new features for their public website.

Markus came late into the project and was the sole developer and was responsible to fix bugs and develop new features for their public website.

.NET, ASP.NET, C#, CSS, JavaScript, LINQ, Microsoft SQL Server, Microsoft Visual Studio, ReSharper, Subversion, WebForms

Regeringskansliet Nov 2010 - January 2011

The customer had a need for new sites in their SharePoint environment.

Markus was part of a team of two people who developed two systems for the State Department (Regeringskansliet). One system handled the legalization of marriages and so-called. "No Objections" document for people seeking visas / show abroad. The other system handled orders and price estimates for document translation with the help of authentication and document management in SharePoint.

.NET, ASP.NET, C#, CSS, HTML, Microsoft SharePoint, Microsoft Visual Studio, Subversion

PPM / Pensionsmyndigheten January 2008 - November 2010

PPM is a part of the public pension system. PPM was fusioned around 2010 together with Försäkringskassan's pension bits and formed the new department Pensionsmyndigheten.

The customer needed a good developer for the writing new functions for the fund trading system.

Markus worked with the development of the existing system that manages the fund trading and account information. Everyone working in Sweden gets a part of their salary in this public pensions savings system therefore the system handled at the time around 6 million pension savers.

Markus worked with:

- Development of system to extract test data from the production environment to test and development environments. This data would also depersonalised in the test and development environment.
- Development of system to enable external stakeholders to make so-called mass changes for the people that have given this actor permission.

CSS, CVS, Eclipse, Emacs, Java, JavaScript, Oracle, Perl, Red Hat, Scrum, SOAP, Spring, Subversion, TOAD, Ubuntu, XML, XSLT, WinForms

GE Money Bank in January 2007 - Jan 2008

The customer needed new web applications for multiple GE Bank branches across Scandinavia.

Markus was part of a development team of about 15-20 people who developed new web applications using EPiServer. During the assignment multiple partner sites were created for the Norwegian branch, Danish branch and Finnish branch. The web applications also contained features for applying for credit loans and regular loans to finance shopping online. Markus also worked with the following features:

- Application forms for loan applications
- EPiServer solution for error handling for a Finnish site
- Code design EPiServer templates and underlying logic for an loan application on a Danish site
- Created reusable components for loan applications
- EPiServer templates and underlying logic
- Optimization and patching of WebServices.

.NET, ASP.NET, C#, COM+, WCF, CSS, EPiServer, JavaScript, Microsoft SQL Server, Microsoft Visual Studio, SOAP, VB.NET, WebService, Visual SourceSafe, WebForms, WinForms

DICE July 2006 - Oct 2006

Worked in team with 10 people. The team built a prototype of a game idea that had the working title Takedown. This game later became Mirror's Edge.

Markus was a gameplay programmer. This meant realizing the vision of game designers into a concrete feature in the game. For example, handle so when user presses jump the avatar in game jumps and also jump up and grab an edge to climb up.

The game engine used was the Unreal Engine.

C++, Microsoft Visual Studio, Perforce, Unreal Script

Courses and certifications

Developing Windows WCF 2016

Course at Addskills called "Developing Windows Communication Foundation Solutions with Microsoft Visual Studio"

MCSD 70-483 Programming in C# 2013

Certification by Microsoft

Certified ScrumMaster 2013

Certification by the Scrum Alliance

Education

Informationsteknik (MSc) 2001-2006

Chalmers University

Languages

Swedish

Native

English

Fluent

German

Basic knowledge