



# What the Certified .NET Architect can do for you.

System development is dangerous, but a certified .NET architect can help you to lessen the risks.

As early as 1994 the Standish Group established that 31% of all projects are terminated before they are finished. Close to 53% costed on average 189% of the original cost estimate. Only around 16% of all projects are completed, according to the investigation, in time, within the budget and with the functionality agreed upon, except for in large companies, in which only about 9% succeed. The typical case, shared by 52.7% of all the cases, was a project that exceeded the time limit, exceeded the budget and furthermore offered up to 49% less functions than were agreed upon.

In a more recent investigation the Gartner Group is reported to have arrived at similar results.

Standish stated that only 4.3% gave *insufficient technical knowledge as a reason for the failures. Instead it was insufficiently specified demands, insufficient participation of users, unrealistic expectations, changed demands and specifications and inadequate planning* which in over 52% of the cases were given as the main reasons.

*The greatest value in engaging a certified .NET architect lies in their education being directed towards just those factors that according to the Standish investigation are often underestimated in development projects.*

## Overriding business demands

A certified .NET architect understands the value of separating overriding business demands from more detailed application- and user requirements. The initial steps in the architectural process the architect has learnt to follow consists of producing and documenting, in collaboration with the business architect, these overriding demands and placing the contemplated system into its operational and businesslike context.

## Detailed user- and application requirements - conceptual design

With a starting point from the overriding business demands that have been documented, the architect can play an important part in the collection and documentation of detailed user- and application requirements. This work includes making a detailed analysis of the current business processes, of the current information structures and of the relevant rules and regulations. All this takes place according to the overriding business demands.

A final result of this step is a conceptual “exterior” architecture of the contemplated system. The contact surface for every incoming service and system component is accurately described so that someone – the architect or someone else – in the next step can establish a detailed internal design for every component individually.

## Logical design

When the outside of the system component has been accurately described the architect or a system designer can establish the architecture and overriding design that is needed in order to implement the component. The foundation is then already laid by the architect and we begin to approach the technology. The architect can help you even here with high competence and access to different architectural patterns and reference architectures. But it is in the initial stages that the architect can be of most use to you, for it is there above all that the conditions for a successful project are decided.

**By all means use a certified .NET architect for the whole project, but start using him or her in as early phases of the project as possible!**

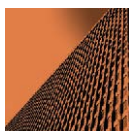


## What is a Certified .NET Architect?

A certified .NET architect has undergone a thorough architect's education of four intensive and qualitative three-day courses. By virtue of this education the architect has an in-depth knowledge of an efficient architectural process fully compatible with generally occurring development processes as for example Rational Unified Process (RUP). The education is designed to give those of you who are using a certified .NET architect the following values:

- Decreased risk in your development project – see overleaf.
- Systems which are designed to achieve defined operational and business goals.
- Systems which to a higher degree than usual are based on your operational and business processes.
- Systems which are flexible enough to be adapted to changed operational and business processes.
- Systems which have the potential of giving your coworkers better access to more relevant information about the business and who are aware of the rules and regulations which contribute towards guaranteeing high quality of the information elements that the system has an effect on.

Both the education and certification is a collaboration between Sundblad & Sundblad and Microsoft Sweden. It is Sundblad & Sundblad who conducts the education; the certificate is signed by Sten and Per Sundblad together with a person responsible at Microsoft Sweden. Here follows a short account of the contents in each and every one of the four three-day courses:



### **Patterns for Architecture and Design**

This course gives the participant a knowledge of how architectural and design patterns can help the architect towards a better and more reliable result.



### **Software Architecture and Architect Roles**

An overriding course which gives a knowledge of different roles of architects and different responsibilities of architects. The participant learns how to combine the one formal standard – IEEE 1471-2000 – which exists for architectural description of software-intensive systems with an architectural framework which helps the architect to structure the architectural process and to document a solution architecture from the perspective of different interested parties.



### **Vision, Information and Requirement Analysis**

This course teaches the participant, together with heads of operations, users, enterprise architects, analysts and others, to collect and document information- and application demands for a contemplated system. The final result is a conceptual model of the whole system, which means a division into components and services (for service-oriented architecture – SOA) together with an accurate external description of each component/service.



### **Design of Service-Oriented .NET Solution**

Thus course starts out from the conceptual model of the system that was produced in the course Vision, Information and Requirement Analysis. The goal is to achieve an overriding logical design model for the inside of each system component or service. When the architect works towards this goal, he or she enlists the aid of reference architecture and of patterns from the pattern course.