



SOFTWARE ENGINEERING STANDARDS & PRACTICES

C# .Net Best Coding Practices

9AAD134037

Department	GF-IS ADM Applications Performance Excellence (APE)
Approver	Giulio Bitella, Global Department Manager
Owner	Tomasz Jastrzębski, Global Leader for Software Engineering (SE)

For the latest distributable version of this and other Software Engineering standards please visit this [link](#) to ABB Library.

WHAT IS THIS?

This document presents set of the programming standards and practices specific to .Net and C# language. For general Best Coding Practices refer to document [9AAD135446](#).

WHAT IS THE PURPOSE OF THIS DOCUMENT?

The purpose of the document is to establish single, complete C# programming standards and practices reference, containing both references to applicable Group-IS standards like InfoSec and Enterprise Architecture (EA) publications as well as practical standard implementation details/guidelines and the recommended software development practices.

TABLE OF CONTENTS

WHAT IS THIS?	I
WHAT IS THE PURPOSE OF THIS DOCUMENT?	I
INTRODUCTION	1
PLATFORMS	1
WEB APPLICATIONS.....	1
DATA ACCESS	1
UNIT TESTS	2
LOGGING AND TRACING	2
REFERENCES	2
RECOMMENDED READING.....	2
REVISION HISTORY	3

INTRODUCTION

This document addresses .Net C# specific standards only and is continuation of the Best Coding Practices document (9AAD135446) which covers:

1. platforms
2. design principles
3. web applications
4. web/api services
5. mobile applications
6. cloud services
7. authentication and authorization
8. data access
9. unit tests
10. logging and tracing
11. use of open source software
12. group standards and policies

PLATFORMS

1. ASP.NET Core is the most preferred framework. (ref. 9AAD135404)
2. Microsoft SQL Server is the most preferred storage engine for tabular and relational data. (ref. 9AAD135312, 9AAD135311)

WEB APPLICATIONS

1. Preferred framework is ASP.NET Core MVC. (ref. 9AAD135404)
2. Model-View-Controller (MVC), Single Page Application (SPA) and Razor Pages are all valid web application architectures, selection shall be made based on project profile.
3. Whenever feasible, Web API methods should be implemented as asynchronous and accept `CancellationToken`. Making `async` method calls configure it properly – use `ConfigureAwait(false)` method wherever applicable, refer to “C# 7.0 in a Nutshell”, Chapter 14: Concurrency and Asynchrony.

DATA ACCESS

1. Use Entity Framework Core (EF Core) for data access whenever possible and appropriate Non-core EF is second, albeit a valid choice.
2. Whenever Entity Framework does not provide sufficient functionality, use standard .Net framework classes available in the `System.Data.SqlClient` namespace. Avoid using other third-party solutions. Albeit convenient to use, they tend to make excessive database calls resulting in performance degradation.

3. Use `async` calls to external resources (DB, backends) when multiple calls are required with no sequential dependencies, pass `CancellationToken` object, configure `async` method properly – use `ConfigureAwait(false)` method wherever applicable – refer to “C# 7.0 in a Nutshell”, Chapter 14: Concurrency and Asynchrony.
4. Make sure Database Context object is always properly disposed to ensure database connection is promptly returned to the connection pool and can be immediately reused.

UNIT TESTS

1. Use NUnit, MSTest or xUnit frameworks only.
2. Follow practices described here: <https://docs.microsoft.com/en-us/dotnet/core/testing/unit-testing-best-practices>, also for .Net Framework (non .Net Core) applications.

LOGGING AND TRACING

Use standard Logging and Tracing mechanisms already available in .Net framework. Any non-standard providers should be configured in application configuration file.

REFERENCES

1. IS Standard for Application Development Frameworks - [9AAD135404](#)
2. Best Coding Practices - [9AAD135446](#)
3. C# Coding Standards - [9AAD134036](#)
4. C# Coding Standards - Field Guide - [9AAD134039](#)
5. SQL Server Coding Standards - [9AAD134842](#)
6. Test Strategy - [9AAD134969](#)
7. Unit Testing Field Guide - [9AAD135249](#)
8. Source Code Management Standards - [9AAD134843](#)
9. User Experience & Design Standards and Practices - [9AAD134800](#)

The latest versions of the above standards are available in ABB Library (<http://library.abb.com>)

RECOMMENDED READING

1. Albahari, J., & Albahari, B. (2018). *C# 7.0 in a nutshell*. Beijing: O'Reilly.
2. Martin, R. C. (2016). *Clean code: A handbook of agile software craftsmanship*. Upper Saddle River, NJ: Prentice Hall.
3. Skeet, J. (2018). *C# in Depth*. Manning Publications Company.
4. Watson, B. (2018). *Writing High-Performance .NET Code* (2nd ed.). Ben Watson.

5. Aviva Solutions. (n.d.). *C# Coding Guidelines*. Retrieved from <https://csharpcodingguidelines.com>

REVISION HISTORY

Rev.	Page	Change Description	Author(s)	Date
A	all	approved	Tomasz Jastrzębski et al.	2019-04-09

Dev

Information Systems

Applications Performance Excellence Department (APE)

Software Engineering Standards & Practices