.NET Platform Complete Architecture (Microsoft Native Components)

Level 1: Microsoft Standards And Specifications

- CLI Implementation (ECMA-335 Standard Implementation)

    - Microsoft's implementation of the Common Language Infrastructure
    - Defines the execution environment for .NET applications
    - Includes Microsoft's CTS and CLS implementations

- CIL (Common Intermediate Language) / MSIL

    - Microsoft's implementation of the intermediate bytecode format
    - Language-agnostic instruction set for .NET languages
    - Runtime-independent code representation
    - Basis for JIT compilation and AOT compilation

- C# Language Specification

    - Microsoft-designed language specification
    - Primary language for .NET platform
    - Microsoft's reference implementation
    - Compiles to CIL/IL code

- Metadata Specification

    - Microsoft's metadata format for assemblies
    - Self-describing component information
    - Type definitions, references, and attributes
    - Runtime reflection and tooling foundation

Level 2: Microsoft Runtime Implementations

- CLR (Common Language Runtime)

    - Original Microsoft runtime for .NET Framework
    - Windows-only implementation
    - Integrated with Windows operating system

- CoreCLR (Core Common Language Runtime)

    - Microsoft's modern cross-platform runtime
    - Foundation for .NET Core and .NET 5+
    - Designed for performance and modularity

Level 3: Runtime Core Components & Subsystems

- Execution Engine

    - Just-In-Time Compiler (RyuJIT)
    - Ahead-of-Time Compiler (CoreRT / .NET Native)

- Interpreter (in specific scenarios/architectures)

- Memory Management System

  - Garbage Collector (GC)
  - Managed Heap Allocator
  - Large Object Heap Manager

- Type System & Metadata

  - Type Loader
  - Metadata Engine
  - Reflection Subsystem

- Exception Handling

  - Exception Dispatch Engine
  - Structured Exception Handling (SEH) Integration

- Threading & Concurrency

  - Thread Pool
  - Synchronization Primitives
  - Timer Queue

- Security & Verification

  - IL Verifier
  - Security Metadata Checker
  - Application Domains (AppDomain - mainly in .NET Framework)

- Interoperability

  - Platform Invoke (P/Invoke)
  - COM Interop
  - Internal Call Mechanisms (FCall/QCall)

- Diagnostics & Debugging

  - Diagnostic Pipeline (EventPipe)
  - Debugger Services
  - Profiling & Performance Counters

Level 4: Microsoft Base Class Library (`System*` namespace)

- System
- System.Collections
- System.Collections.Generic
- System.IO
- System.Linq
- System.Net
- System.Reflection

- System.Text
- System.Threading
- System.Xml

Level 5: Microsoft Framework Class Library

- Windows Desktop Frameworks

    - Windows Forms - Microsoft's desktop UI framework
    - WPF (Windows Presentation Foundation) - Modern Windows UI framework
    - WinUI - Native Windows UI platform

- Web Application Frameworks

    - ASP.NET - Original Microsoft web framework
    - ASP.NET Core - Modern cross-platform web framework
    - Blazor - Microsoft's WebAssembly-based framework

- Data Access Technologies

    - ADO.NET - Microsoft's data access framework
    - Entity Framework - Microsoft ORM framework
    - LINQ to SQL - Microsoft's SQL integration

- Communication Frameworks

    - WCF (Windows Communication Foundation)
    - Web API (ASP.NET Web API)
    - SignalR - Microsoft's real-time framework

Level 6: Microsoft Development Tools

- Compiler Infrastructure

    - Roslyn Compiler Platform
    - Microsoft.CodeAnalysis libraries
    - C# and VB.NET compilers

- Build and Packaging Tools

    - MSBuild - Microsoft build engine
    - NuGet - Microsoft package manager
    - .NET SDK tools

- Integrated Development Environment

    - Visual Studio - Microsoft's primary IDE
    - Visual Studio for Mac - Microsoft's macOS IDE
    - Visual Studio Code - Microsoft's lightweight editor

- Command Line Tools

- dotnet CLI - Microsoft's command-line interface
- Developer PowerShell modules
- Microsoft build tools

Level 7: Microsoft Cloud and Enterprise Services

- Azure Integration Components

  - Azure SDK for .NET
  - Azure Functions runtime
  - Azure App Service integration

- Enterprise Services

  - Windows Communication Foundation (WCF)
  - Windows Workflow Foundation (WWF)
  - Enterprise Library components

Level 8: Microsoft Platform Integration

- Windows Operating System Integration

  - Windows Registry access
  - Windows Event Log integration
  - Windows Service support
  - COM and COM+ interoperability

- Microsoft Office Integration

  - Office Primary Interop Assemblies
  - Office development tools
  - SharePoint integration

- Microsoft Database Integration

  - SQL Server integration
  - SQL Server Compact Edition
  - LocalDB support

Note: This list includes only Microsoft-developed and maintained components that are part of the official .NET platform. Third-party implementations like Mono, Xamarin (now part of Microsoft but originally third-party), and community projects are excluded as per the request.