

Nasper Developer Guide

[Document subtitle]



**1. Introduction**

**1.1** **Purpose**

This document has been updated to describe the expanded capabilities of the Nasper Solution Accelerator, which generates UI razor pages as per JSON configuration files.

**1.2 Scope**

The Nasper Solution Accelerator now provides tools to generate a comprehensive Front-end system architecture including business logic services and all aimed at reducing development time and enforcing consistency across large-scale projects.

**2. System Overview**

The extended capabilities of Nasper allow for the generation of:

Nethon.JSON

The nethon.JSOn file maintainable Front-end development in the applications.

**3. Architecture**

**3.1 High-Level Architecture**

* **Nasper Solution Accelerator:** NasperCore application that reads various JSON configurations and generates corresponding Razorpages and UI design.

**3.2 Components**

**1. Configuration Loaders**

Parse distinct JSON files for different architectural requirements.

**2. Code Generators**

Generate razor pages for UI.

**4. Process Flow**

**1. Configuration Specification:** Developers specify the architectural components in corresponding JSON file.

**2. Code Generation:**

* The Nasper Solution Accelerator reads the JSON configurations.
* Creates razor pages and generates the UI design stub for specified components

**3. Compilation and Deployment:**

* The generated code is compiled and deployed.

**4. Runtime:**

* The Spring Boot application serves the business logic through RESTful endpoints managed by generated controllers, accesses data through JPA repositories, and utilizes service classes for business operations.

**5. Technologies Used**

* **C#:** Primary programming language.
* **NewtonSoft:** JSON processing for reading configuration files.

**6. Code generator link:**

Download the code from Git hub using below link.



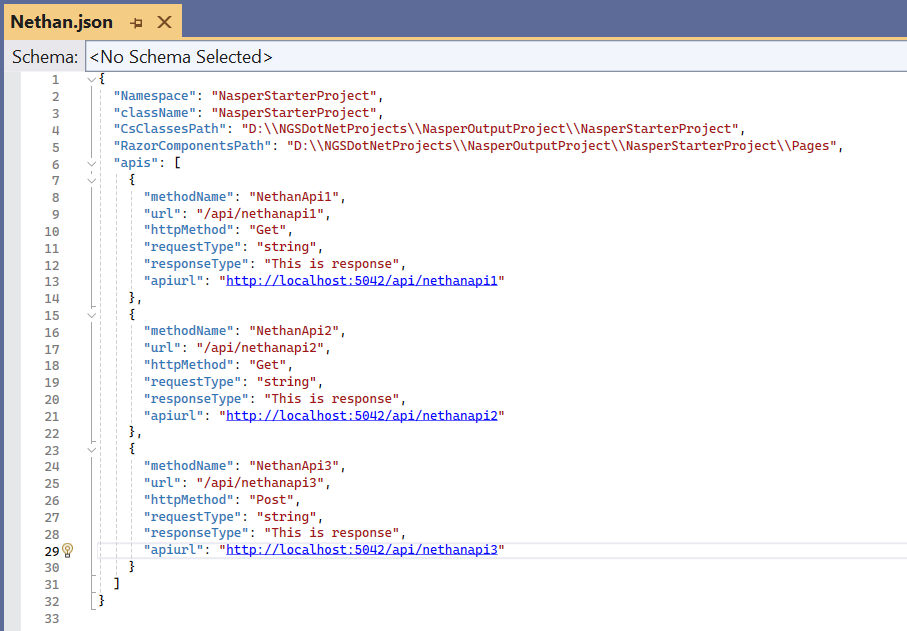
https://github.com/mounikapotla911/NGS/blob/main/NethanCore.zip

**Prerequisites:**

Visual Studio 2022 (Community version)

.Net 8 SDK

Please provide the properties values as mentioned below to generate Razor views.



Namespace: Provide the target project namespace name

className: Name of the class name

CsClassesPath: Destination path of the target application

RazorComponentsPath: Destination path of the target application

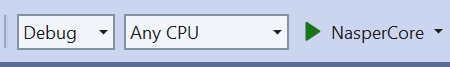
Api: It consists of List of API properties.

1. methodName: Declare the name of the method.
2. URL: Specify the Context path URL.
3. method:  mention the method.
4. requestType: Provide the RequestClass(Data type) based on your requirement
5. responseType: provide the ResponseClass (Data type), based on your requirement.

By default, you can give it as null.

f) apiurl: provide api url with port

Run the NasperCore application once the all properties are configured.



Click NasperCore button or F5 to run the application.

After successful process, the stubs will generate in the target application.