

Microsoft.Extensions.Configuration.Json

August 18, 2026

Microsoft.Extensions.Configuration.Json

Kind: Service

Source: Pams/Web/Pams.Web/Pams.Web.csproj (line 1)

Part of: Pams

NuGet package dependency

Microsoft.Extensions.Configuration.Json is a NuGet dependency in Pams.Web.csproj that adds JSON configuration providers to the ASP.NET application. It reads configuration values from JSON files such as application settings files and exposes them through the .NET configuration system.

Diagram

```
sequenceDiagram
    participant App as Pams.Web Application
    participant Config as Configuration Builder
    participant Json as Microsoft.Extensions.Configuration.Json
    participant File as JSON Settings File

    App->>Config: Build configuration
    Config->>Json: Add JSON configuration provider
    Json->>File: Read configuration values
    File-->>Json: Return JSON content
    Json-->>Config: Provide configuration keys
    Config-->>App: Return configured values
```

Usage

```
type ApiSettings = {
    baseUrl: string;
};

async function loadApiSettings(): Promise<ApiSettings> {
```

```

const response = await fetch("/api/settings");
if (!response.ok) {
    throw new Error("Unable to load application settings");
}

return response.json();
}

const settings = await loadApiSettings();
const response = await fetch(`${settings.baseUrl}/status`);
console.log(await response.json());

```

AI Coding Instructions

- Keep JSON-backed server settings in the ASP.NET configuration pipeline rather than duplicating values in client code.
- Match JSON property paths with the configuration keys read by `IConfiguration` or bound options classes.
- Treat settings returned to TypeScript clients as public values; do not expose connection strings, credentials, or other server-only configuration.
- Update `Pams.Web.csproj` package references when changing the JSON configuration provider version.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Injected or called by (1)

- `Pams.Web` — `Pams/Web/Pams.Web/Pams.Web.csproj` :1