

System.Net.Http.WebRequest

August 18, 2026

System.Net.Http.WebRequest

Kind: Service

Source: Pams/Logic/Pams.Storage/Pams.Storage.csproj (line 1)

Part of: Pams

NuGet package dependency

`System.Net.Http.WebRequest` is a NuGet package dependency declared in `Pams.Storage.csproj`. It supplies legacy HTTP request APIs used by `Pams.Storage` when code creates or handles `WebRequest`-based network calls. Keep this dependency scoped to legacy request paths while newer code follows the system's current HTTP integration approach.

Diagram

```
sequenceDiagram
    participant Client
    participant Storage as Pams.Storage
    participant WebRequest as System.Net.Http.WebRequest
    participant Remote as Remote HTTP Endpoint

    Client->>Storage: Request storage operation
    Storage->>WebRequest: Create WebRequest
    WebRequest->>Remote: Send HTTP request
    Remote-->>WebRequest: Return HTTP response
    WebRequest-->>Storage: Return response data
    Storage-->>Client: Return operation result
```

Usage

```
// The System.Net.Http.WebRequest package runs inside Pams.Storage.
// TypeScript clients call the application endpoint that triggers the storage flow.
```

```

async function getStorageItem(id: string) {
  const response = await fetch(`/api/storage/items/${encodeURIComponent(id)}`);

  if (!response.ok) {
    throw new Error(`Storage request failed: ${response.status}`);
  }

  return response.json();
}

const item = await getStorageItem("document-id");
console.log(item);

```

AI Coding Instructions

- Treat `System.Net.Http.WebRequest` as a dependency of `Pams.Storage`, not as a browser or TypeScript package.
- Keep `WebRequest`-based calls contained in legacy storage integration code.
- Check response status, streams, and disposal behavior when changing .NET code that creates `WebRequest` instances.
- Do not add direct client-side dependencies on this package; expose storage operations through application APIs instead.

Used by

2 references from 2 files. 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 (2)

- `Pams.Email` — `Pams/Logic/Pams.Email/Pams.Email.csproj :1`
- `Pams.Storage` — `Pams/Logic/Pams.Storage/Pams.Storage.csproj :1`