
tyde Documentation

Release latest

Apr 25, 2022

1	Getting Started	3
1.1	What is Tyde	3
1.2	The Setup	3
2	Releases	5
2.1	Alpha	5
3	License	7



A simple easy to use library to manage JWT Sessions for your client services.

Supports * *.Net 6*

GETTING STARTED

1.1 What is Tyde

Tyde is a small library to manage sessions while you consume JWT tokens in your services. If you have ever integrated with a with an API service which requires JWT Session tokens, you know the hustle of writting a session handler to allow your service to run with no interruptions.

Many a time, when creating an integration service we find ourselves, we find ourselves having to create a logic which refreshes JWT tokens or some form of authentication.

This sometimes can create design anomalies. Tyde, tries to solve this problem by abstracting the session management code.

Install the package from [here](#)

1.2 The Setup

Example: Let's say you have a :code: *WeatherService*, and it requires Sessions to be refreshed after every 60 seconds. Given an authentication response like this;

you can manage sessions using Tyde like so;

- Begin by injecting the package to your instance of HttpClient like so;

```
services.AddHttpClient<ITydeAuthService, TydeAuthService>(config =>
{
    config.BaseAddress = new Uri("https://localhost:7157");
})
.AddTyde(opts =>
{
    opts.AuthenticationUrl = new Uri("https://localhost:7157/api/AuthAPI/SignIn");
    opts.AuthorizingParameters = new Dictionary<string, string>()
    {
        { "username", "j0ni" },
        { "password", "sd fdsd" }
    };
});
```

Finally, add *TydeDelegatingHandler* from *Tyde.Core* to the *HttpMessageHandler*

```
services.AddHttpClient<IWeatherService, WeatherService>(config =>
{
    config.BaseAddress = new Uri("https://localhost:7157");
})
.AddTyde(opts =>
{
    opts.AuthenticationUrl = new Uri("https://localhost:7157/api/AuthAPI/SignIn");
    opts.AuthorizingParameters = new Dictionary<string, string>()
    {
        {"username", "j0ni" },
        {"password", "sdfdsdsd" }
    };
}).AddHttpMessageHandler(c => c.GetService<Tyde.Core.TydeHttpDelegatingHandler>()); //
↳mandatory
```

Now, all requests in the *WeatherService* will be Authenticated as need.

Tyde is a simple package which uses the already robust *HttpClient* library from microsoft. Ensuring the library has a small footprint, while delivering a perfect solution.

RELEASES

2.1 Alpha

- 0.1.2 :First Release

LICENSE

MIT License

Copyright (c) 2022 John Nyingi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.