Design decisions

Folder structure and project structure

Folders are organized in a way that is reasonably easy to find everything – in this case mainly sorted by interface implementation.

Use of interfaces

Interfaces allow for polymorphism and greatly reduce amount of code duplication and make the code easier to test.

Console Logger implementation

Console doesn't have any async method equivalent, so we delegate this into another thread through usage of Task.

File Logger implementation

File logger requires a path for the file to be created and will append the message to the file rather than overwrite. Used File.AppendAllTextAsync() over StreamWriter.WriteLineAsync(), since the former has native async file I/O.

In case we run into an error, try catch should be able to write the error to the console using already existing console implementation.

TCP implementation

This is probably most complex implementation as it requires few things before it can be set up properly. It does contain 3 ways of setting up IP End Point through constructor, containing IP address and port to use by the client.

In case we run into an error, try catch should be able to write the error to the console using already existing console implementation.

Main Logger Class implementation

Logger contains a list of loggers, which once initialized, is able to register different loggers at the same time and then log the message based upon what loggers were registered.

Then we await the result to complete all tasks. Console log is a bit of an exception as it's a fallback mechanism to log the errors from other loggers.