Demo

Filebased Apps

1. Python file `print(hello from Python App)`
2. C# file `fileapp.cs`
    ```csharp
    Console.WriteLine("Hello from filebased app.");
    ```
3. dotnet run `fileapp.cs`
4. Show it is compiled: `~/Library/Application Support/dotnet/runfile`
   * Windows: `~\Appdata\local\temp\dotnet\runfile`
5. Args:
    ```csharp
    if(args.Length> 0) {
      string message = string.Join(" ", args);
      Console.WriteLine(message);
    }
    ```
    * `dotnet run fileapp.cs`

Web - Minimal API & SDK's/Packages, etc.

1. New file `webapi.cs`
    ```csharp
    var app = WebApplication.Create();
    app.MapGet("/", () => "Hello, Minimal API!");
    app.Run();
    ```
    * The above will not work, need to say it is web sdk: `#:sdk Microsoft.NET.Sdk.Web`
2. Run it with `dotnet run webapi.cs`and then:
    ```bash
    curl http://localhost:5000
    ```

SSE (Server Sent Events)

1. In `webapi.cs` add a new endpoint:
    ```csharp
    app.MapGet("/sse", async context=>
    {
      context.Response.ContentType = "text/event-stream";
      for (var i = 0; i < 5; i++)
      {
        await context.Response.WriteAsync($"data: Message {i}\n\n");
        await context.Response.Body.FlushAsync();
        await Task.Delay(1000);
      }
    });
    ```
2. Browse to: `http://localhost:5000/sse`
3. Create UI: ask GHCP
    ```text
    ```

    Can you explain to me whet the code in webapi.cs is doing
    ```
    * then:
    ```text
    can you now create a minimal. modern looking web UI frontend for the
/sse endpoint
    ```
    ```