Backlog

Jackson Mowry

<2024-09-03 Tue>

1 Product Name

Jack Sparrow (previously named FARPCs)

2 Backlog

1. Define XML Schema

   - As a user of the tool I would like to have a standardized XML which can be used to ensure my schema files are correct, so that I don't make mistakes when writing them.

2. Parse XML Schema (Messages)

   - As a user of the tool my XML needs to be parsed so that new code can be generated for each of the output languages specified at run time, so that my input changes the output.

3. Parse XML Schema (Procedures)

   - As a user of the tool my XML should be parsed so that server side "prototypes" for each procedure can be defined, so that I have a simple API to work with on my backend.

4. Detect Message Circular Dependencies

    - As a user of the tool I would like to be notifed if any of my messages contain circular dependencies, so that I know what to fix in my schema file.

5. Define Binary Encoding

    - As a user of the tool I would like a diagram/explaination as to how messages will be encoded when sent of the network, so that I can assess the potential benefits of this solution for my usecase.

6. Decide on a Templating Language/Tool

    - As a user of the tool I would like a simple templating tool that will allow me to define now code generation paths for a language/framework of my choosing, so that this solution is more appliciable for my usecase.

7. Prototype Client Code Generation (JavaScript)

    - As a user of the tool I would like to see a simple implementation of what the client-side API will look like, so that I can see if it will fit within my product.

8. Prototype Client Code Generation (Python)

    - As a user of the tool I would like to see a rough implementation in Python so I can begin exploring how the tool will fit into my product.

9. Implement a Simple HTTP Server in C

- As a user of the tool I need to be able to service request via HTTP, so that I can easily integrate with browser based apps.

10. Implement a Ultra Fast HTTP Server in C Utilizing `io_uring`

    - As a user of the tool I need to be able to handle thousands of requests per second with ultralow latency, so that I can continue to grow my product.

11. Prototype Server Code Generation (C)

    - As a user of the tool I would like to see how code is generated for the C language (utilizing a simple HTTP server), so that I can begin fitting the solution into my product.

12. Polished Client Code Generation (JavaScript)

    - As a user of the tool I would like to have a production ready implementation of the JavaScript code to begin finalizing my, so that I can get ready to ship my product to users.

13. Polished Server Code Generation (C)

    - As a user of the tool I need a finalized version of the C server code so that I can finish writing my server side code around the framework.

14. Tutorial to Add New Languages

    - As a user of the tool I'd like a simple tutorial showing how to add support for my language on either the server or client side, so that I can integrate this tool into our custom stack.

15. Direct Communication Over TCP

    - As a user of the tool I'd like to cut out the overhead of HTTP and allow my client to communicate with the server directly over HTTP, so that I can reduce latency.