

Artigo

[Larissa Prussak](#) · Out. 28, 2021 4min de leitura

[Open Exchange](#)

Uma breve introdução sobre como desenhar diagramas com a biblioteca do mermaid

Olá pessoal!

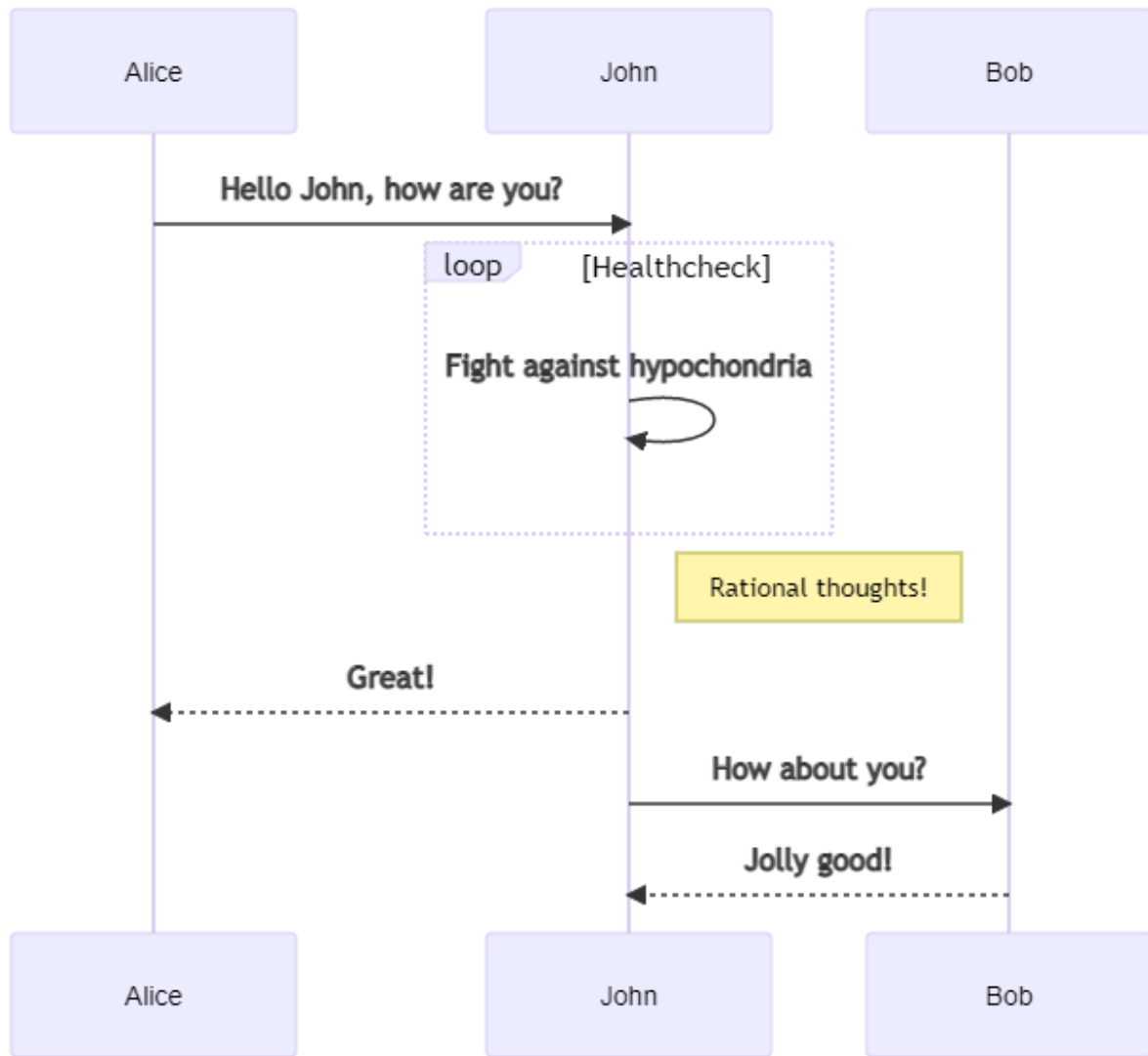
O [@José Roberto Pereira](#) e [@Henrique Dias](#) propuseram uma nova forma de visualizar mensagens no IRIS Interoperability em uma atualização recente do [MessageViewer](#). Nessa atualização, tentaram dar aos usuários uma visualização baseada em um diagrama de sequência UML. Você pode obter mais informações no artigo [anterior](#).

A fim de obter todos os cálculos de geometria complicados para desenhar esse diagrama, usaram a incrível biblioteca de código aberto [mermaid](#) JS. E o que eles gostariam de compartilhar com você neste artigo, é como usar esta biblioteca. Vamos nos concentrar apenas no diagrama de sequência, mas esteja ciente de que essa biblioteca permite que você faça muito [mais](#).

O Mermaid usa uma sintaxe inspirada no Markdown para definir diagramas. É super intuitivo, então acho que é melhor mostrar um exemplo em vez de escrever muito texto chato:

```
sequenceDiagram
Alice->>John: Hello John, how are you?
loop Healthcheck
    John->>John: Fight against hypochondria
end
Note right of John: Rational thoughts!
John-->>Alice: Great!
John->>Bob: How about you?
```

Esta definição permite que o mecanismo do mermaid renderize o seguinte diagrama, diretamente em uma página da web usando SVG:



Esse exemplo foi obtido da documentação do mermaid e você pode experimentá-lo neste [editor online](#). [Existem muitas configurações](#) que você pode tentar nele.

Como você pode ver, na definição do diagrama você só precisa especificar os atores / participantes e quais eventos / mensagens eles enviam uns aos outros.

sequenceDiagram

Alice->>John: Hello John, how are you?

loop Healthcheck

John->>John: Fight against hypochondria

end

Note right of John: Rational thoughts!

John-->>Alice: Great!

John->>Bob: How about you?

Messages

Participants

E tudo o que você precisa para ter o diagrama em sua página da web, é um contêiner div com a especificação do diagrama e um código JS que inicializa o mecanismo mermaid e renderiza o diagrama.

```
<div class="mermaid">
sequenceDiagram
Alice->>John: Hello John, how are you?
loop Healthcheck
    John->>John: Fight against hypochondria
end
Note right of John: Rational thoughts!
John-->>Alice: Great!
John->>Bob: How about you?
</div>
```

```
mermaid.initialize({
  startOnLoad: true,
  theme: 'forest'
});
```

Você pode encontrar este exemplo neste [fiddle](#).

Esta é a base do frontend do trabalho proposto. Para o back-end, tudo o que precisamos fazer é configurar um endpoint REST que recupera mensagens de uma sessão de interoperabilidade IRIS, formatar em um objeto JSON adequado e enviá-lo de volta para o frontend. Como o foco deste artigo é o código de front-end, não prestarei atenção na implementação de back-end, mas você pode verificar nas classes [despacho](#) e [serviço](#)

O back-end envia de volta um JSON como este:

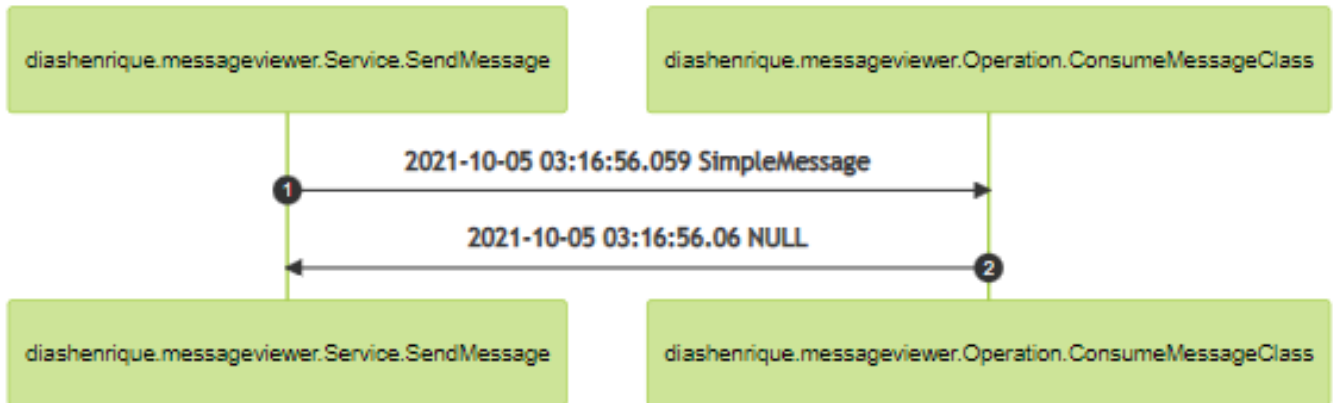
```
{
  "participants": [
    "diashenrique.messageviewer.Service.SendMessage",
    "diashenrique.messageviewer.Operation.ConsumeMessageClass"
  ],
  "messages": [
    {
      "id": "1182",
      "from": "diashenrique.messageviewer.Service.SendMessage",
      "to": "diashenrique.messageviewer.Operation.ConsumeMessageClass",
      "message": "2021-10-05 03:16:56.059 SimpleMessage"
    },
    {
      "id": "1183",
      "from": "diashenrique.messageviewer.Operation.ConsumeMessageClass",
      "to": "diashenrique.messageviewer.Service.SendMessage",
      "message": "2021-10-05 03:16:56.06 NULL"
    }
  ]
}
```

Finalmente, com funções JS simples, você pode transformar este JSON em uma especificação de diagrama de sequência do mermaid, como este:

```
sequenceDiagram
autonumber
```

```
participant P0 as diashenrique.messageviewer.Service.SendMessage
participant P1 as diashenrique.messageviewer.Operation.ConsumeMessageClass
P0->>P1: 2021-10-05 03:16:56.059 SimpleMessage
P1->>P0: 2021-10-05 03:16:56.06 NULL
```

E este é o diagrama de sequência renderizado



Você pode verificar o código JS completo [aqui](#).

Então é isso. Espero que este artigo possa trazer a você algo útil que possa ajudá-lo em seus projetos incríveis.

Até mais!

[#Frontend](#) [#Interoperabilidade](#) [#JavaScript](#) [#InterSystems IRIS](#)
[Confira o aplicativo relacionado no InterSystems Open Exchange](#)

URL de origem: <https://pt.community.intersystems.com/post/uma-breve-introdu%C3%A7%C3%A3o-sobre-como-desenhar-diagramas-com-biblioteca-do-mermaid>