

Curso **SUBIDO** DE TRÁFEGO



Pedro Sobral

MÓDULO - Google Tag Manager (GTM)

API na prática (Parte 2)

SUMÁRIO



00'10"**Introdução****Cap. 1****Pág. 2**

00'47"**API na prática****Cap. 2****Pág. 2**

1



00'10"

Introdução

Neste material, você vai aprender mais sobre as configurações de API, mas há alguns pré-requisitos que você precisa atender primeiro: o primeiro deles é ter consumido o material anterior (4.1) e o segundo é ter consumido material em que explico o que é API e para que serve (4.0).

Para continuar aprendendo tudo na prática, prossiga com a leitura!

2



00'47"

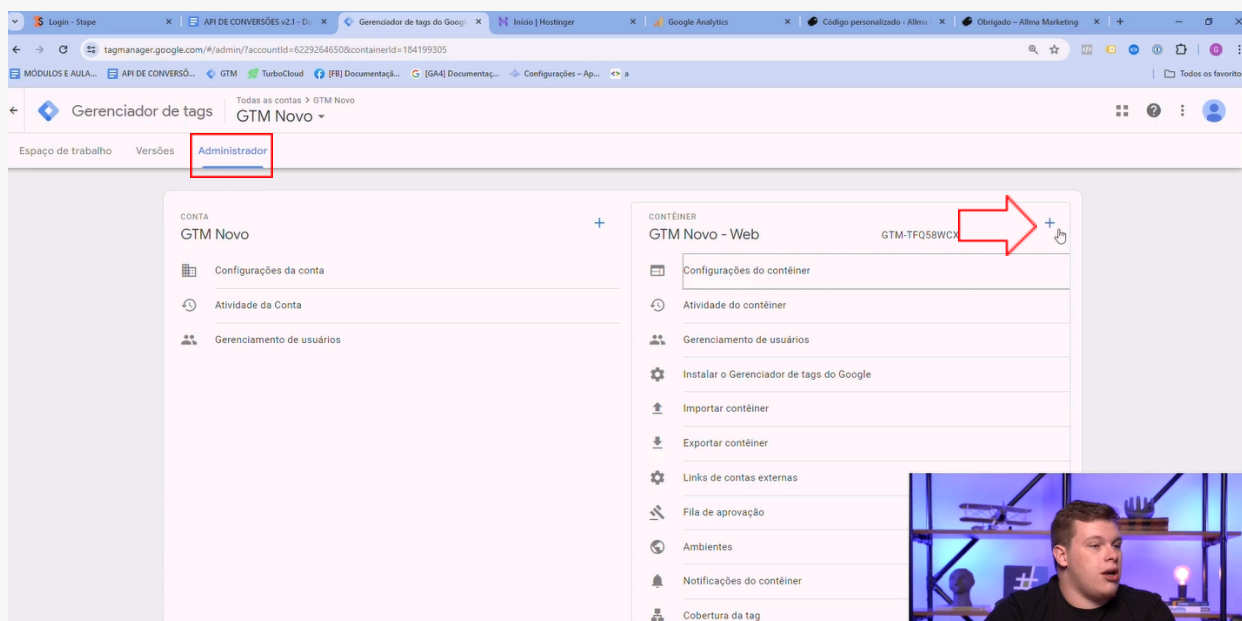
API na prática

Tudo o que a gente tem em nosso pixel Web, a gente tem que ter espelhado em nossa tag, em nossos eventos do GA4, do Meta API. Então, se eu tenho Lead no Meta Web, eu tenho que ter Lead no Meta API. É importante você saber disso porque senão os eventos não serão deduplicados.

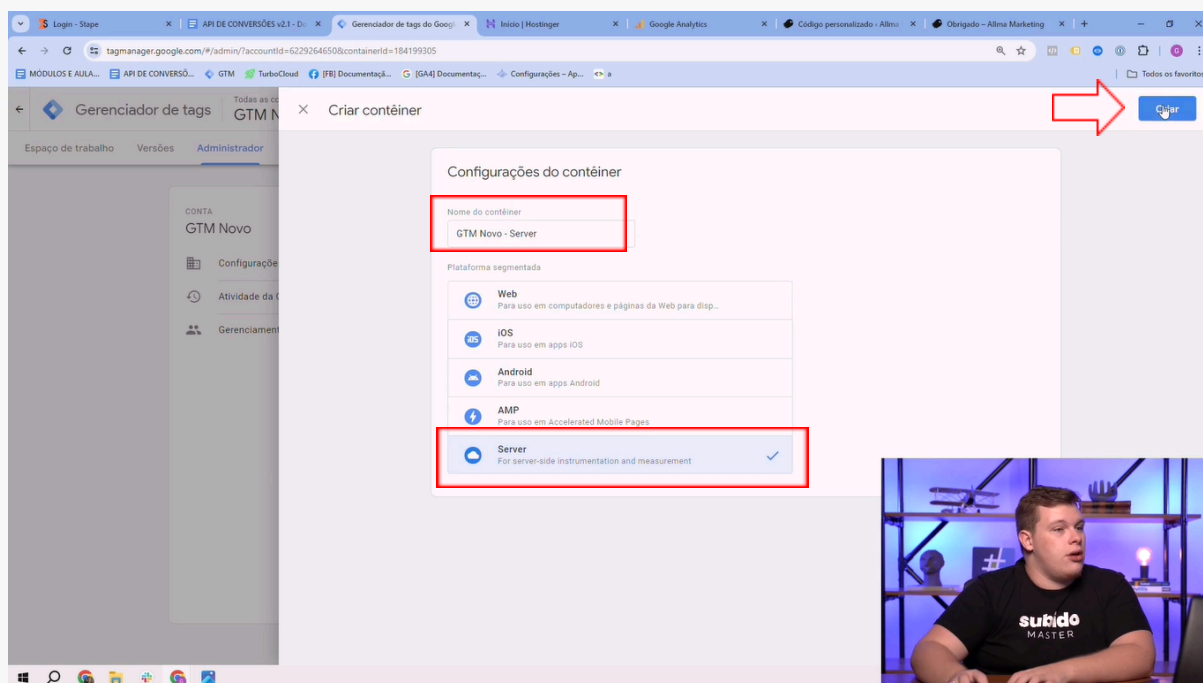
O próximo passo, então, é criar o **container Server**. Acompanhe o passo a passo a seguir.

----- PASSO A PASSO -----

01. Vá em “Administrador” para clicar no “+” para criar novo container.

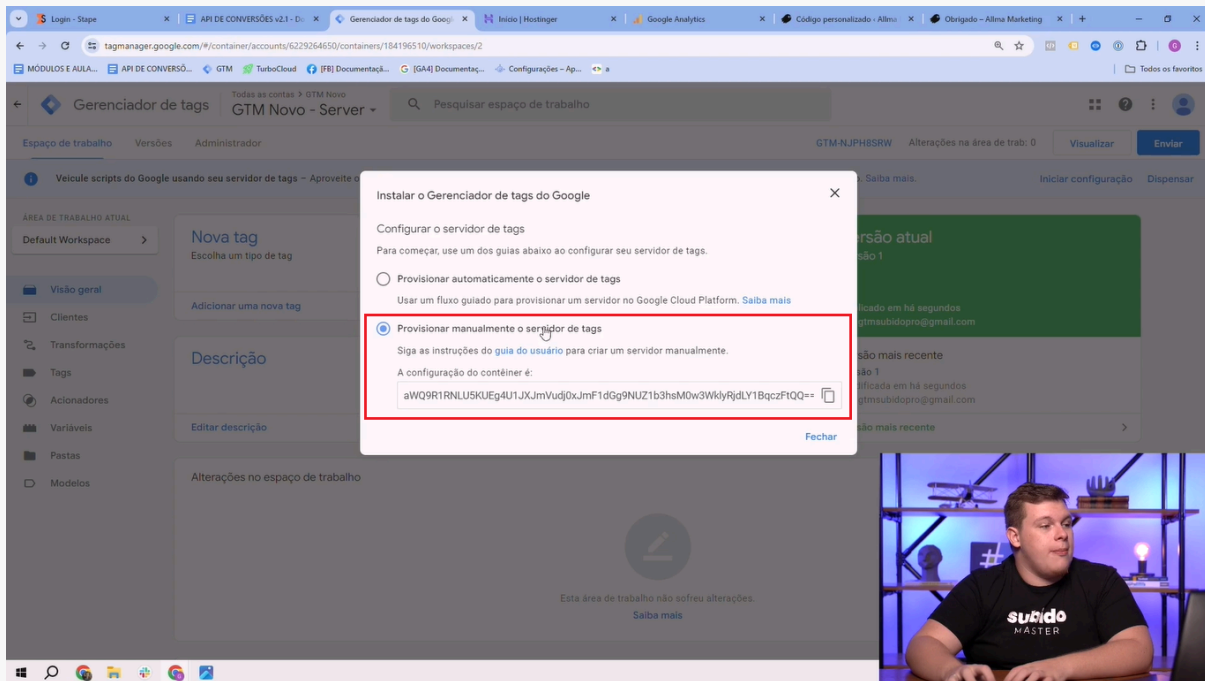


02. Depois, clique em “Server”, nomeie e clique em “Criar”.



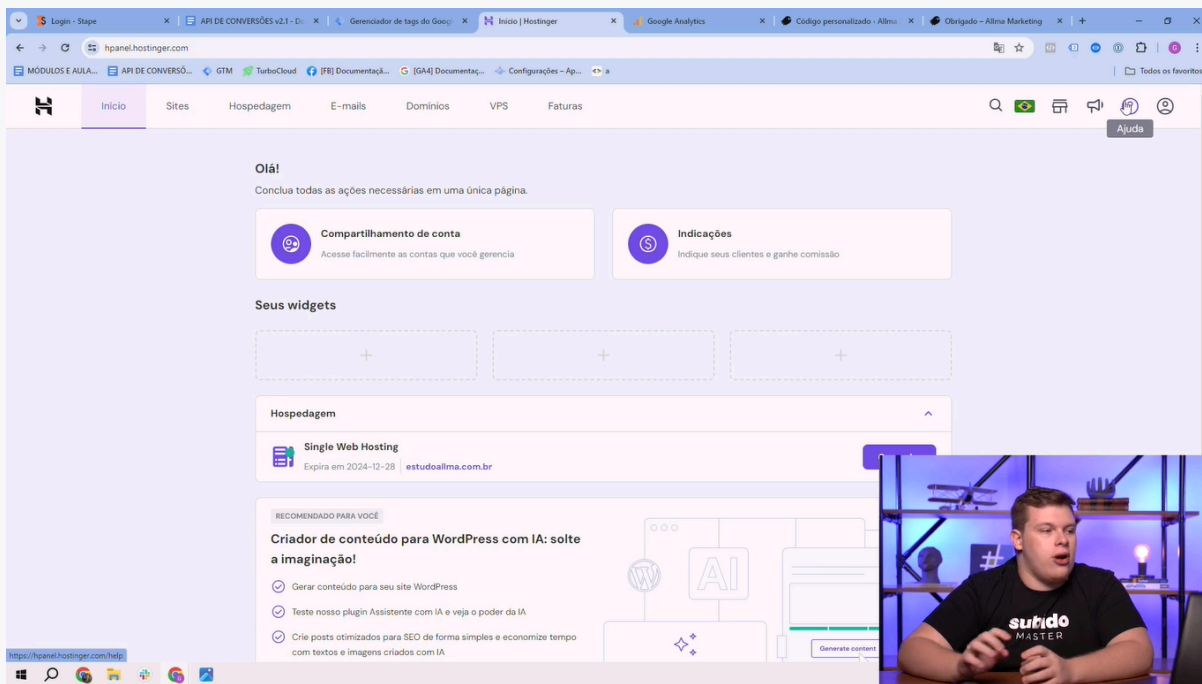
03. Com o container criado, aqui teremos uma pequena diferença. A gente não vai mais provisionar automaticamente. **A gente vai**

provisionar manualmente. Vamos copiar a configuração do container e fechar.

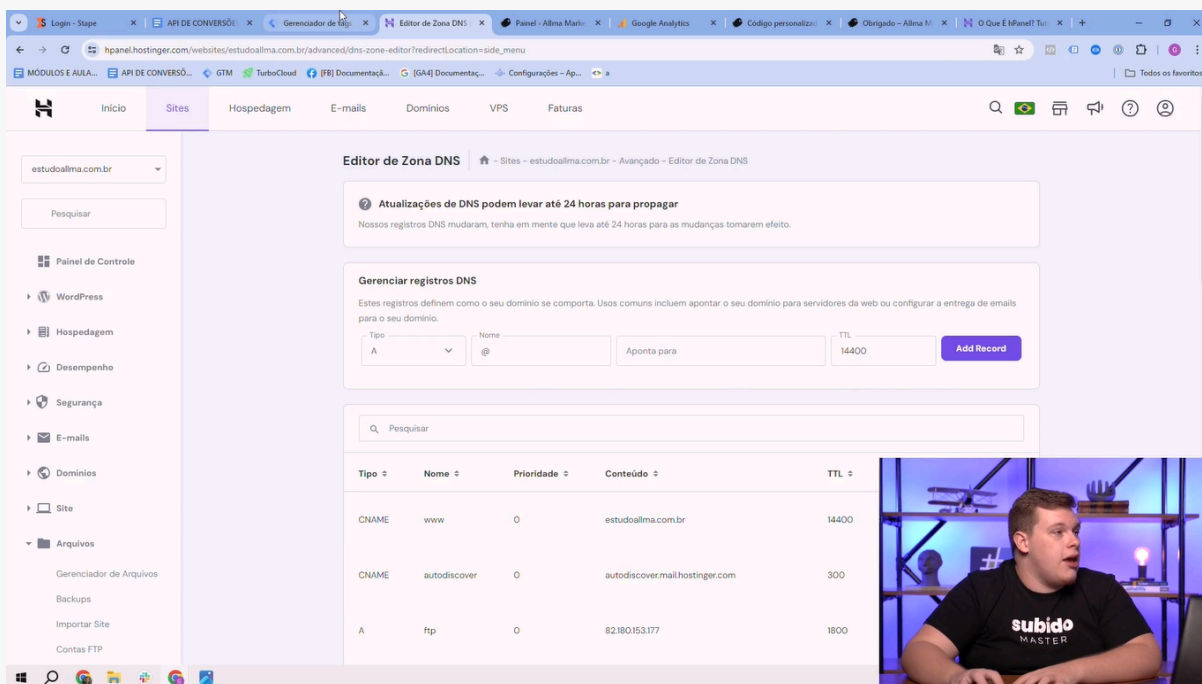


Agora, a gente tem que ir para a nossa hospedagem, **onde o nosso domínio está hospedado**. Eu hospedei na Hostinger, mas o seu pode estar em algum outro.

04. Acessando o site da hospedagem, eu fiz o login com a minha conta e não tem segredo algum quanto a isso. Você fará o mesmo em qualquer hospedagem que seja sua.



05. O que vai mudar um pouco é como é por dentro da hospedagem, mas, de qualquer forma, **o que a gente precisa fazer é ir até o gerenciador e procurar por DNS zone editor**. Depois, vamos deixar essa página aberta.



Se você usa o CDN (o mais famoso é o Cloudflare), você vai ter um processo um pouco diferente do que você vai ver a seguir.

“Edu, eu tenho uma CDN e não é Cloudflare.”

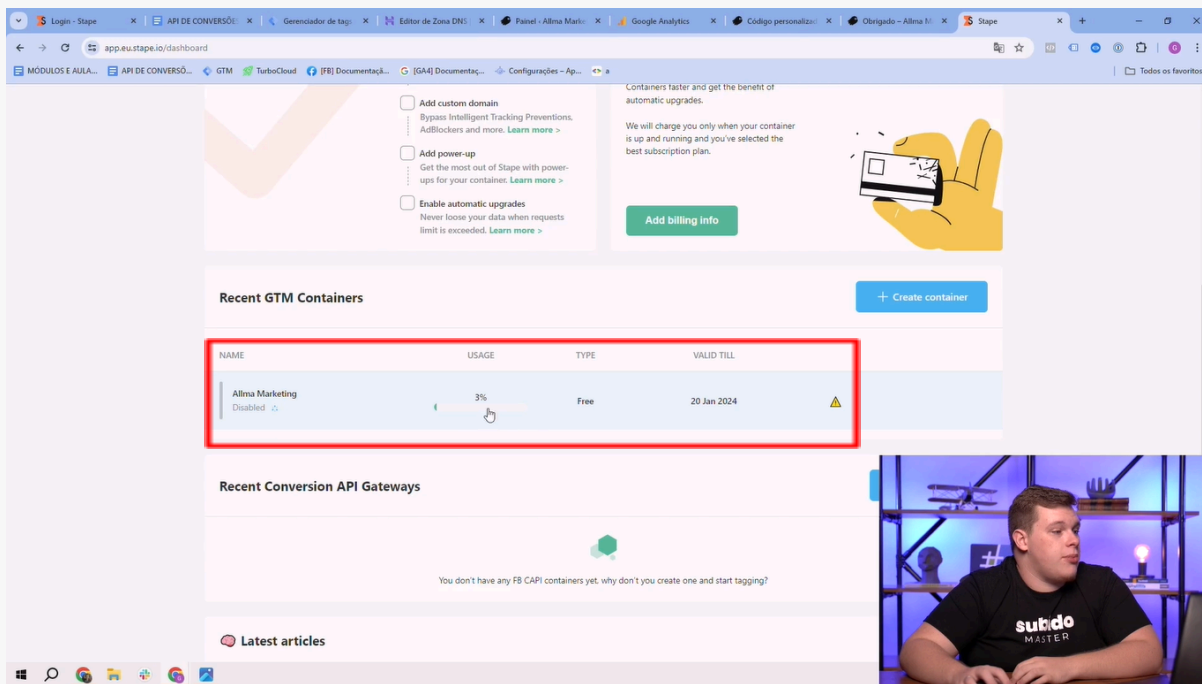
Nesse caso, você terá que entrar em contato com o suporte do seu CDN, explicar que você quer instalar um API de conversões do Facebook e que precisa de uma conexão com a Stape, bem como que ele instale os registros.

Na Cloudflare, como ela é um pouco maior, permite que você mesmo faça isso.

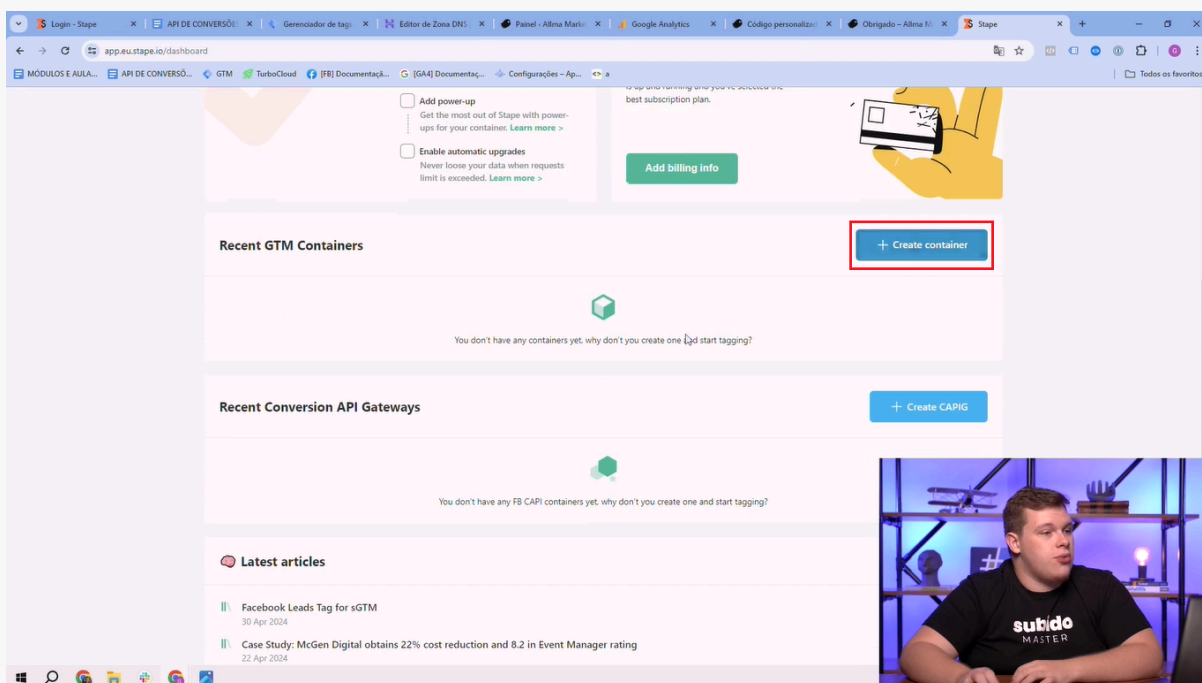
Em seguida, vamos para o Stape iO, que é a nossa ferramenta que será utilizada para fazer a nossa API de conversões.

Para entrar no Stape iO, a gente vai pesquisar por, literalmente, isso no Google, para fazer o seu login.

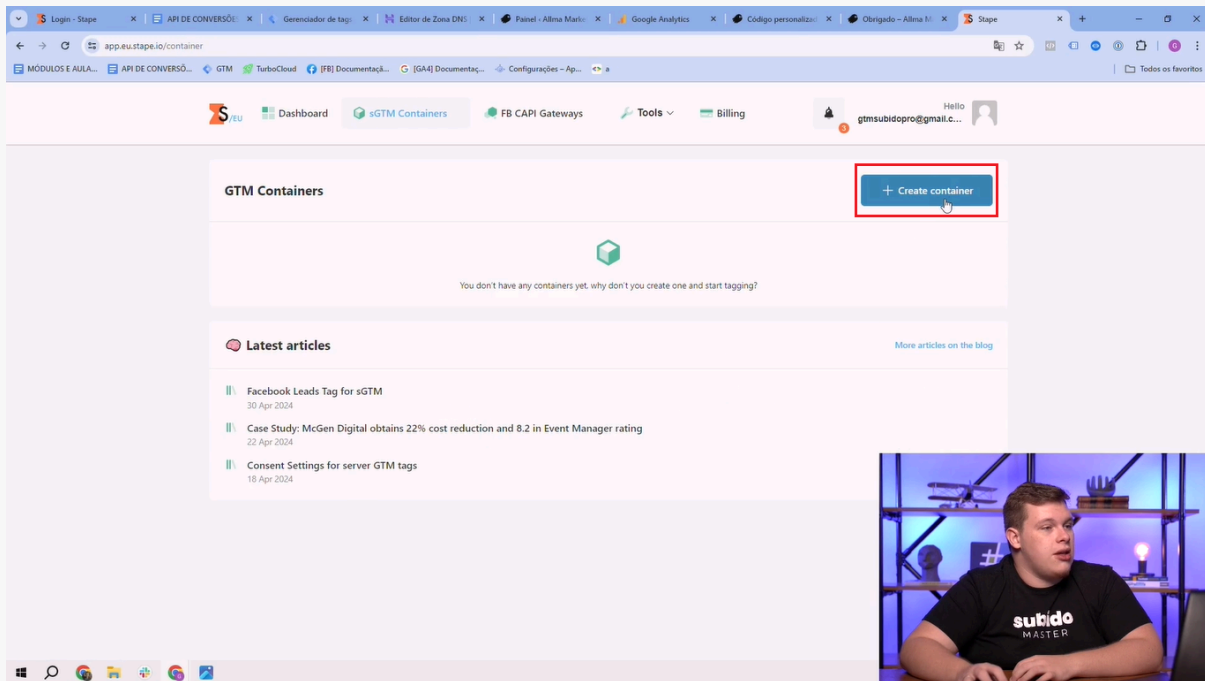
06. Ele vai abrir já em um dos meus containers quando eu clicar em **“Dashboard”**.



07. Como eu quero criar um novo, esse será excluído para que o processo seja refeito. Eu tenho duas opções. Uma delas é clicar em **“Create container”**, em **“Dashboard”**:

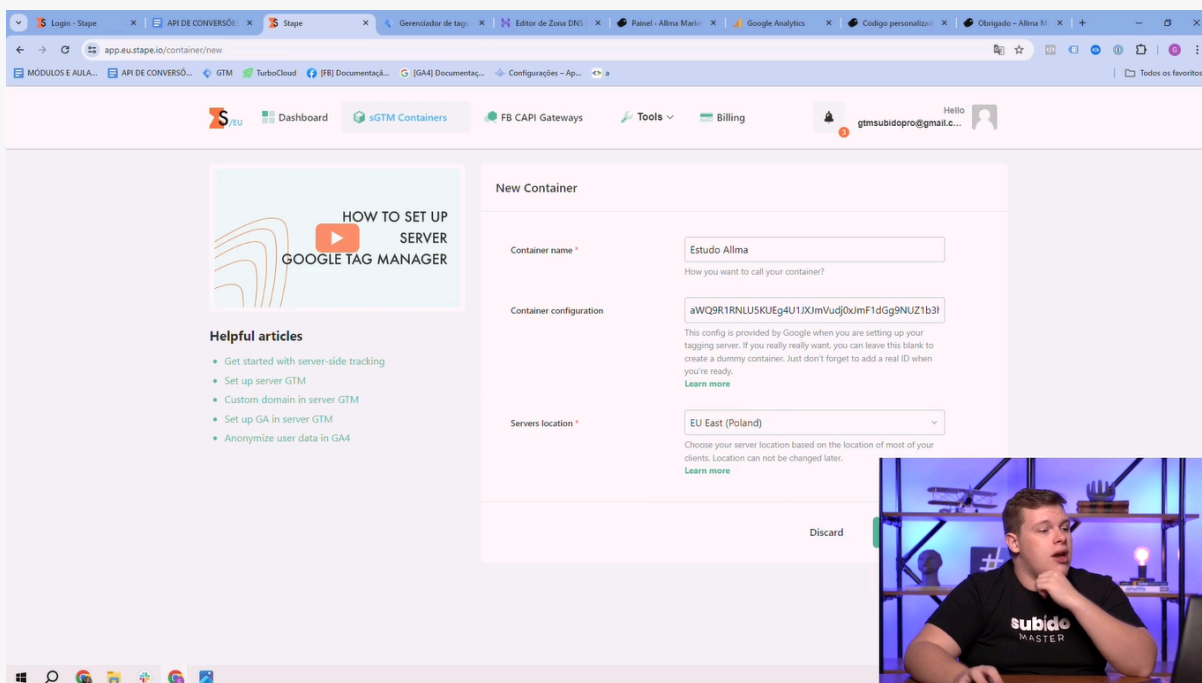
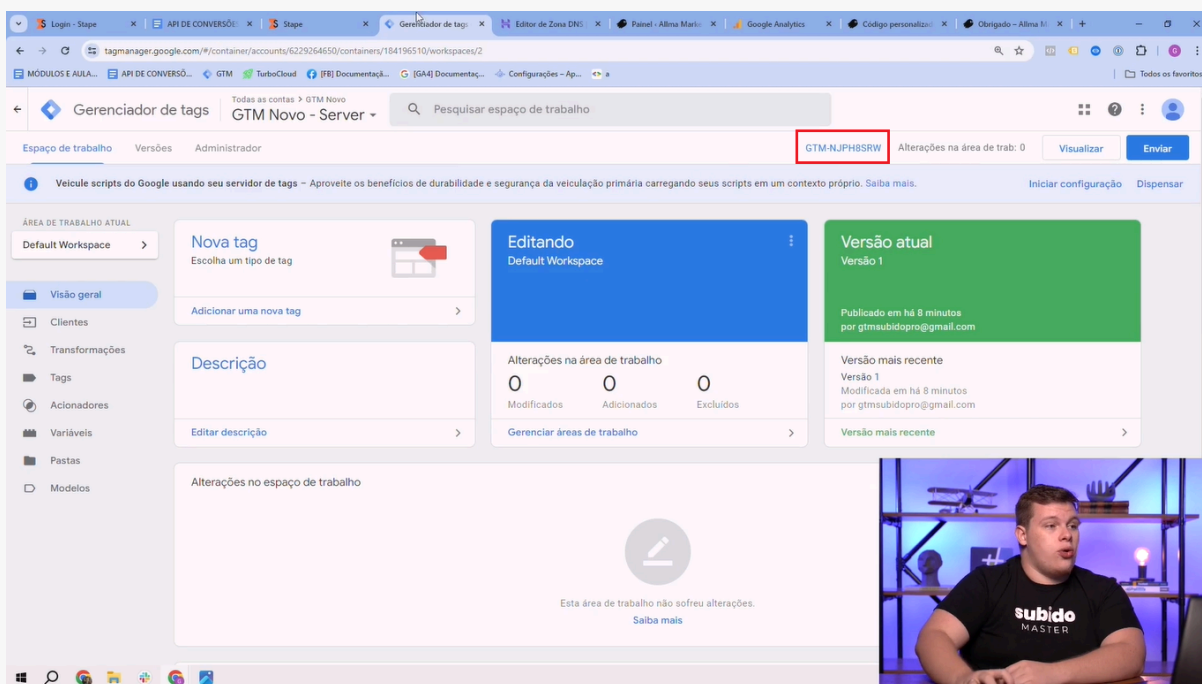


08. E a outra delas é ir em **“sGTM Containers”** e clicar em **“Create container”**:



Ao clicar em **“Create container”**, a gente vai, primeiro, dar um nome ao container e, depois, preencher o campo **“Container configuration”**. O preenchimento desse campo deve ser feito com a configuração que foi copiada quando selecionamos a opção de provisionar manualmente.

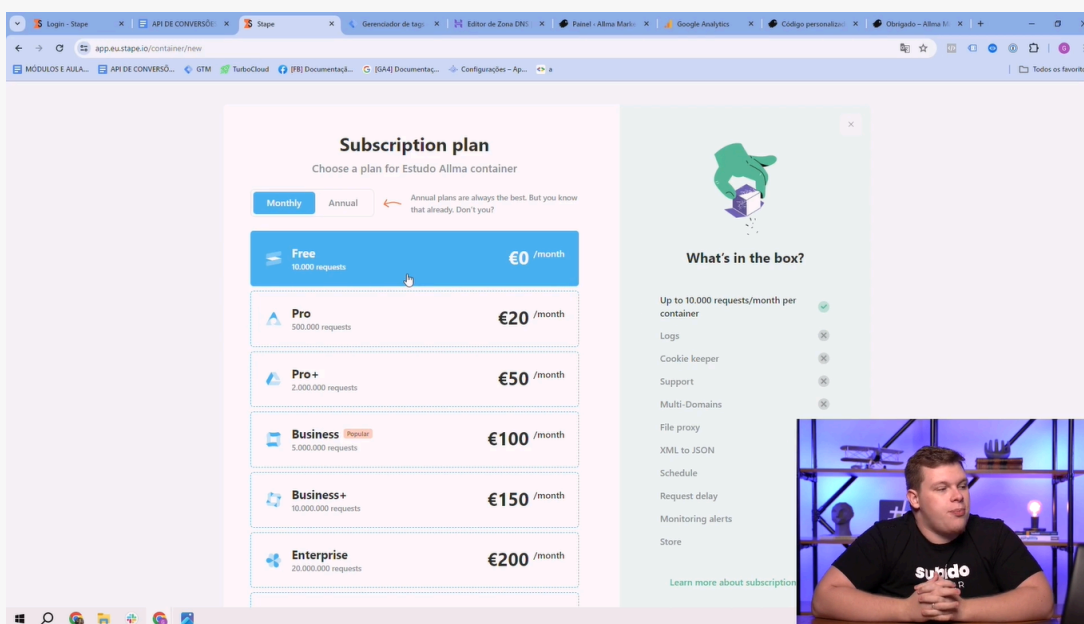
09. Se você não tiver salvado, você volta ao GTM e clica no nome do GTM. Aquele pop-up vai reaparecer e você pode copiar a configuração novamente.



É PAGO OU GRATUITO?

Uma dúvida comum é se a API é paga ou não. **A resposta é que pode ser, mas também pode não ser.**

Se você está fazendo uma API muito simples para um negócio local ou para um lançamento menor, muito provavelmente você não vai precisar pagar.



O plano gratuito possibilita até 10 mil requests por mês. Esses requests, na maioria das vezes, são eventos que serão disparados.

Então, se você está disparando evento de Page View, de Lead e de Contact, o Lead que entrar na tua página vai disparar o Page View, depois vai disparar novamente o Page View (porque abriu outra aba) e o Contact, ou seja, você teve quatro requests.

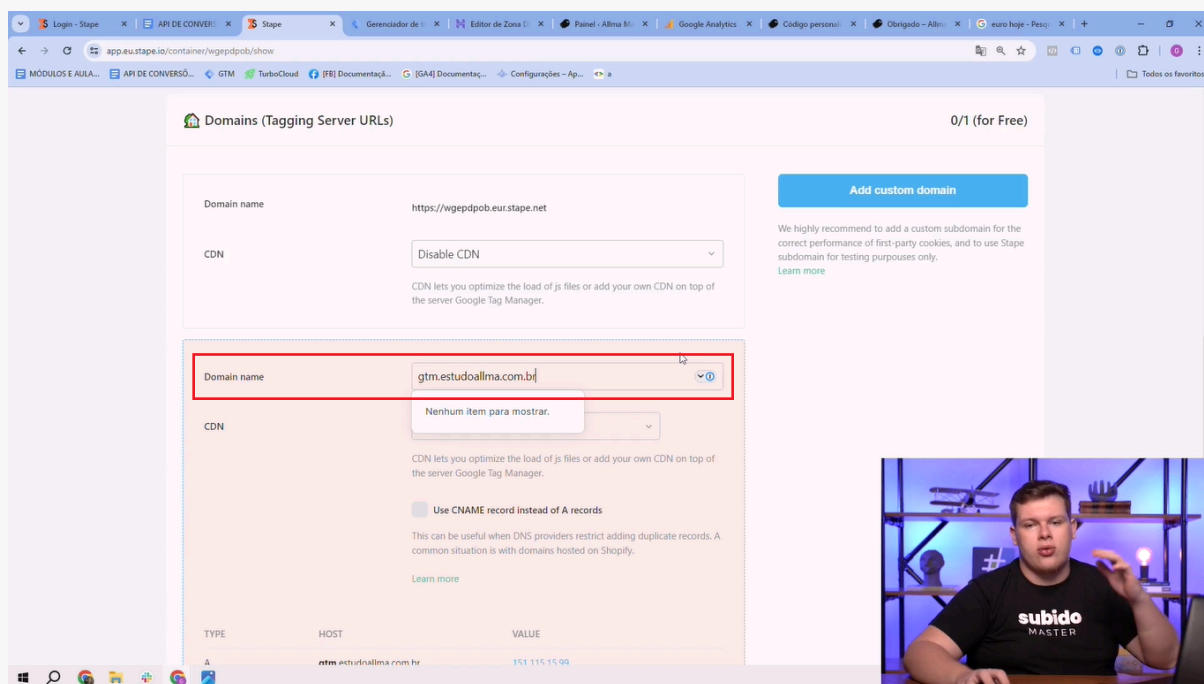
Alguns vão disparar só Page View, alguns vão disparar só Lead e assim por diante.

O ponto é que **se você estiver com o *Advanced Matching* disparando alguns outros parâmetros, vai aumentando o nível de requisições.** Então, não dá para ter certeza se você teve 10 mil disparos no pixel por mês e se você vai precisar ou não pagar.

Porém, caso você ultrapasse os 10 mil requests, você recebe um e-mail avisando que o Stape será desativado e eles te dão a opção de pagar os 20 euros por mês. Mas você também tem a opção de esperar pelo próximo mês para reativar o plano.

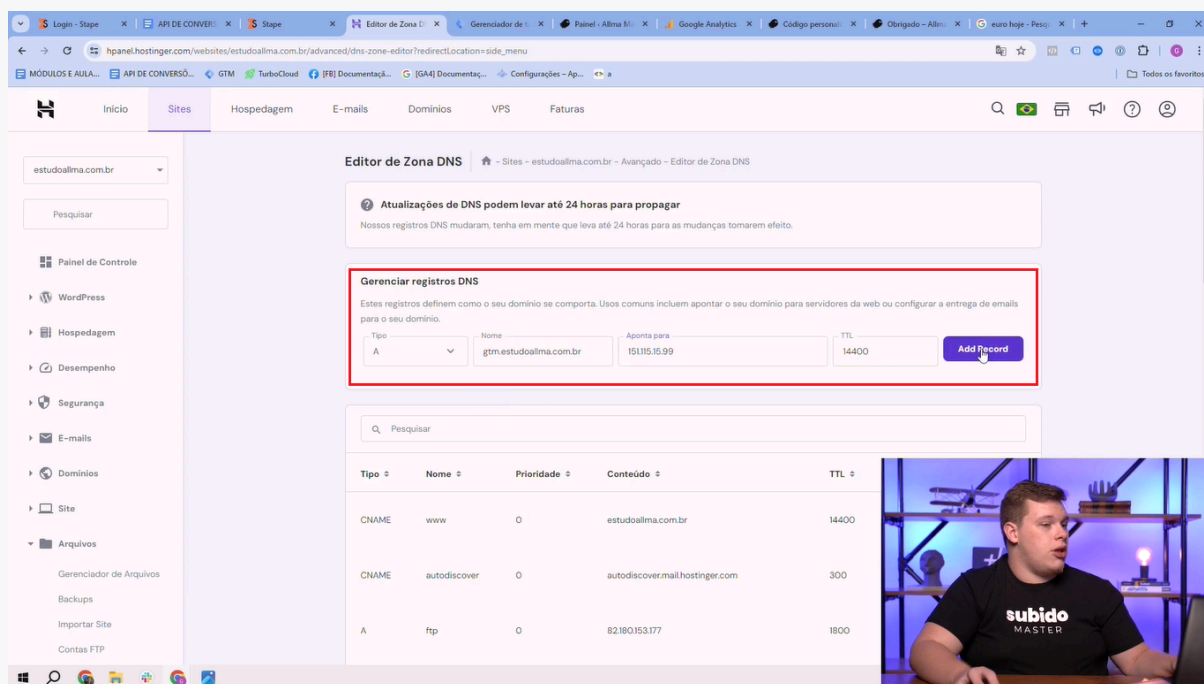
11. O próximo passo é inserir o *Add custom domain*. **A gente tem que criar um subdomínio personalizado para que os dados do nosso lead/cliente passem por esse domínio e, depois, cheguem ao Stape seguros.**

Para isso, a gente vai clicar no Stape e descer até chegar em *Domains (Tagging Server URLs)* e clicar em **“Add custom domain”** para inserir o **“gtm.[o seu domínio]”**.



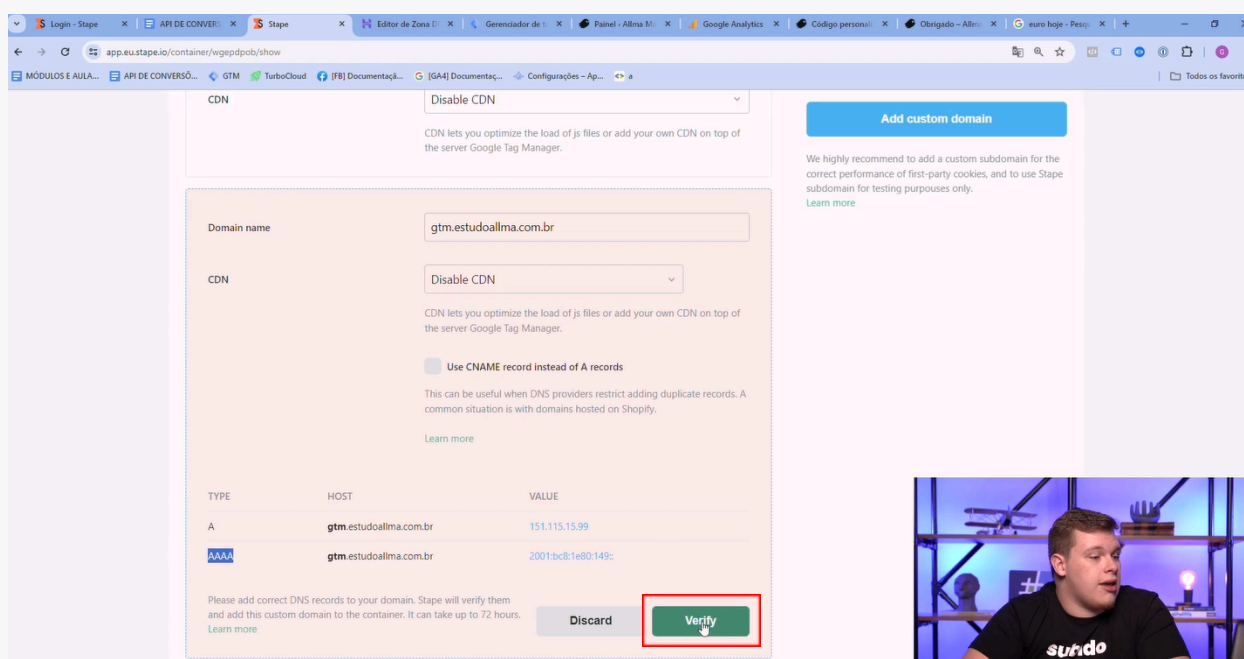
Eu indico que você insira o **“gtm”** por uma questão de padronização.

- 12.** Em seguida, ele vai pedir para a gente instalar dois tipos de registro, o **tipo A** e o **tipo AAAA**. E o que a gente tem que fazer é copiar o tipo A e o valor, voltar ao Hpanel, colá-los nos respectivos espaços indicados e adicionar o registro. Depois, a gente confirma e repete o processo com o tipo AAAA.



Para confirmar se está tudo certo, a gente desce a tela para ver se está aparecendo.

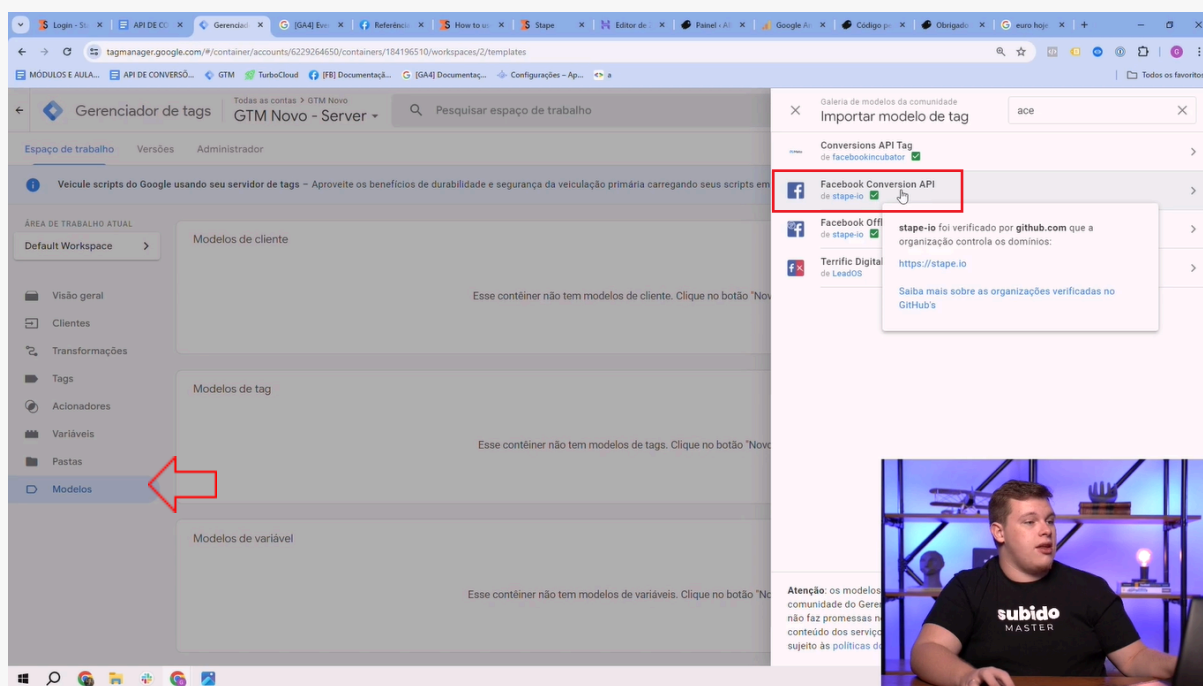
13. Depois, voltamos ao Stape para clicar em **“Verify”**. Na maioria das vezes, isso não será verificado na hora porque ele precisa passar por vários DNSs. Então, pode demorar até 72 horas.



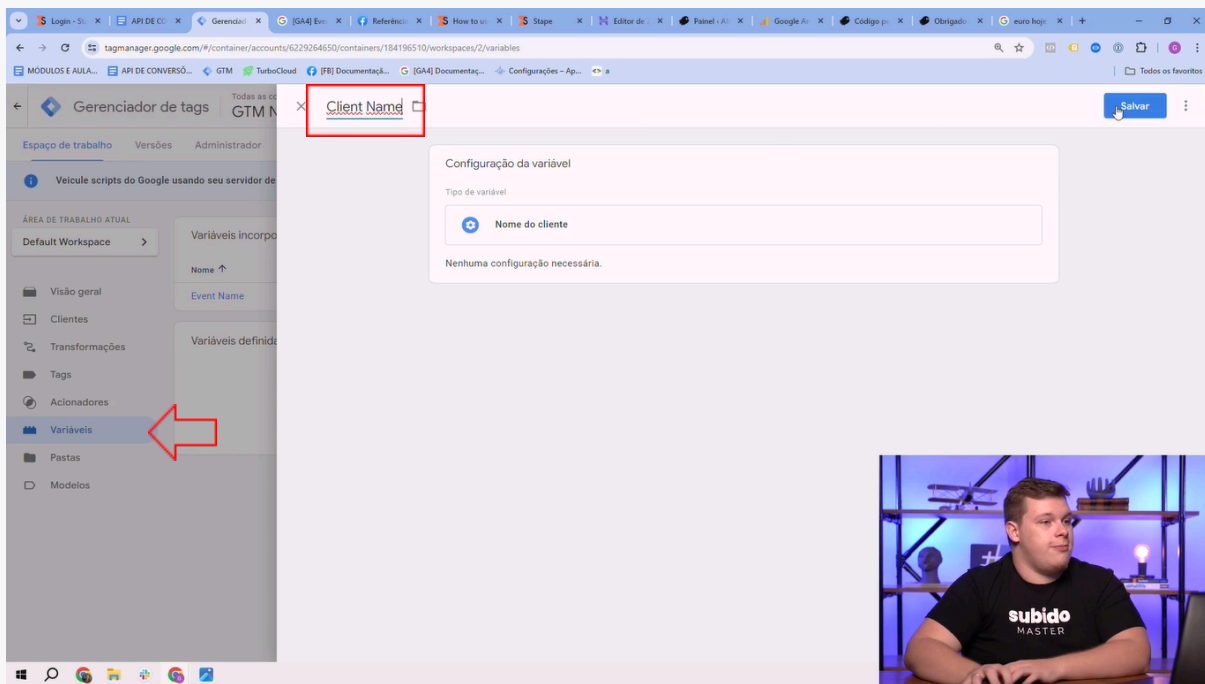
Se você estiver usando o CDN, clique [neste link](#) e abra este post que ensina exatamente o passo a passo de como deve ser feito na Cloudflare.

Agora, a gente vai abrir as nossas referências de evento para caso seja necessário usá-las.

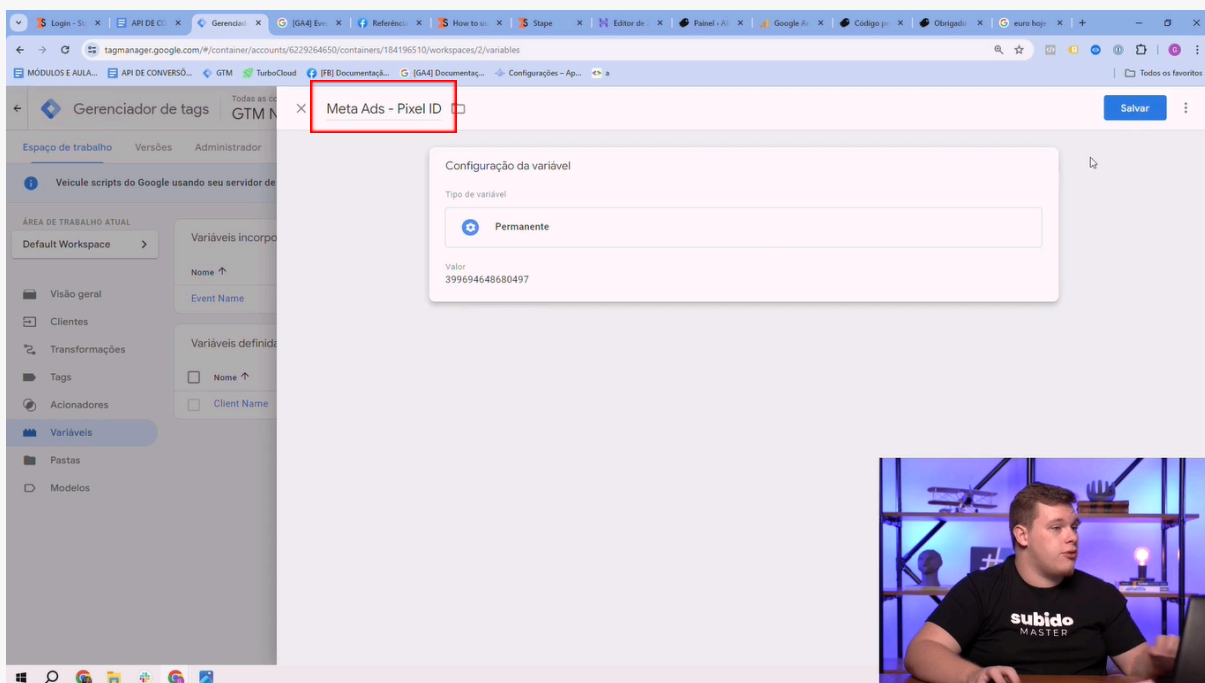
- 14.** O próximo passo é o Container Server. Nele, a gente vai criar um modelo de tag que se chama *Meta Conversions API* (por *stape-io*). Então, voltamos ao nosso container server, dentro do GTM, em **“Modelos”**, para pesquisar na galeria um modelo de tag. Pesquisamos por **“Facebook”** para encontrar o **“Facebook Conversions API”** e adicionar ao espaço de trabalho.



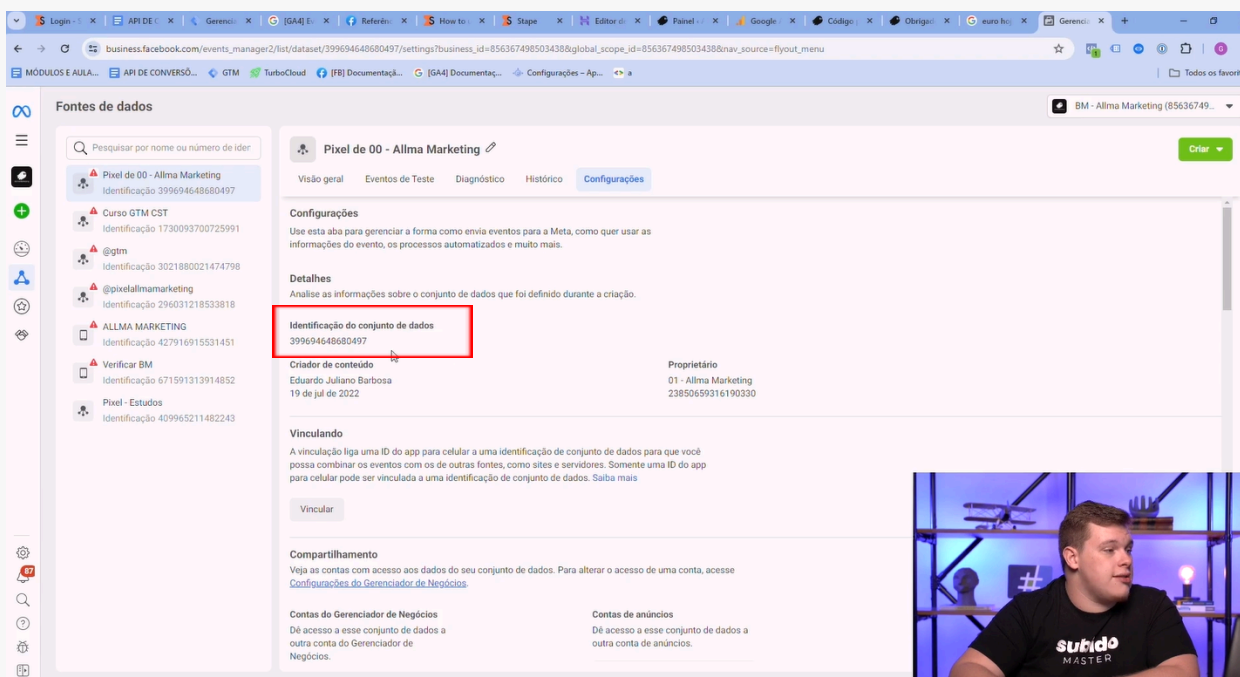
- 15.** Depois, vamos em **“Variáveis”** para criar uma nova variável chamada **“Client Name”**. A **“Event Name”** já estava incorporada.



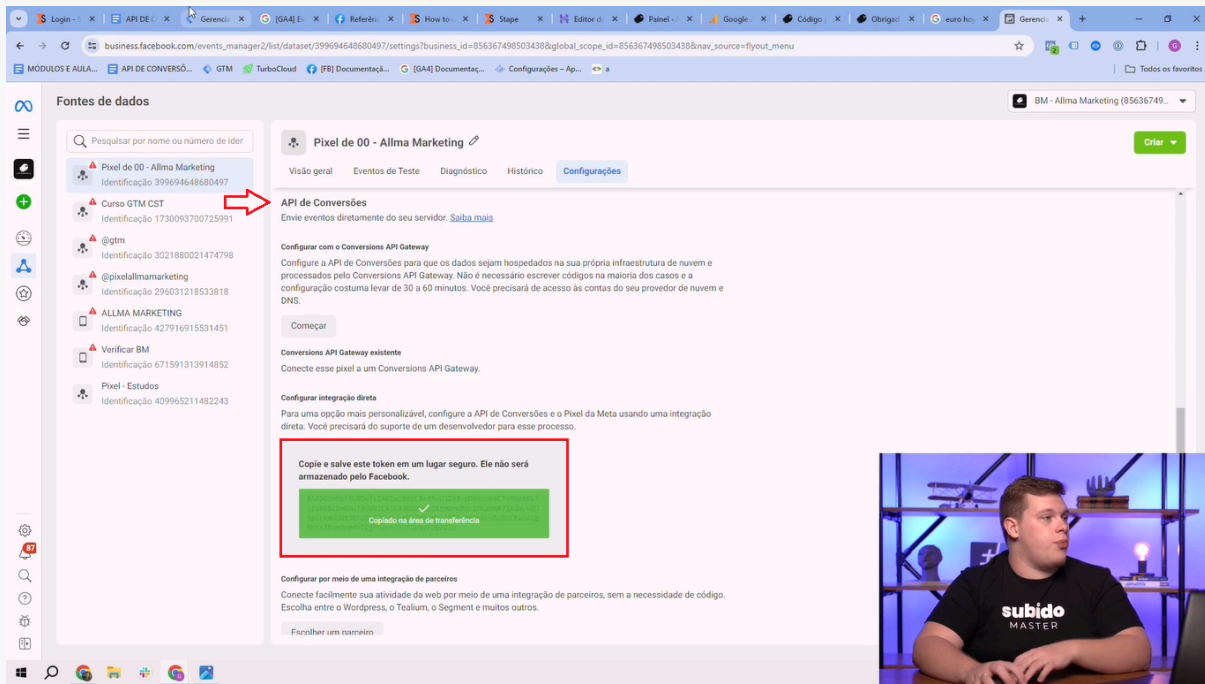
16. Agora, temos que criar o **Meta API - Pixel ID**, ele é o ID do pixel mesmo.



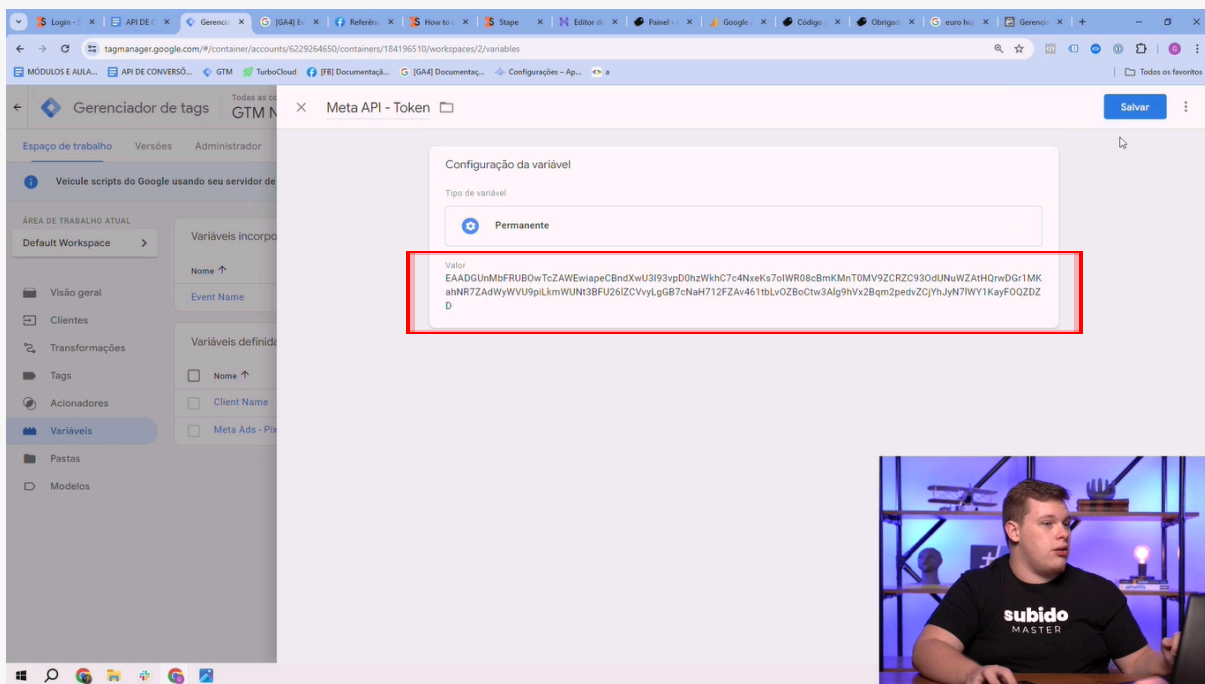
17. Por fim, precisamos pegar o *Meta API - Token*. Para isso, a gente vai nas configurações do gerenciador de eventos para pegar o código que deve ser inserido em **“Valor”**.



18. Ao descer a página, em *API de Conversões*, vamos encontrar o **“Gerar token de acesso”**, clicamos nele para ele gerar o link e, em seguida, copiamos.



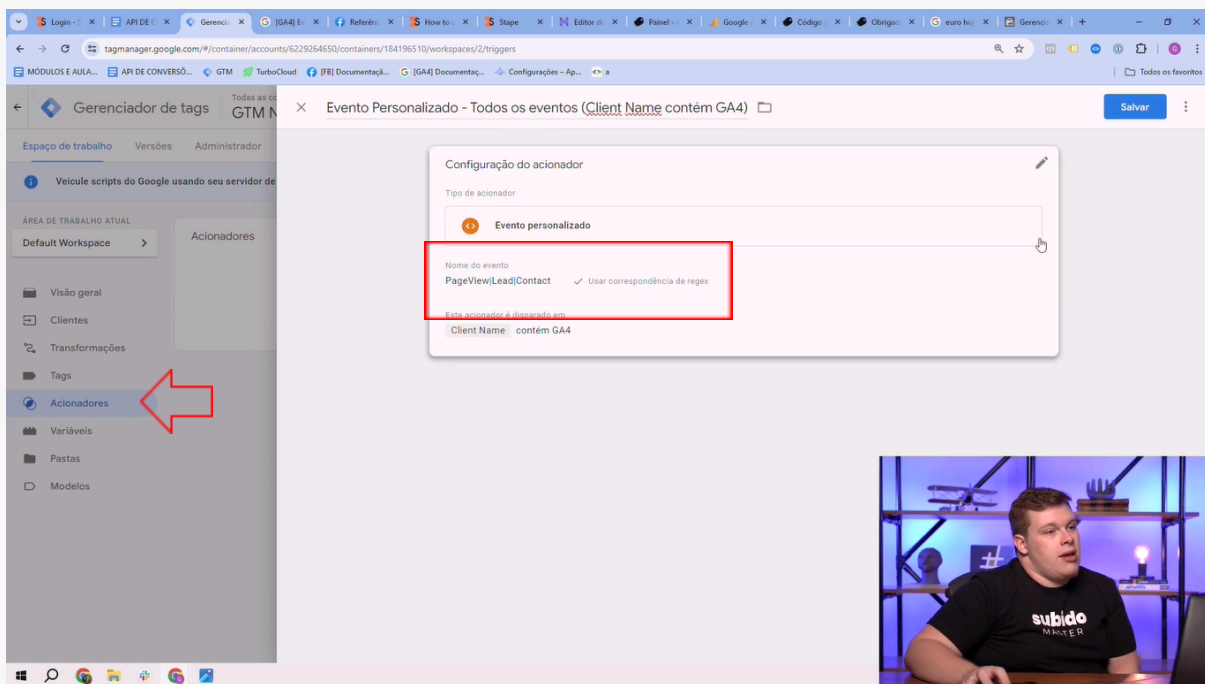
19. Depois, é só colar o token de acesso em **“Valor”** para salvar a variável.



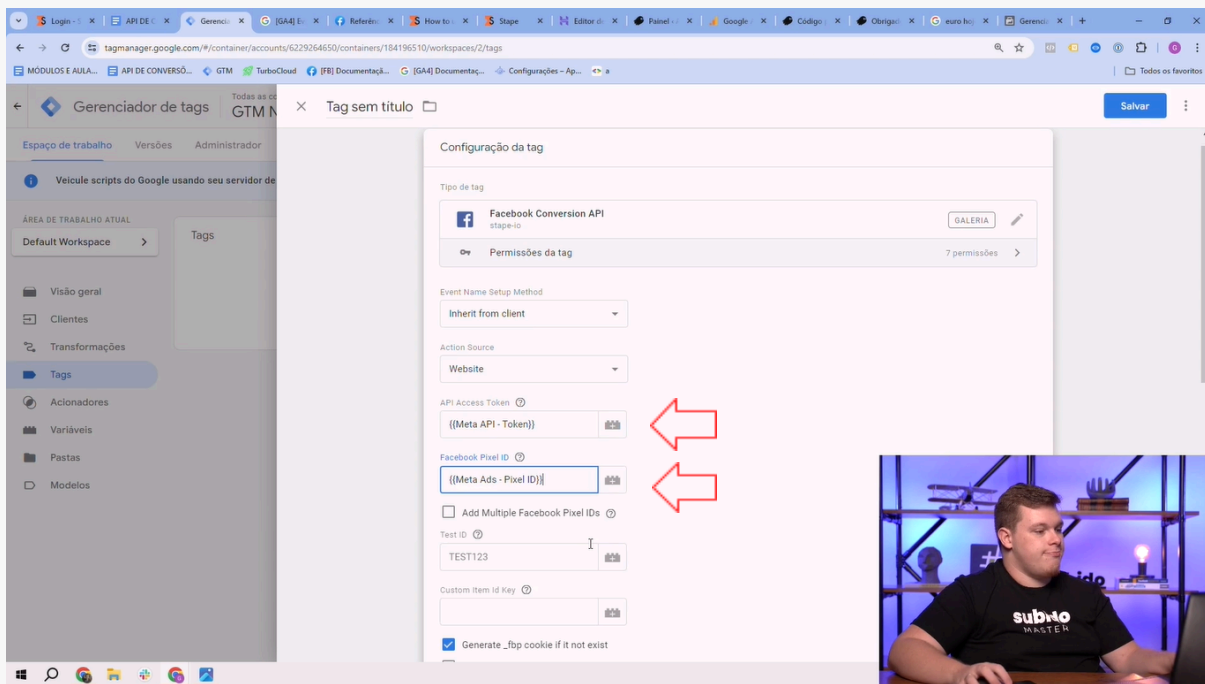
Com essa parte das variáveis finalizadas, vamos aos acionadores para criar um novo acionador de evento personalizado onde, em todos os eventos, o **Client Name contém GA4**.

20. Em “**Nome do evento**”, precisamos colocar todos os nomes dos eventos que a gente vai querer disparar no Facebook. Os eventos que a gente quer disparar são: Page View, Lead e Contact. Inserimos o nome da seguinte forma: “**PageView|Lead|Contact**”.

É obrigatório que a gente use a correspondência regex. Sem isso o seu API não vai funcionar.



21. Depois disso, vamos para a nossa **tag**. Ela deve se chamar **Meta API tag** e é um modelo que a gente baixou anteriormente. Tudo o que a gente tem que fazer na configuração dessa tag é inserir o **API access token** e o **Facebook Pixel ID**.



Em seguida, selecionamos o acionador de disparo, que foi o que criamos.



“Edu, fiz a API e disparei só Lead e Contact, mas depois eu instalei o evento de Search.”

Depois de configurar o evento de Search em seu Container Web e disparar o evento do API, você tem que ir até o acionador e adicionar **“|Search”** ao nome do evento, caso contrário não vai funcionar.

Ele só dispara o Container Server com base nesses acionadores.

Agora, a gente vai adicionar o que chamamos de **Test Event Code**.

22. Quando a gente vai em nosso gerenciador de eventos, clica em **“Eventos de teste”** e clica em **“Confirme se os eventos do seu servidor estão configurados corretamente”**, ele traz essa visão do **Test Event Code**.

 Pixel de 00 - Allma Marketing 

[Visão geral](#) [Eventos de Teste](#) [Diagnóstico](#) [Histórico](#) [Configurações](#)

Testar seus eventos

Verifique se os seus eventos estão configurados corretamente. O objetivo dos eventos de teste é ajudar você a confirmar se os seus eventos do site, app e servidor estão sendo recebidos conforme o esperado. Não há qualquer garantia de que o comportamento observado na página do evento de teste do Gerenciador de Eventos será igual em outros ambientes da Meta.

Nenhuma restrição no processamento de dados resultante das suas preferências de anúncios e controles de dados e status de consentimento de terceiros se aplicará aos seus eventos de teste.

[Saiba mais](#)

Confirme se os eventos do seu servidor estão configurados corretamente

Você pode realizar um teste para verificar se o servidor está se conectando e compartilhando dados de eventos. Certifique-se de que a página Testar eventos permaneça aberta no Gerenciador de Eventos durante o teste.

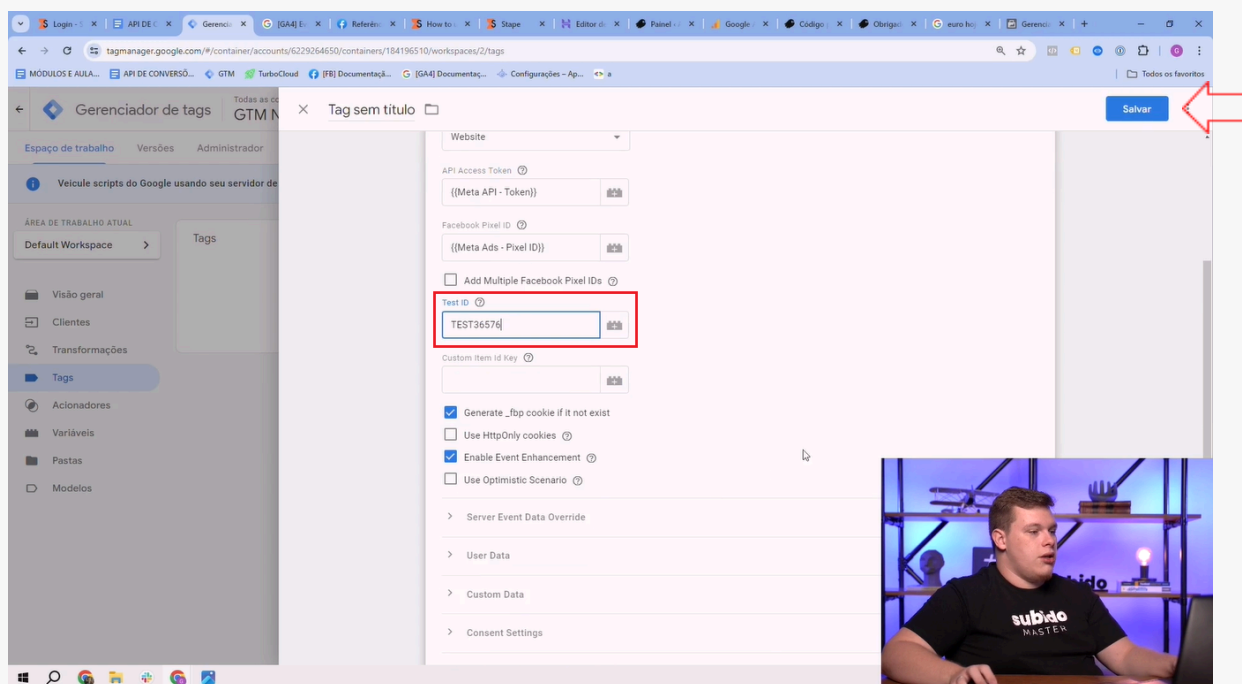
Para executar o teste:

1. Abra o Explorador da Graph API e certifique-se de que o código de teste a seguir corresponda ao que é mostrado na carga.

`test_event_code: TEST36576` [Copiar](#)
2. No Explorador da Graph API, clique em "Enviar". Os eventos recebidos aparecerão nesta página. A fonte da ação enviada da carga do Explorador da Graph API deve corresponder à opção selecionada na etapa 1 para que os eventos sejam exibidos corretamente.

[Explorador da Graph API](#)

23. Copiamos esse teste, colamos em *Test ID* e salvamos.

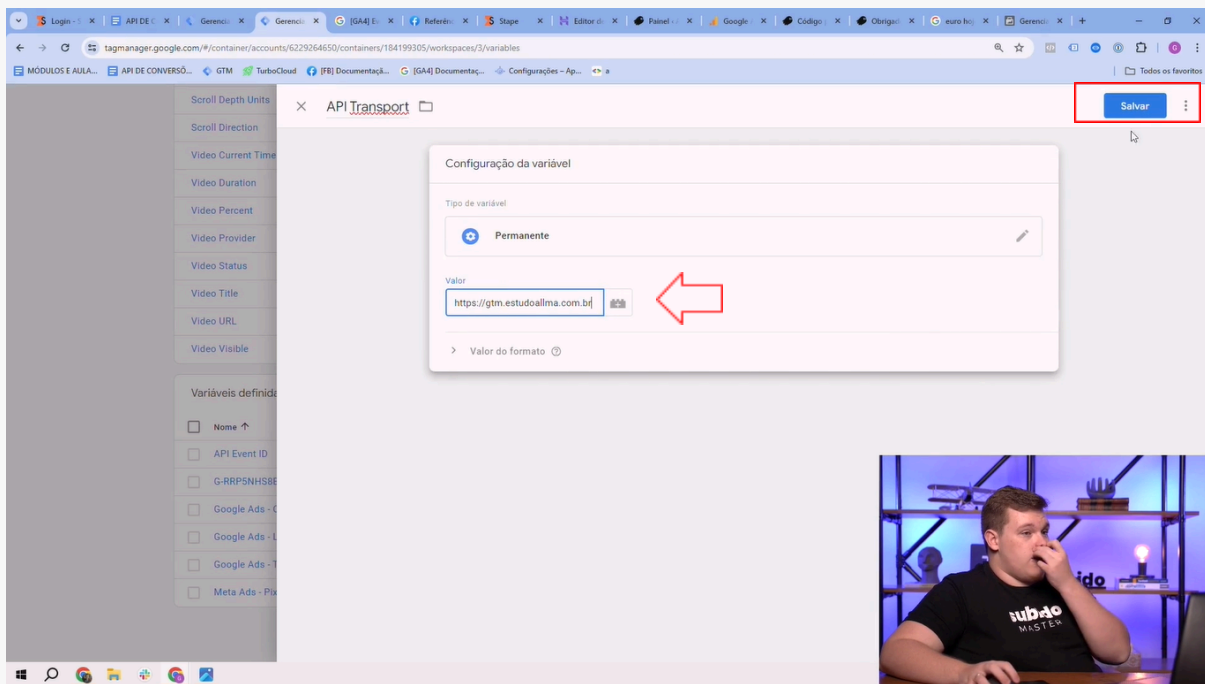


Se a gente coloca o *Test Event Code*, ele vai disparar certinho, mas somente nos debugs. Se você tirar do *Test*, ele não vai disparar nos debugs e vai parecer que não está dando certo, mas vai disparar para o usuário final.

Então, o ideal é que você coloque o Test, faça todos os testes necessários e, depois, tire ele e publique o container para ver se está tudo funcionando corretamente.

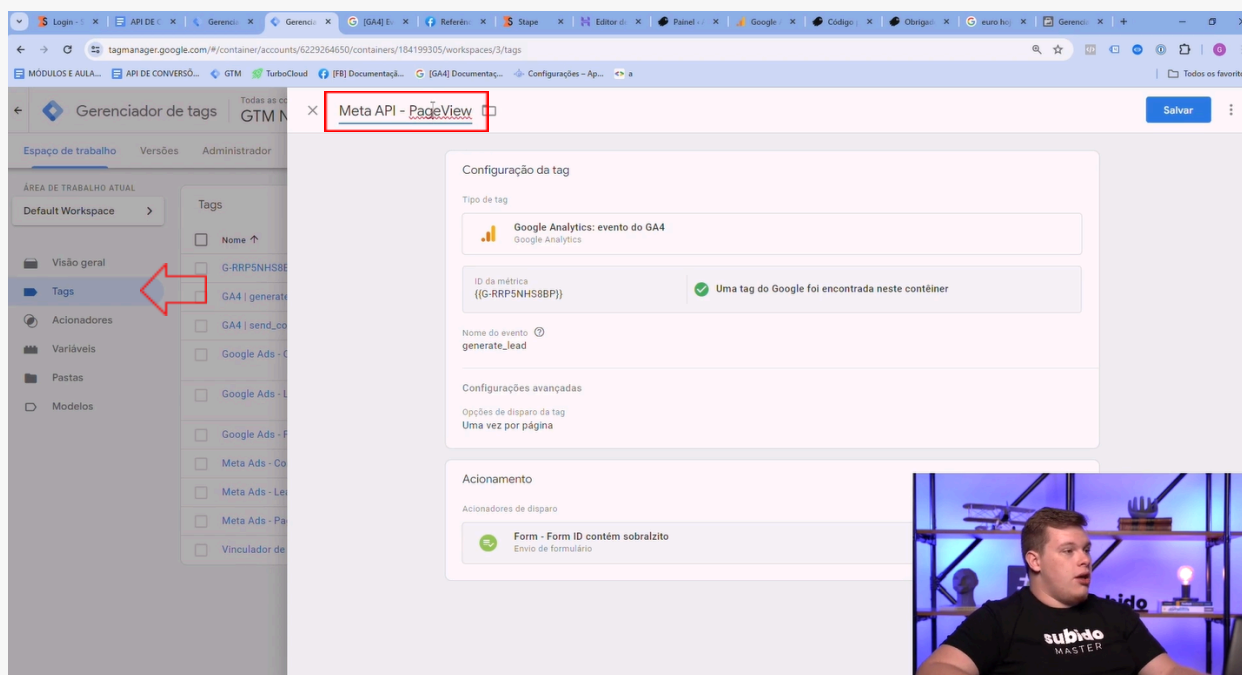
Finalizamos toda a parte do nosso container Server e, agora, é hora de retornar ao container Web para criar uma variável permanente. Essa variável deve se chamar *API Transport*.

- 24.** Em “**Variável definida pelo usuário**”, a gente vai criar uma nova variável. Escolhemos pelo tipo permanente e o valor dela será aquele domínio que usamos no Stape.



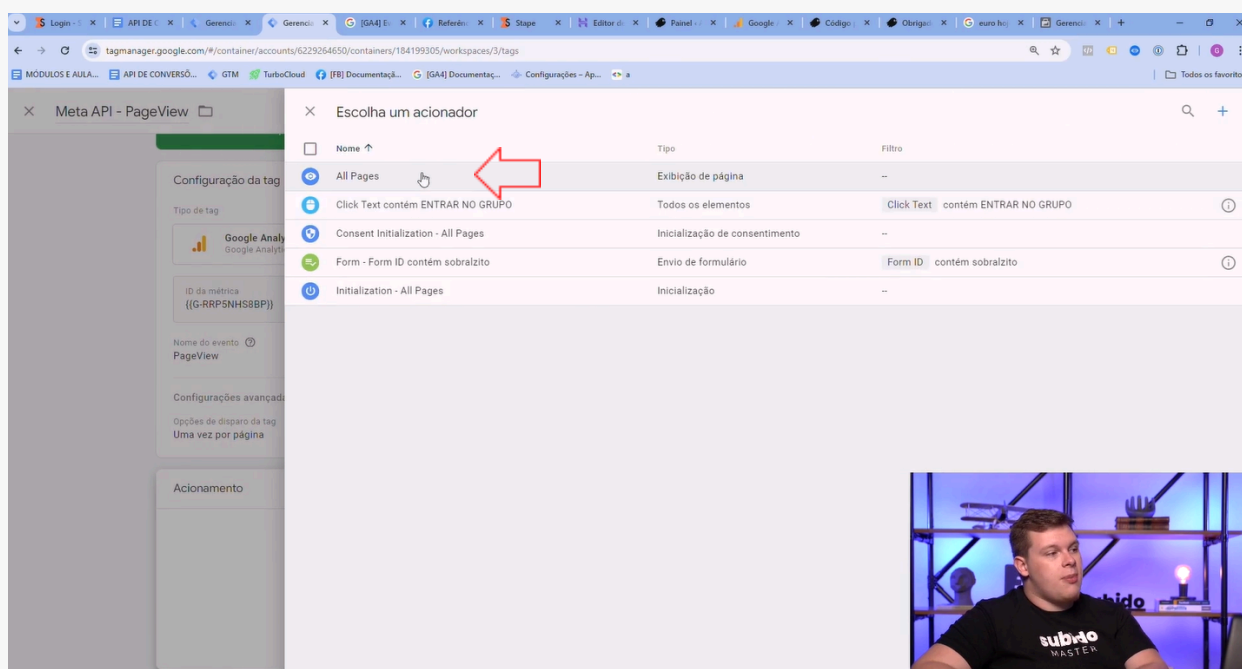
Agora, vamos instalar as tags de Meta API. Até agora, a API está quase toda configurada, mas ainda não criamos as tags que vão disparar o Meta API.

- 25.** Então, a gente tem que entrar em nosso container Web para, em **“Tags”**, criar a nossa tag. Porém, há uma diferença aqui. **A partir de agora, a gente não vai mais nomear como estávamos nomeando. Vamos chamar de Meta API - PageView.**



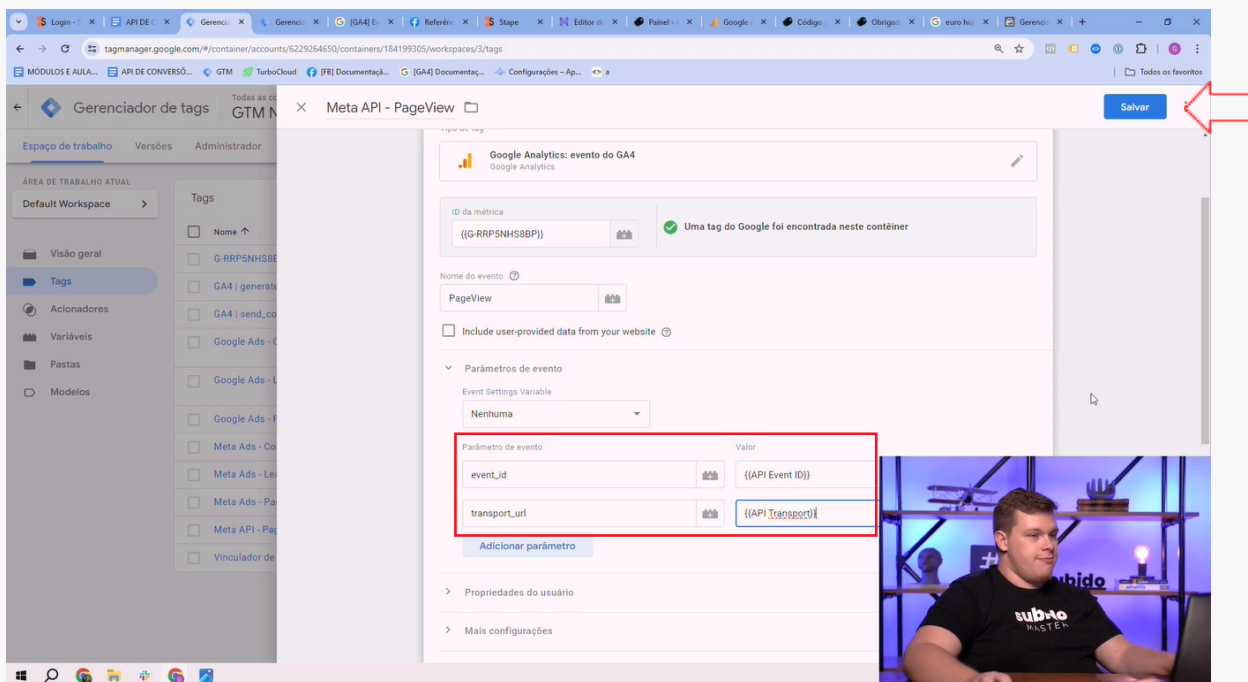
É preciso prestar bastante atenção ao nome do evento, pois ele precisa ser igual ao que está no *Meta Ads - Page View - Web*.

26. Nessa tag, usamos o mesmo ID da métrica, que é o nosso fluxo de dados do GTM, e mudamos o nosso acionador para que ele dispare em **“All pages”**.

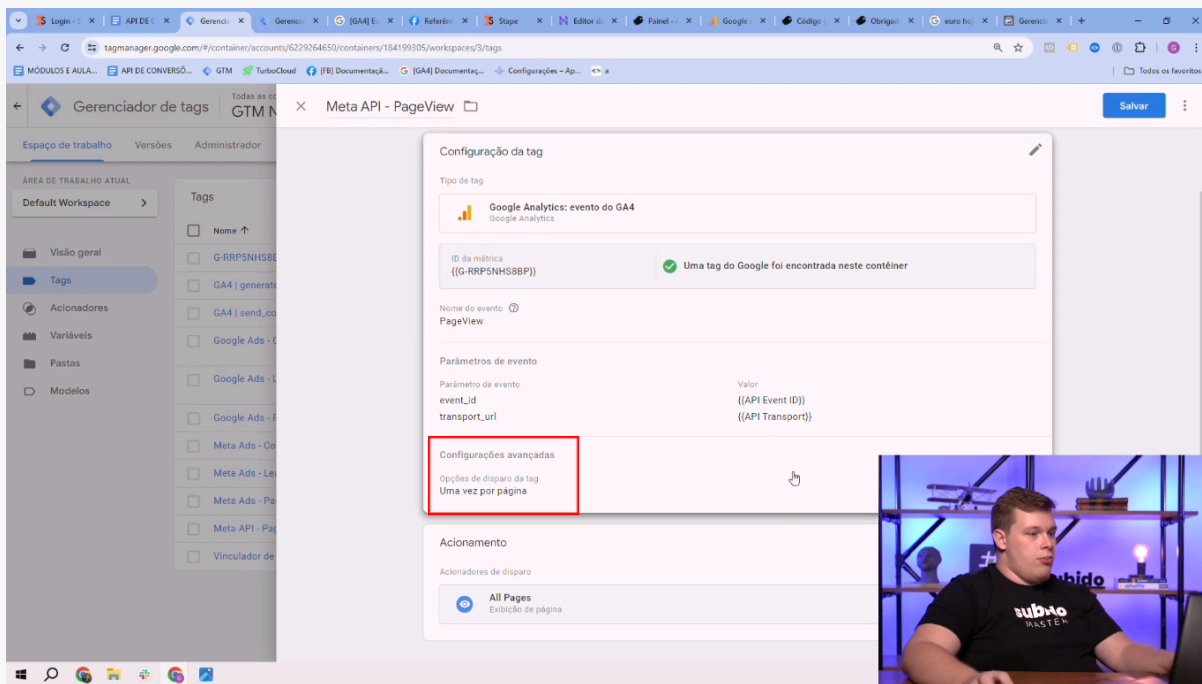


27. Feito isso, temos que adicionar dois parâmetros a essa tag: `event_id` (`{{API Event ID}}`) e `transport_url` (`{{API Transport}}`).

Então, vamos em parâmetros e fazemos as adições. Com isso configurado, salvamos a tag.



28. Nas configurações avançadas, tem que estar “Uma vez por página”



Agora, temos que fazer a mesma coisa para Lead e para Contact, mas é só duplicar essa tag de **Meta API - PageView**.

Para finalizar, a gente vai fazer o debug do container Server e do Web. A gente vai ver se os eventos de teste dão certo no gerenciador de eventos, depois vamos remover o *Test Event Code* e, por fim, vamos publicar tudo.

Fechou?

Com essa segunda parte, você já completa o seu treinamento Jedi e está pronto para fazer todas as configurações na sua API. Se ainda tiver alguma dúvida, leia e releia este PDF e o PDF anterior quantas vezes precisar.

Tamo junto!