Setting Up Facebook Conversion Tracking in GTM for Dentli Online Booking (Client-Side Pixel & Server-Side CAPI)

To help your agency track completed online bookings on Facebook, this guide covers two methods using Google Tag Manager (GTM): a **browser-based Meta Pixel implementation** (using Facebook's JS pixel) and a **server-side implementation** using **Facebook's Conversions API (CAPI)**. We will track a Dentli Online Booking event called **online_booking_completed**, which the booking widget pushes to the website's data layer when a booking is finished.

This guide assumes an intermediate understanding of conversion tracking.

Browser-Based Tracking with Facebook Pixel (Client-Side)

Using the Meta (Facebook) Pixel in GTM is the straightforward way to track conversions. In this approach, the user's browser runs Facebook's JavaScript to send the conversion event. We will use GTM's web container to fire the Pixel code when an online_booking_completed event occurs on the site (from the data layer). This method is easy to implement and is the foundation of Facebook tracking – Facebook itself recommends continuing to use the Pixel and supplementing it with server-side events for better data capture.

Why use the client-side Pixel?

It's the standard approach to report conversions (like completed bookings) directly from the browser. Facebook's Pixel allows you to track user actions (page views, sign-ups, purchases, etc.) and feed that data into Facebook Ads Manager for optimization and audience building. For our booking scenario, we can use a *standard Facebook event* called "**Schedule**" – Facebook provides a predefined *Schedule* event specifically for when someone **makes a booking**. Using standard events is recommended, as they are recognized by Facebook for ad optimization.

Steps to implement the Meta Pixel for online bookings in GTM:

1. Add the Meta Pixel base code in GTM:

Obtain your Facebook Pixel ID from the Facebook Events Manager (under *Data Sources*) and set up a new tag in your GTM web container for the Pixel base code. You can use a **Custom HTML tag** or the **Community Meta Pixel Template**. The base code (provided by Facebook) initializes the pixel with your Pixel ID and typically fires a default PageView. Configure this tag to fire on All Pages (so the pixel loads on every page).

Important: If you operate in a region with privacy laws (GDPR, etc.), **do not fire the pixel until the user has given consent**. In GTM you can implement Consent Checks or add blocking triggers to ensure the Pixel tag only runs when appropriate consent is present. This compliance step is crucial before tracking any user data.

2. Create a trigger for the booking completion event:

Next, set up a **Custom Event trigger** in GTM that listens for the online_booking_completed event. You may already have this trigger if you followed our guide on how to set up GTM events for Online Booking. If you do, skip the rest of this step.

Our booking widget pushes dataLayer.push({ event: 'online_booking_completed', ... }) when a booking is completed.

In GTM, create a Custom Event Trigger and set the Event Name to

"online_booking_completed", and set this trigger fires on **All Custom Events**. Click **Save**. This trigger will now fire the conversion tag that we associate it with, whenever a booking is completed.



3. Set up the Facebook Pixel event tag in GTM:

Now create a new tag in GTM for the Facebook conversion event. If using the Community Meta Pixel template, choose **Meta/Facebook Pixel** as the tag type; otherwise, use a Custom HTML tag with Facebook's fbq() call (aka. Their conversion tracking snippet). We will not go any further into using custom HTML in this guide.

In the community tag's configuration, input your Pixel ID (the same one you used in the base code). Then specify the event you want to send. **Use the standard "Schedule" event** for Dentli's Online Booking (shown below).

Important: If you operate in a region with privacy laws (GDPR, etc.), **do not fire the pixel until the user has given consent**. In GTM you can implement Consent Checks or add blocking triggers to ensure the Pixel tag only runs when appropriate consent is present. This compliance step is crucial before tracking any user data.

Tag Configuration	
Тад Туре	
Facebook Pixel facebookarchive	GALLERY
•• Tag permissions	4 permissions
Facebook Pixel ID(s)	
e.g. 12345678910	
Enhanced Ecommerce dataLayer Integration ③	
Event Name	
(Standard	
Schedule 👻	
Custom	
O Variable	
Consent Granted (GDPR)	
{{Consent State}}	
Enable Advanced Matching	
> Data Processing Options	
> Object Properties	
> More Settings	
> Advanced Settings	

Include conversion details (optional): If available, send additional parameters like the booking value or currency. For example, if the data layer provides a revenue or price variable, you can pass that to Meta for reporting. The Facebook Pixel template allows adding **Object Properties** (custom parameters) to the event. For instance, you would add a row for "currency" and "value" and use GTM variables that pull those from the data layer (e.g. currency code and transaction total). This way Meta will know the booking's value, which is useful for ads optimization and reporting.

Set the firing trigger for this tag to the **Custom Event trigger** you created ("online_booking_completed"). This ensures the conversion tag fires **only** when a booking is completed, not on all pages.

4. Test and validate the Pixel implementation:

Before publishing, test the setup thoroughly. Enable **Preview Mode** in GTM and perform a test booking on the site. In the GTM debug panel, confirm that when the booking is completed, the **Facebook Pixel event tag** fired (look for the online_booking_completed event and see that your Facebook tag was triggered). Also use the **Facebook Pixel Helper** browser extension to verify that the "Schedule" (or custom) event was sent to Facebook – it should show up with the correct parameters. Additionally, go to Facebook Events Manager > *Test Events* and see if your event is received there in real time. If everything looks good, publish your GTM container changes. Facebook will now record an "Online Booking Completed" conversion whenever a user completes a booking on your site via the Pixel.



GTM Preview mode showing a custom dataLayer event (registrationComplete in this example) in the event stream. You should see online_booking_completed when you trigger complete a booking.

Compliance Reminder

Ensure you **have user consent** before tracking. Sending data about your visitors to third-party platforms like Facebook Pixel should only happen after they agree, as required by regulations. Implement GTM's consent controls or triggers so that no Pixel events (especially those containing personal data) fire without the proper opt-in. *Never bypass consent*, and avoid sending any Personally Identifiable Information (PII) directly in Pixel events. (Facebook's Advanced Matching feature, which can send hashed emails, phone numbers, etc., is useful but **requires explicit user consent** as well.)

Server-Side Tracking with Facebook Conversions API (CAPI)

The second method uses Facebook's **Conversions API (CAPI)** via a GTM **server-side container**. This approach sends the conversion event from your server (or cloud instance) to Facebook, rather than directly from the user's browser. The advantage of server-side tracking is **greater data accuracy and resilience**: it's **less impacted by browser limitations, ad blockers, or iOS privacy restrictions**. Facebook recommends using the Conversions API alongside the Pixel to **improve event measurement and ads performance**, since you can capture conversions that the Pixel might miss. In fact, Meta has indicated that server-sent events are increasingly prioritized when both server and browser events are present. Some benefits of implementing CAPI include:

- More reliable conversion tracking: Server-side events aren't blocked by ad blockers or browser privacy features, leading to more complete data collection. This improved data connectivity means Facebook can attribute and optimize your campaigns more effectively (often lowering cost per action).
- Improved measurement accuracy: With CAPI, you'll see more conversions reported (especially from users on iOS 14+ or browsers that limit cookies). This helps in getting correct conversion counts and return on ad spend. Facebook notes that server events can improve measurement and reduce the cost per action by capturing data the pixel alone cannot.
- **Greater data control:** Using your own server endpoint lets you control what data is sent to Facebook. You can include extra details (e.g. an order ID, user email hash, etc.) or

filter out information. In short, **you own the data pipeline**, which can increase compliance and flexibility.

Note: The CAPI method is much more advanced. It requires setting up a GTM **Server container** (which runs on a cloud server you manage). If your or your team is new to server-side tagging, be prepared for an initial setup process. Facebook has a **Partner Integration** for GTM Server-Side to help connect your Business Manager to the GTM server container. Below is a high-level outline of the steps to implement CAPI for the online_booking_completed event:

Steps to implement Facebook Conversions API via GTM Server-Side:

- Deploy a GTM Server-Side container: In your GTM account, create a new Container of type "Server". You'll need to deploy this server container (for example, using Google Cloud App Engine or another cloud service) so that it has a live URL (endpoint) to receive events. GTM Server requires you to have a server environment – it won't work without one. Follow Google's instructions or use the automatic deployment to Google Cloud. It's recommended to set up a custom domain for your server container (e.g. stats.yourdomain.com) to maintain first-party context, though it's not strictly required for basic functionality. Once deployed, note the server container URL (you'll need it to send events from the web container).
 - Facebook CAPI credentials: In your Facebook Events Manager, generate an API Access Token for the Conversions API (under the Pixel settings or via the Setup tool) this token will authenticate your server events to Facebook. Also obtain your Pixel ID (same as used for the browser pixel) if you haven't already. These two pieces (Pixel ID and Access Token) are required to configure the Facebook server tag. (Optionally, you can also get a Test Event Code from Events Manager to use during testing.)
- 2. Forward the booking event from the web to the server: Now you need to send the online_booking_completed event from the website's GTM (browser) to your new GTM server container. The common approach is to use a Google Analytics 4 (GA4) event tag as a transport mechanism. If you already use GA4 on the site, you can leverage that; if not, you can set up a dedicated GA4 configuration and event tag just for this purpose. Here's how it works:
 - In the GTM web container: Create a GA4 Event tag that fires on the online_booking_completed event trigger (same trigger used for the Pixel).
 In this GA4 tag, input a Measurement ID (from a GA4 property you can create a placeholder GA4 property if you're not otherwise using GA4) and set up the event

parameters. Include key details like the event name (e.g. "online_booking_completed" or map it to a GA4 event like "purchase"), value, currency, transaction_id, etc., using variables from the data layer. **Crucially, set the tag's "Transport URL" field to your GTM Server container URL**. This tells the GA4 tag to send hit data to your server container instead of directly to Google's servers.

- When a booking completes, the GA4 tag will fire and send an HTTP request to your server container with all the event data (this happens behind the scenes in the browser). In the server container, ensure you have a GA4 Client set up (this client will "catch" incoming GA4 events). The GA4 client will parse the event and make its data available to server-side tags.
- Essentially, the flow is: User completes booking -> dataLayer event -> GTM web fires GA4 event (with booking data) to server -> GTM server receives GA4 event -> triggers Facebook CAPI tag. This pipeline allows the same booking event to be transmitted server-side.
- 4. Configure the Facebook CAPI tag in the server container: In your GTM Server container (accessible via GTM UI), add a tag to send the event to Facebook. GTM doesn't have a built-in Facebook tag by default, but you can add a community template. Go to Templates in the server container, search the gallery for "Facebook Conversions API Tag", and add it. Now create a new tag using this template. In the tag settings, enter the Facebook Pixel ID and the API Access Token you prepared. Set the Action Source to "Website" (since the conversion originated from your website. If you have a Test Event Code for debugging, you can enter it as well (this will route events to the Test Events tool in FB).
 - Event configuration: The Facebook CAPI tag will, by default, pick up event data from the incoming request (the GA4 client data). Ensure the tag knows when to fire: under **Triggering**, set the trigger to fire on the relevant event. A simple option is to trigger on "All Pages" in the server container (since only events coming via the GA4 client will be present anyway). A more precise method is to trigger on a custom condition: e.g., Trigger Type: Custom (Event), Condition: {{Event Name}} equals online_booking_completed and {{Client Name}} equals GA4. Or you can use a broader trigger where **Client Name** equals GA4 (meaning any GA4 event triggers it) as suggested in some guides. The key is that when the GA4 client processes an incoming booking event, it will activate the Facebook tag. The tag will read the event data (parameters you sent like value, currency, etc.) and send an equivalent server-to-server Facebook event. By default, the tag will use the same event name as the GA4 event – if you sent it as a GA4 "purchase" event, the FB tag might interpret it as a Purchase; if as a custom name, it may send a custom event. Many implementations map a GA4 "purchase" event to a Facebook "Purchase" or

"Lead" event in the tag settings. For a booking, you might configure the FB tag's event name to "Schedule" or similar. (*If using the GA4 event name directly, consider naming the GA4 event to match a standard FB event where possible.*)

- User data & advanced matching (optional): One advantage of server-side is the ability to send user information for better matching (e.g. email, phone, etc., which Facebook will hash). If your data layer or backend has the user's info (and you have user consent), you can include it in the GA4 event as user_data parameters (e.g. user_data.email_address). The FB CAPI tag will hash and forward these to improve match rates. Only do this if it's legally allowed – do not send personal data without proper consent and disclosure. Facebook's server tag will automatically hash personal identifiers, but you must still ensure you're compliant.
- 5. Test the server-side setup and ensure deduplication: Testing CAPI events is crucial. Publish your server container (in Preview mode you can use the built-in debugger). In Facebook Events Manager, go to Test Events, input your Test Event Code (if used), and then trigger a booking on your site. The server-side event should appear in Facebook's test console. Verify that the event and parameters (value, currency, etc.) are correct.

If you are **running both the Pixel and CAPI in parallel** (which is the ideal scenario), you need to prevent double-counting of the same conversion. Facebook uses an **Event ID** to deduplicate events that are sent via both methods. Implement a solution to generate a unique event_id for each booking and send it with both the browser Pixel event and the server CAPI event. For example, you can have the booking dataLayer push include an event_id, or generate one in GTM (perhaps in a variable) and attach it as a parameter in both the Pixel tag and the GA4 tag. Facebook will recognize the identical event_id and count the conversion only once. (*If you only use CAPI without a Pixel on that event, you don't need an event_id, since there's no duplicate source.*) Also, check that both browser and server events are attributed correctly in your Ads Manager.

6. Maintain compliance and monitor data: Just because the event is sent server-side doesn't exempt it from privacy rules. Ensure user consent for server-tracked events as well, just as you do for Pixel. From a user's perspective, an event sent via CAPI is still their personal data being shared. So, integrate your consent management so that if a user opts out, you disable the server event sending too. Additionally, avoid sending sensitive personal data in the event payload. If you include user identifiers for matching, they should be hashed and only sent when allowed. Keep your privacy policy updated to disclose use of both browser cookies (Pixel) and server tracking.

Best Practices and Final Tips

- Use the Pixel (JS) as a foundation: Start with the client-side Facebook Pixel setup it's simpler and immediately effective for tracking conversions. Make sure to use Facebook's standard events whenever possible (e.g. using the Schedule event for a booking) for better optimization and reporting. Custom events can still be used for tracking and building custom audiences, but standard events have the advantage of being recognized by Facebook's ad algorithms.
- Augment with Conversions API for better data: Implementing the server-side (CAPI) in addition to the Pixel can significantly improve your conversion tracking fidelity. It helps recover lost conversions (due to browsers blocking third-party cookies or users opting out of tracking on the browser) by sending them through a direct server channel. According to Meta, adding CAPI can improve measurement and reduce cost per action thanks to more complete data, and it gives you more control over the data you share. In practice, this often means your Facebook Ads reporting will show higher attribution and your campaigns' optimization algorithms will have more conversions to learn from, potentially boosting performance.
- Always respect user privacy and consent: Compliance isn't optional it's a key part
 of any tracking deployment. Obtain explicit consent before firing the Pixel or sending
 CAPI events for a user. In GTM, use Consent Mode or custom triggers to ensure no
 tracking runs without permission. If you plan to send personal data (for advanced
 matching on Facebook), double-check that you have the legal right to do so and that
 users have agreed to this level of data sharing. Clearly inform users in your site's privacy
 policy about the use of Facebook Pixel and Conversions API for tracking conversions.
- Test, monitor, and refine: After setup, monitor your Events in Facebook Events Manager. Use the Test Events tool and the Diagnostics section to catch any issues. Ensure that the online_booking_completed (Schedule) events are coming through with the correct parameters and that they're attributed properly in your ad reports. It's also wise to test scenarios with users who have not consented, to confirm that no unintended data is sent. Going forward, maintain your implementation – if your booking flow or data layer changes, update the GTM tags/triggers accordingly. And remember, you can always adjust what data you send via CAPI to optimize performance or compliance as needed (for example, sending a transaction_id for better deduplication or adding more user info if match rates are low, provided you have consent).

By following these steps, you will have a robust Facebook conversion tracking setup for "online_booking_completed" events. The **client-side Pixel** will handle immediate conversion tracking in the user's browser, and the **server-side Conversions API** will bolster your data by sending events from your server. This combined approach helps ensure Facebook receives the booking completion data reliably, which in turn helps your marketing team measure campaign success and optimize ads for what truly matters – completed bookings. **Always keep both data**

accuracy and user trust in mind: implement tracking in a way that is effective for your goals but also transparent and respectful to users. With the Pixel and CAPI configured in GTM, your agency and clients should see improved Facebook reporting and the ability to drive better optimizations for online bookings.

Good luck with your implementation!