Meta CAPI Integration

Funnel Step	Meta Event Name	Trigger Condition	Required Parameters	Custom Data Fields	Event_ID Convention
Landing Page View	ViewContent	User loads landing page	event_name, event_time, event_id, action_source=" website", event_source_ur l, user_data (phone/IP/UA if available)	utm_source, utm_medium, utm_campaign, utm_content, utm_term	viewcontent_<se ssionId>
CTA Click (to Telegram)	Lead	User clicks "Open in Telegram" button	Same as above	button_id, utm_*	cta_<sessionId>
Start Bot	Start_Bot (custom)	Telegram /start triggered	event_name, event_time, event_id, action_source=" app", user_data (telegram_user_id hashed)	utm_*, language, currency_pref	startbot_<teleg ramId>

First Expense Entry	`CustomEvent` (`First_Entry`)	User logs first expense	`event_name`, `event_time`, `event_id`, `action_source="app"`	`entry_mode` (voice/photo/text), `amount`, `category_assigned`, `currency`	`firstentry_<telegramId>`
Subscription	`Subscribe`	User pays recurring	`event_name`, `event_time`, `event_id`, `action_source="app"`, `currency`, `value` (subscription price)	`subscription_plan` (monthly/weekly), `payment_method`, `billing_cycle_day`	`sub_<transactionId>`
Cancel Subscription	`Unsubscribe`	User cancels paid plan	Same as above	`subscription_plan`, `reason` (if collected), `days_active`	`cancel_<transactionId>`
Reactivate Subscription	`CustomEvent` (`Reactivate_Sub`)	User reactivates subscription	Same as above	`subscription_plan`, `discount_code` (if used)	`reactivate_<transactionId>`
Upgrade/Downgrade Plan	`CustomEvent` (`Plan_Change`)	User changes subscription tier	Same as above	`old_plan`, `new_plan`, `value_diff`, `currency`	`planchange_<transactionId>`

Error/Drop-off Event	CustomEvent (Drop_Off)	User quits during onboarding or checkout	Same as above	step_at_exit, reason (if known), session_duration	dropoff_<sessionId>

Onboarding – Step 1 (Language Selection)	CompleteRegistration	User selects preferred language	event_name, event_time, event_id, action_source, user_data	language_selected	onboard_lang_{userID}_{timestamp}
Onboarding – Step 2 (Name Input)	CompleteRegistration	User enters name	same as above	user_name	onboard_name_{userID}_{timestamp}
Onboarding – Step 3 (Phone Number Input)	CompleteRegistration	User provides phone number	same as above	phone_number	onboard_phone_{userID}_{timestamp}
Onboarding – Step 4 (Currency Selection)	CompleteRegistration	User selects main currency (USD/UZS)	same as above	currency_selected	onboard_currency_{userID}_{timestamp}
Onboarding – Step 5 (Start Capital Input)	CompleteRegistration	User sets initial capital	same as above	start_capital_amount	onboard_capital_{userID}_{timestamp}

Complete Onboarding	`CompleteRegistration`	User completes onboarding (5 steps)	`event_name`, `event_time`, `event_id`, `action_source="app"`, `user_data` (telegram_user_id hashed)	`step_count=5`, `language_selected`, `user_name`, `phone_number`, `Currency_selected`, `start_capital_amount`	`onboarding_<telegramId>`
App Open / Session Start	`AppLaunch`	User opens bot after being inactive for 12h+	`event_name`, `event_time`, `event_id`, `action_source`, `user_data`	`session_id`, `app_version`	`session_{userID}_{timestamp}`
7-Day Retention	`7d_Return` (custom)	User returns and logs again within 7 days	Same as above	`days_since_start`, `entries_count`	`ret7_<telegramId>`
Retention – 14 Days	`14d_Return` (custom event)	User active on day 14 after registration	`event_name`, `event_time`, `event_id`, `action_source`, `user_data`	`days_active=14`	`retention14_{userID}`
Retention – 1 Month	`1m_Return` (custom event)	User active on day 30	same as above	`days_active=30`	`retention30_{userID}`

Retention – 2 Months	2m_Return (custom event)	User active on day 60	same as above	days_active=60	retention60_{userID}
Retention – 3 Months	3m_Return (custom event)	User active on day 90	same as above	days_active=90	retention90_{userID}
Retention – 6 Months	6m_Return (custom event)	User active on day 180	same as above	days_active=180	retention180_{userID}
Retention – 12 Months	12m_Return (custom event)	User active on day 365	same as above	days_active=365	retention365_{userID}

Notes for Dev

- **event_id**: Always unique. Use `<eventType>_<telegramId>` or `<eventType>_<transactionId>` to prevent duplication.
- **user_data**: Must be **hashed (SHA256)** before sending (email, phone, telegram_user_id if mapped, IP, UA).
- **currency/value**: Required for any revenue-linked event (`Purchase`, `Subscribe`, `StartTrial`).
- **action_source**:
  - `"website"` → LP or web events
  - `"app"` → Telegram bot events
- **event_source_url**: Only needed for LP/website events.

Meta CAPI events and parameters we'll send

Notes

- All user_data fields that identify a person are SHA256-hashed server-side: email (em), phone (ph), external_id (Telegram ID). client_ip_address and client_user_agent are NOT hashed per Meta spec.
- Website events include event_source_url. App/bot/server events omit it.
- Every event includes: event_time (unix sec), event_name, action_source, event_id.

Event	event_name	action_source	user_data (hashed)	extra user_data	custom_data we send	event_id format	Sent from
Landing Page View	ViewContent	website	—	client_ip_address, client_user_agent	utm_source, utm_medium, utm_campaign, utm_content, utm_term	viewcontent_	Website → /api/marketing/meta
CTA Click (to Telegram)	Lead	website	—	client_ip_address, client_user_agent	button_id, utm_*	cta_	Website → /api/marketing/meta
Start Bot	Start_Bot (custom)	app	external_id, ph?	—	language, currency_pref, utm_*	startbot_	Bot → /api/marketing/meta

Event	event_name	action_source	user_data (hashed)	extra user_data	custom_data we send	event_id format	Sent from
First Expense Entry	First_Entry (custom)	app	external_id, ph?	—	entry_mode (voice/photo/text), amount, category_assigned, currency	firstentry_	Mini app/bot → `/api/marketing/meta`
Subscribe	Subscribe	app	external_id, ph?	—	currency, value (price), subscription_plan (weekly/monthly), payment_method, billing_cycle_day, transaction_id	sub_	Server webhook (Atmos)
Cancel Subscription	Unsubscribe	app	external_id	—	subscription_plan, reason?, days_active?	cancel_	Server webhook (Atmos)
Reactivate Subscription	Reactivate_Sub (custom)	app	external_id	—	subscription_plan, discount_code?, currency, value	reactivate_	Subscription API (renew)

Event	event_name	action_source	user_data (hashed)	extra user_data	custom_data we send	event_id format	Sent from
Upgrade/Downgrade	Plan_Change (custom)	app	external_id	—	old_plan, new_plan, value_diff, currency	planchange_	Server webhook (Atmos)
Error/Drop-off	Drop_Off (custom)	website/app	external_id?	client_ip_address, client_user_agent (if web)	step_at_exit, reason?, session_duration	dropoff_	Website/app
Onboarding – Language	CompleteRegistration	app	external_id, ph?	—	status="language", language_selected	onboard_lang_{userID}_{ts}	Bot
Onboarding – Name	CompleteRegistration	app	external_id	—	status="name", user_name	onboard_name_{userID}_{ts}	Bot
Onboarding – Phone	CompleteRegistration	app	external_id, ph	—	status="phone"	onboard_phone_{userID}_{ts}	Bot
Onboarding – Currency	CompleteRegistration	app	external_id	—	status="currency", currency_selected	onboard_currency_{userID}_{ts}	Bot

Event	event_name	action_source	user_data (hashed)	extra user_data	custom_data we send	event_id format	Sent from
Onboarding – Start Capital	CompleteRegistration	app	external_id	—	status="capital", start_capital_amount	onboard_capital_{userID}_{ts}	Bot
Complete Onboarding	CompleteRegistration	app	external_id, ph	—	step_count=5, language_selected, user_name, phone_number, currency_selected, start_capital_amount	onboarding_	Bot
App Open / Session Start	AppLaunch	app	external_id	—	session_id, app_version?	session_{userID}_{ts}	Mini app
7-day Return	7d_Return (custom)	app	external_id	—	days_since_start, entries_count	ret7_	Mini app/bot
14/30/60/90/180/365 Return	14d_Return / 1m_Return / 2m_Return / 3m_Return / 6m_Return /	app	external_id	—	days_active	retention{N}_{userID}	Mini app/bot

Event	event_name	action_source	user_data (hashed)	extra user_data	custom_data we send	event_id format	Sent from
	12m_Return (custom)						

If you want additional matching params (when available), we can also include: email (em), country, city, state, zip; fbc/fbp; content_name/content_ids/contents for LP events.

- Endpoint used: `/api/marketing/meta` (server adds client_ip_address and client_user_agent; hashes user_data; logs to `marketing_events`).