TravelTide — Feature Dictionary & Rationale (SQL Output)

Scope. This document describes the user-level features produced by the final SQL extraction used to create `TravelTideFinal.csv`.
Cohort. Sessions with `session_start > '2023-01-04'`; users with > 7 sessions are retained.
Trip states. For any `trip_id`, exactly one of: **0)** no booking occurred, **1)** booking occurred & trip completed, **2)** booking occurred & trip cancelled.
Level. One row per `user_id`.

> **Note:** Columns like `log_*`, `cluster`, `cluster_name`, `perk`, and `trip_duration_type` are added later in the notebook and are *not* part of the SQL export.

Identifiers & Demographics

user_id

- **Definition:** Unique user identifier.
- **Why:** Join key across tables; basis for user-level aggregation.

age

- **Definition:** `DATE_PART('year', AGE(session_start, birthdate))` averaged over the user's sessions, then rounded.
- **Why:** Age bands can correlate with travel cadence, trip length, and spend patterns.
- **Notes:** Computed relative to session times; rounded to reduce noise.

married, has_children

- **Definition:** Binary demographic flags (cast to integers) aggregated with `MIN()` at user level.
- **Why:** Household composition can influence party size, bag usage, and hotel vs. flight

propensity.

- **Notes:** `MIN()` preserves a “true if ever true” behavior given binary casting in this cohort.

Engagement & Session Quality

total_sessions

- **Definition:** Count of sessions per user within cohort window.
- **Why:** Activity volume proxy; supports normalization of other rates if needed.

total_clicks

- **Definition:** Sum of `page_clicks` across sessions.
- **Why:** Engagement depth; higher can signal interest or friction.

avg_session_duration_sec

- **Definition:** Average of `(session_end - session_start)` in seconds, rounded.
- **Why:** Attention proxy; very short or very long can indicate distinct behaviors (decisive vs. exploratory).

Conversion & Trip Completion

total_completed_trips

- **Definition:** Count of distinct `trip_id` present in `completed_trips` (i.e., there exists at least one **non-cancellation** session for the `trip_id` **and** there are **no** cancellation sessions for that `trip_id`). Any `trip_id` with a cancellation session is dropped.
- **Why:** Realized value; anchors monetization metrics.

booking_conversion_rate

- **Definition:** `ROUND(COUNT(DISTINCT trip_id) / COUNT(*), 2)` at user level.
- **Why:** Session-to-trip efficiency proxy; helps separate browsers from bookers.
- **Notes:** Session-based denominator by design.

Baggage, Distance & Spend

total_checked_bags

- **Definition:** Sum of `checked_bags` for trips counted as completed.
- **Why:** Signal for Free Checked Bag perk; correlates with party size and route type.

total_distance_km

- **Definition:** Sum of Haversine distance between home and destination airports over completed trips; rounded.
- **Why:** Travel radius; helps distinguish local vs. long-haul travelers.

money_spent_flight

- **Definition:** For sessions with `flight_booked = TRUE` & `cancellation = FALSE`: `base_fare_usd * seats * (2 if return_flight_booked else 1) * (1 - COALESCE(flight_discount_amount, 0))`, summed.
- **Why:** Monetization signal; indicates value of flight relationship.
- **Notes:** Discount-adjusted; round-trip multiplier included.

money_spent_hotel

- **Definition:** For sessions with `hotel_booked = TRUE` & `cancellation = FALSE`: `hotel_per_room_usd * rooms * nights * (1 - COALESCE(hotel_discount_amount, 0))`, summed, where `nights = GREATEST(check_out_time::date, check_in_time::date) -`

`LEAST(check_out_time::date, check_in_time::date)`.

- **Why:** Hotel monetization; supports perks like Free Hotel Meal or Free Night.

hotel_loyalty_score

- **Definition:** `1 / (# of distinct hotel brands)` booked on completed trips (0 if none), rounded to 3 decimals.
- **Why:** Brand concentration proxy; higher = more loyal to a brand (candidate for hotel-centric perks).

Timing & Duration

avg_days_booking_to_trip

- **Definition:** Average days from booking session end to either flight departure or hotel check-in (completed trips only), rounded.
- **Why:** Lead-time behavior; useful for messaging cadence and cancellation sensitivity signals.

avg_trip_duration_days

- **Definition:** Average of `(return_time - departure_time)` for flights, else `(check_out - check_in)` for hotel-only trips (completed trips only), rounded.
- **Why:** Trip length profile; separates short breaks from longer stays.

Product Mix & Incentives

flight_only_rate

- **Definition:** Share of distinct trips that are flight only (booked flight, no hotel, not cancelled), rounded to 2 decimals.

- **Why:** Product preference indicator; informs perk relevance.

hotel_only_rate

- **Definition:** Share of distinct trips that are hotel only (booked hotel, no flight, not cancelled), rounded to 2 decimals.
- **Why:** Hotel inclination; supports hotel-first messaging and perks.

both_booked_rate

- **Definition:** Share of distinct trips with both flight and hotel booked (not cancelled), rounded to 2 decimals.
- **Why:** Bundling propensity; supports Free Night with Flight positioning.

discount_usage_rate

- **Definition:** Share of distinct trips where a discount was present and a booking occurred (not cancelled), rounded to 2 decimals.
- **Why:** Price sensitivity proxy; candidates for Exclusive Discounts messaging.

cancellation_per_booking_rate

- **Definition:** Share of distinct trips with cancellation events, rounded to 2 decimals.
- **Why:** Volatility/flex preference; candidates for Free Cancellation emphasis.

Notebook-Added (Not from SQL)

- **log_money_spent_flight, log_money_spent_hotel, log_avg_days_booking_to_trip**
  Why: Skew reduction and interpretability checks during modeling.
- **cluster, cluster_name, perk, trip_duration_type**
  Why: Final segmentation outputs and presentation helpers.

Provenance & Assumptions

- Cohort filters and session constraints applied as in the SQL.
- `completed_trips` reflects booked-and-not-cancelled logic; aligned with the three trip states.
- Rounding is applied to several metrics for readability; raw values were used for modeling where appropriate.

Known Data Issues & Workarounds (Important)

1) Negative `nights` in hotels

- **Issue:** Some rows had `nights` ≤ 0 due to inconsistent `check_in_time` / `check_out_time`.
- **Fix used in SQL:** Compute nights as a row-wise max–min of the timestamps: `GREATEST(check_out_time::date, check_in_time::date) - LEAST(check_out_time::date, check_in_time::date)`
- **Impact:** Eliminates negative durations and preserves valid long stays. All hotel spend and trip-duration metrics derived from this are now non-negative and consistent.

2) Cancellation sessions flip core flags to TRUE

- **Issue:** Any cancellation session had `flight_discount`, `hotel_discount`, `flight_booked`, and `hotel_booked` all set to TRUE, regardless of the actual booking session state.
- **Workarounds we adopted:**
    - Do not rely on raw session flags from cancellation sessions for conversions/discount usage.
    - Use the `completed_trips` CTE wherever we need finalized behavior (booked & not cancelled).
    - For discount usage and related rates, count only non-cancellation cases; cancellation sessions are explicitly excluded in the logic.

- **Interpretation tip:** Rates that summarize finalized bookings are safe. Where we intentionally use session-level signals, treat them as intent rather than confirmed outcomes.

3) `hotel_name` contains brand + city in one field

- **Issue:** `hotel_name` encodes both brand and city (e.g., "Brand - City").
- **Approach:** Split on `' - '` and take the brand portion for loyalty measures: `SPLIT_PART(hotel_name, ' - ', 1)`
- **Assumption:** Stable delimiter pattern "brand - city". If the delimiter is missing, the full string acts as the brand token.
- **Why:** Loyalty logic should reflect brand concentration, not city variety.

4) Why some features don't use `completed_trips`

- **Intent vs. Outcome:** Not all aggregates flow through `completed_trips` on purpose. Session-based engagement or conversion-style ratios using session counts can be informative about initial intent, funnel behavior, or friction *before* a booking is finalized.
- **Balance with cancellations:** We surface `cancellation_per_booking_rate` so that intent-heavy users who later cancel are contextualized.
- **Rule of thumb:** Outcome-focused metrics → use `completed_trips`. Intent-level metrics → may be session-based, then interpreted together with cancellation rate.

Practical Implications for Reuse

- For **KPI dashboards** or **new models**:
    - Prefer the **trip-level** (completed) metrics for performance & monetization views.
    - Use **session-based** features for diagnosing funnel behavior/intent and always read them alongside `cancellation_per_booking_rate`.
    - Treat the split of `hotel_name` as a **brand proxy**, not a perfect taxonomy.

This dictionary accompanies the final SQL query and `TravelTideFinal.csv` and is intended for stakeholders and

technical peers to understand each feature’s logic and purpose.