**Capstone Build Summary — NBA Ticketing & Renewal Optimization**

**Architecture & Data Model**

* **Azure SQL Database** created (serverless).
* **Star schema** stood up with T-SQL: DimDate, DimGame, DimSeat, DimCustomer, DimSalesTerritory, DimChannel, DimPlan, plus facts FactTicketSales, FactAttendance, FactRenewals, FactForecast, and staging stg\_\*.
* **Partition strategy** implemented:
  + FactTicketSales by **SaleDate**
  + FactAttendance by **GameDate**
* **RLS design**: SecurityUserMap maps **UPN → TerritoryId**; enforced in Power BI (no rep-level exceptions).
* **Indexes** added (including IX\_FactForecast\_GameDateMetric) for report performance.

**Ingestion & ETL (2A–2B)**

* Loaded synthetic source files to **staging** with bcp (handled UTF-8, row terminators, header skipping).
* Fixed a schema mismatch in games.csv via helper table stg\_Game\_csv → mapped into stg\_Game.
* Generated and loaded **ticket sales** in five chunks; deduped stg\_TicketSales (resolved MERGE 8672 duplicates).
* Executed loader procs:
  + Dims: sp\_Load\_DimGame/Seat/Customer/Channel/Plan/Territory, sp\_Load\_SecurityUserMap
  + Facts: sp\_Load\_FactTicketSales, sp\_Load\_FactAttendance, sp\_Load\_FactRenewals
* **Row sanity** confirmed (Dims populated; Facts filled from staged files).

**Forecasting Pipeline (2D)**

* Built **forecast\_job.py**:
  + **Aggregate Attendance (GameId=0)**: Prophet (weekly seasonality), with MA28 fallback.
  + **Aggregate Revenue (GameId=0)**: **MA28 + weekend factor** (always runs).
  + **Per-game forecasts**: rule-based using trailing averages + weekend/promo multipliers, one row per future game per metric.
  + Safety: clears future-dated rows before insert to avoid duplicates.
* Successful runs wrote:
  + Initially **56 rows** (attendance only), then **112 rows** (attendance + revenue) aggregate; plus per-game rows as applicable.
* **FactForecast** relationships corrected:
  + DimDate[Date] (1) → FactForecast[ForecastDate] (\*)
  + DimGame[GameId] (1) → FactForecast[GameId] (\*)

**Power BI Model (2C)**

* Connected to Azure SQL in **Import** mode.
* Created **RangeStart/RangeEnd** parameters and applied folding filters:
  + FactTicketSales[SaleDate] between parameters
  + vw\_FactAttendance[GameDate] between parameters
* Added SQL **view** vw\_FactAttendance (joins GameDate) to preserve query folding for Incremental Refresh.
* **Incremental Refresh** configured: **Store 5y**, **Refresh last 60d**, **Detect changes = ModifiedAt** (both tickets & attendance).
* **Relationships** verified:
  + DimDate → FactTicketSales, DimDate → vw\_FactAttendance, DimDate → FactForecast
  + DimGame → FactTicketSales / vw\_FactAttendance / FactForecast
* **RLS role** TerritoryUser implemented with UPN→TerritoryId mapping.
* **Measures**:
  + Core revenue/volume/attendance/renewal stack
  + Time-intel & YoY
  + Pricing & demand (Discount Rate, Fee Rate, Underpriced Weeknight flag, Avg Days to Game, ARPA)
  + Forecast hooks (Forecast vs Actual, variance, MAPE)
  + Delivered **30+ measures** total (plus 15 extra KPI measures we added).

**Visuals & Pages**

* **Exec KPI**: Net Revenue, Tickets, ATP, YoY, Discount Rate; Revenue vs LY; Gross→Net waterfall; Channel mix.
* **Pricing & Demand**: ATP by Zone/Opponent; underpriced weeknights table; lead-time distribution; discount by channel; ARPA.
* **Attendance & Ops**: Attendance trend; Show vs No-Show; Show Rate; Weekend/Promo uplifts; revenue vs tickets scatter.
* **Renewals**: Renewal Rate, Churn %, Renewal Revenue, Upsell %, funnel; plan/territory breakdowns.
* **Game Drill-through**: game header, revenue/attendance cards, ATP vs discount by zone, show/no-show donut, lead-time, optional seat map.
* **Forecast**: Attendance & Revenue 8-week lines, variance bars, MAPE.

**Orchestration & Automation (2E)**

* Created **ADF pipeline** PL\_Master\_Daily:
  + One Stored Procedure activity running **dbo.sp\_Run\_Daily\_Load** (calls all loaders in order).
  + **Daily trigger @ 03:00 AM** (Central Time) enabled.
* Documented option to schedule **forecast\_job.py** nightly (3:15 AM) via Azure Automation or ADF custom activity.
* Provided monitoring, alerting, and troubleshooting runbook (folding errors, relationship duplicates, staging dedupe, etc.)

**Issues Solved (so there is no more manual adjustments later)**

* **bcp 0-row loads** → enforced UTF-8 + newline flags and fixed games schema via helper table.
* **MERGE duplicate key** on ticket facts → deduped staging and hardened loader pattern.
* **Power BI relationship error** (“duplicate on one side”) → corrected FactForecast to the **many** side.
* **Incremental Refresh folding** for attendance → introduced vw\_FactAttendance to expose GameDate without breaking folding.
* **Revenue forecast with short history** → added **MA28** fallback so the line is always there.

**Outcome vs. Target**

* **Unified data → insight**: Azure SQL star-schema with automated daily loads (ADF)
* **Actionable visuals**: 30+ measures, drill-through game page, cohort & pricing views, exec scorecards; **RLS** by territory
* **Forecasting**: Prophet + rule-based revenue; **aggregate + per-game** forecasts; hooks for accuracy (MAPE)
* **Performance**: Incremental Refresh set (5y/60d); indexes applied; staging excluded from model load
* **Revenue impact levers**: Flags for **under-priced weeknights**, uplift calculations for **promo/weekend**, renewal upsell %