PROJECT SPECIFICATION

SaaS Revenue Lifecycle Analyzer

A Decision-Driven Revenue Intelligence Platform

Executive Summary

This project builds a unified analytics platform that identifies revenue leakage across the entire customer lifecycle: from lead acquisition through conversion, retention, and expansion. Unlike typical portfolio projects that analyze funnel conversion OR churn prediction in isolation, this platform treats them as interconnected parts of a single revenue engine.

The core differentiator: every output answers ***"What should the business do Monday morning?"*** with quantified dollar impact and confidence intervals.

The Business Pitch

"We built a Revenue Intelligence Platform that answers the question every CRO asks: Where are we losing ARR across the customer lifecycle, and what's the highest-ROI fix?"

The platform surfaces three critical insights:

1. **Funnel Leakage:** Which stages lose the most pipeline value, segmented by channel, deal size, and rep
2. **Retention Risk:** Which customers are likely to churn and how much ARR is at risk next quarter
3. **Prioritized Actions:** Top 3 initiatives ranked by expected ARR impact and ROI, with what-if scenario modeling

Why This Project Stands Out

Differentiation from Typical Portfolio Projects

Typical Project	This Project	Why It Matters
Uses Kaggle Telco Churn dataset	Generates realistic synthetic SaaS data with documented assumptions	Shows domain understanding, not just tool proficiency

Optimizes for model accuracy (AUC, F1)	Optimizes for calibrated probabilities + decision thresholds	VPs don't care about AUC; they care if the probabilities are actionable
Analyzes funnel OR churn separately	Unified view of CAC, LTV, and LTV:CAC ratio by segment	Mirrors how CROs and RevOps teams actually think
Delivers Jupyter notebook	Ships interactive dashboard + executive memo	Demonstrates product thinking and communication skills
Shows charts without recommendations	Every output includes "so what" action + dollar impact	Proves you can translate analysis into business decisions

Project Architecture

System Components

The platform consists of four integrated layers:

4. **Data Layer:** Synthetic SaaS data generator + schema mimicking HubSpot/Salesforce CRM
5. **Analysis Layer:** Python analytics engine for funnel diagnostics, churn modeling, and revenue calculations
6. **Presentation Layer:** Interactive Streamlit dashboard with what-if simulator
7. **Documentation Layer:** Executive memo, methodology docs, and business-centric README

Data Model Specification

Core Entities

Generate synthetic data that demonstrates domain expertise. Document all assumptions explicitly in the codebase and README.

1. Leads

Field	Type	Notes
lead_id	UUID	Primary key
created_at	timestamp	Lead creation date
source_channel	enum	organic, paid_search, paid_social, referral, outbound, event
company_size	enum	smb (<50), mid_market (50-500), enterprise (>500)
industry	enum	tech, healthcare, finance, retail, manufacturing, other
estimated_deal_size	float	Initial ARR estimate based on company_size tier
assigned_rep_id	UUID	FK to sales_reps

2. Opportunities (Stage History)

Field	Type	Notes
opportunity_id	UUID	Primary key

lead_id	UUID	FK to leads
current_stage	enum	lead, mql, sql, proposal, negotiation, closed_won, closed_lost
stage_entered_at	timestamp	When opportunity entered current stage
deal_value	float	Actual ARR if closed_won, else projected
close_date	date	Actual or expected close date
loss_reason	enum	If closed_lost: budget, competition, timing, no_decision, other

3. Stage Transitions (for time-to-stage analysis)

Field	Type	Notes
transition_id	UUID	Primary key
opportunity_id	UUID	FK to opportunities
from_stage	enum	Previous stage (null if first)
to_stage	enum	New stage
transitioned_at	timestamp	Enables time-to-stage and velocity analysis

4. Customers

Field	Type	Notes
customer_id	UUID	Primary key
opportunity_id	UUID	FK to winning opportunity
start_date	date	Contract start
contract_type	enum	monthly, annual, multi_year
mrr	float	Monthly recurring revenue
product_tier	enum	starter, professional, enterprise
churned_at	date	Null if active
churn_reason	enum	If churned: price, competition, no_value, company_closed, other

5. Usage Events (for health scoring)

Field	Type	Notes
event_id	UUID	Primary key
customer_id	UUID	FK to customers
event_date	date	Date of activity
logins	int	Daily login count
features_used	int	Count of distinct features accessed
support_tickets	int	Support tickets opened
api_calls	int	API usage (if applicable)

6. Sales Reps

Field	Type	Notes
rep_id	UUID	Primary key
name	string	Rep name
region	enum	west, central, east, international
tenure_months	int	For rep vs territory analysis

segment	enum	smb, mid_market, enterprise (assigned territory)

Analysis Requirements

Module 1: Funnel Diagnostics

Goal: Identify where pipeline leaks and quantify the dollar impact of each drop-off point.

Required Analyses:

- **Stage Conversion Rates:** Calculate conversion percentage at each stage transition (lead→MQL→SQL→proposal→negotiation→closed)
- **Segmentation:** Break down conversions by channel, deal size tier, company size, industry, and sales rep
- **Time-to-Stage (Velocity):** Calculate median and p75 days between stages; flag abnormally slow deals
- **Cohort Analysis:** Track conversion by lead cohort week/month to identify trends over time
- **Loss Reason Analysis:** Aggregate closed_lost by reason and stage where deals died
- **Rep Performance (with controls):** Separate "rep effect" from "territory/lead quality effect" using logistic regression controlling for deal size, channel, region

Key Outputs:

- Customer Acquisition Cost (CAC) by channel: Total marketing/sales spend ÷ customers acquired
- Pipeline value lost at each stage (not just count, but $)
- "If we improve [stage] conversion by X%, expected incremental ARR = $Y (with confidence interval)"

Module 2: Churn Prediction & Retention Analysis

Goal: Predict which customers will churn, calculate ARR at risk, and identify actionable retention levers.

Required Analyses:

- **Churn Prediction Model:** Train model to predict probability of churn within next 90 days
  - Features: contract_type, tenure, mrr, product_tier, usage metrics (logins, features_used), support_tickets, time_since_last_login
  - Critical: Use time-based train/test split to avoid data leakage
- **Model Calibration:** Ensure predicted probabilities are accurate (reliability diagram). A VP needs to trust that "30% churn risk" actually means 30%
- **Customer Health Score:** Combine churn probability with usage frequency into Green/Yellow/Red status for Account Executives
- **Churn Driver Analysis:** Identify which factors most strongly predict churn (feature importance + SHAP values)
- **Segment Risk Profiles:** Which customer segments (by contract type, product tier, company size) have highest churn rates?

Key Outputs:

- Revenue at Risk Next Quarter: Sum of (MRR × 12 × churn_probability) for all customers
- Lifetime Value (LTV) by segment
- Net Revenue Retention (NRR) calculation
- Intervention threshold: "Intervene on customers where (expected saved ARR − cost of intervention) is maximized"

Module 3: Unified Revenue Intelligence

Goal: Connect acquisition and retention into a single view that answers: "Where should we invest next?"

Required Calculations:

- **LTV:CAC Ratio by Segment:** The north star metric for SaaS unit economics. Calculate by channel, company size, and product tier
- **Payback Period:** Months to recover CAC, by segment
- **Revenue Leakage Waterfall:** Visual showing total addressable pipeline → lost at each funnel stage → won → churned → retained ARR
- **Action Prioritization Matrix:** Rank potential improvements by (expected ARR impact) × (confidence) ÷ (estimated effort)

What-If Simulator Requirements:

- User inputs: "Improve [stage] conversion by [X]%" or "Reduce churn by [Y]%"
- Output: Projected ARR impact with confidence interval
- Include assumptions panel showing baseline rates and data quality caveats

Deliverables Specification

1. Interactive Dashboard (Streamlit)

A functional web application that hiring managers can interact with directly.

Required Views:

- **Executive Summary:** Key metrics (total pipeline, conversion rate, ARR at risk, LTV:CAC) with trend indicators
- **Funnel Analysis:** Interactive funnel visualization with segment filters
- **Customer Health:** Sortable table of customers with health score, churn probability, MRR, and recommended action
- **Revenue at Risk:** Breakdown by segment with drill-down capability
- **What-If Simulator:** Sliders for adjusting conversion rates and churn, with real-time ARR impact calculation
- **Prioritized Actions:** Top 3 recommendations with expected impact and confidence level

2. Executive Memo (1-2 pages)

A professional written summary demonstrating business communication skills.

Structure:

- **Situation:** Brief context on the revenue leakage problem
- **Findings:** Top 3 insights with specific dollar amounts
- **Recommendations:** Prioritized actions with expected ROI
- **Next Steps:** Specific asks for stakeholders

3. GitHub Repository

README Requirements (Business-First):

- Lead with the business pitch, not technical stack
- Include screenshot/GIF of the dashboard
- Link to live demo (Streamlit Cloud or similar)
- "Key Findings" section with 3 headline insights
- Technical details below the fold

Code Organization:

- data/ — Synthetic data generator with documented assumptions
- analysis/ — Modular Python scripts for funnel and churn analysis
- models/ — Trained models with evaluation metrics
- app/ — Streamlit dashboard code
- docs/ — Executive memo, methodology documentation

Technical Implementation Guide

Recommended Tech Stack

Component	Technology	Rationale
Data Generation	Python (Faker, NumPy)	Realistic synthetic data with controlled distributions
Data Storage	SQLite or DuckDB	Lightweight, portable, demonstrates SQL proficiency
Analysis	Pandas, NumPy, SciPy	Industry standard for data manipulation
ML Models	scikit-learn, XGBoost	Well-documented, production-ready
Model Interpretation	SHAP	Explainable feature importance
Visualization	Plotly	Interactive charts for dashboard
Dashboard	Streamlit	Rapid deployment, easy hiring manager access
Deployment	Streamlit Cloud	Free, shareable URL

Synthetic Data Generation Guidelines

The synthetic data should demonstrate domain understanding. Document all assumptions in the code and README.

Realistic Parameters to Implement:

- Lead volume: 500-1000 leads per month with seasonal variation
- Conversion rates: Lead→MQL ~25%, MQL→SQL ~40%, SQL→Closed Won ~20% (vary by segment)
- Deal sizes: SMB $500-2K MRR, Mid-Market $2K-10K, Enterprise $10K-50K
- Sales cycle: SMB 15-30 days, Mid-Market 30-60 days, Enterprise 60-120 days
- Churn rates: Monthly contracts ~5%/month, Annual ~15%/year, Multi-year ~8%/year
- Usage correlation: Lower usage = higher churn probability (build this relationship into the data)
- Rep variation: Some reps 20% better/worse than average (for rep effect analysis)

Model Development Best Practices

- **Time-based splits:** Train on months 1-9, validate on months 10-11, test on month 12. Never random split for time-series churn data.

- **Calibration:** Use Platt scaling or isotonic regression. Include reliability diagram in documentation.
- **Threshold selection:** Don't use 0.5. Find optimal threshold that maximizes (saved ARR - intervention cost).
- **Feature leakage check:** Ensure no features use information from after the prediction point.

Quality Checklist

Before considering this project complete, verify the following:

Business Quality

- Every chart and metric answers "so what?" with a clear business implication
- All recommendations include dollar impact estimates
- LTV:CAC ratio is calculated and prominently featured
- What-if simulator produces realistic, bounded outputs
- Executive memo is clear, concise, and jargon-free

Technical Quality

- No data leakage in churn model (verified with time-based split)
- Model probabilities are calibrated (reliability diagram included)
- Confidence intervals included for key estimates
- Code is modular, documented, and follows PEP 8
- Synthetic data assumptions are documented

Presentation Quality

- Dashboard loads quickly and works on mobile
- README leads with business value, not technical stack
- Live demo link is included and functional
- Screenshots/GIFs demonstrate key features
- Project can be explained in 2 minutes (elevator pitch ready)

— End of Specification —