

1. Core Recommendation Logic

Scoring principle

Every **sub-scenario** and **question option** contributes a **weighted signal**.

Each signal maps to **one or more offerings**.

Scores accumulate per offering.

At the end:

Highest score → Primary Offering

Second highest → Supporting Offering

Third highest → Optional Capability

2. Master Signal → Offering Mapping

This is the **central scoring dictionary** used by the kiosk engine.

Signal Category	Data	AI	AMM	DPDE
Data Platform Modernization	3	1	0	0
Platform Consolidation / Unification	3	1	0	0
Real-Time Analytics	3	2	0	0
AI-Ready Data Foundation	1	3	0	0
Data Governance	2	2	0	0
BI Optimization	2	1	0	1
AI Analytics Foundation	1	3	0	0
Real-Time AI Foundation	1	3	0	0
App Modernization	0	0	3	0
Cloud Optimization	0	0	2	0
Cloud-Native Transformation	0	1	3	2
DevOps Modernization	0	1	3	2
API / Integration Modernization	0	1	3	2
AI Integration (apps or platforms)	1	3	2	2
Dev Platform Modernization	0	0	1	3
Quality Engineering	0	1	0	3
Experience Engineering	0	2	0	3

AI Engineering	1	3	0	2
----------------	---	---	---	---

3. Weight Contribution Model

Step 1: Sub-scenario scoring

Example:

Sub-Scenario	Signal	Weight
Legacy data warehouse	Data Platform Modernization	3
Need real-time analytics	Real-Time Analytics	3

Add to offering scores using mapping table.

Step 2: Question scoring

Each selected option:

Total offering score +=
 (option weight) × (signal → offering weight)

Example:

Option:

Signal: Real-Time Analytics
 Weight: +2

Mapping:

Data +3
 AI +2

Score added:

Data = $2 \times 3 = +6$
 AI = $2 \times 2 = +4$

4. Calculation Algorithm (Pseudo Logic)

Step-by-step engine logic

Initialize:

Data = 0

AI = 0

AMM = 0

DPDE = 0

Step 1: Add sub-scenario signal

For selected sub-scenario:

 get signalPath and weight

 lookup offering multipliers

 offeringScore += weight × multiplier

Step 2: Process each question

For each answer:

 get signalPath and weight

 lookup offering multipliers

 offeringScore += weight × multiplier

Step 3: Rank offerings

Sort scores descending

Primary = highest

Supporting = second

Optional = third (if score > threshold)

5. Example End-to-End Calculation

User path (Data Transformation scenario)

Selections:

Step	Signal	Weight
------	--------	--------

Sub-scenario: Legacy warehouse	Data Platform Modernization	3
Q1: On-prem DW	Data Platform Modernization	3
Q2: Days to insights	Data Platform Modernization	2
Q3: Duplicate pipelines	Platform Consolidation	3
Q4: Unified platform goal	Platform Consolidation	3
Q5: Batch workloads	Data Platform Modernization	3

Step-by-step scoring

Sub-scenario

Signal: Data Platform Modernization (+3)

Mapping:

Data +3

AI +1

Scores:

Data = $3 \times 3 = 9$

AI = $3 \times 1 = 3$

Q1 (+3 DPM)

Data = +9

AI = +3

Totals:

Data = 18

AI = 6

Q2 (+2 DPM)

Data = +6

AI = +2

Totals:

Data = 24

AI = 8

Q3 (+3 Platform Consolidation)

Mapping:

Data +3

AI +1

Scores:

Data = +9

AI = +3

Totals:

Data = 33

AI = 11

Q4 (+3 Platform Consolidation)

Data = +9

AI = +3

Totals:

Data = 42

AI = 14

Q5 (+3 DPM)

Data = +9

AI = +3

Final totals:

Data = 51

AI = 17

AMM = 0

DPDE = 0

Final Recommendation

Rank	Offering	Score
1	Data Transformation	51
2	AI Integration	17
3	AMM	0
4	DPDE	0

Output:

Primary: Data Transformation

Supporting: AI Integration

Optional: None

6. Threshold Rules (Recommended for kiosk)

Rule	Value
Supporting shown if	Score \geq 40% of primary
Optional shown if	Score \geq 25% of primary
Minimum score to display	\geq 10 points

7. Final Kiosk Output Model

At recommendation screen:

Primary Offering:

Highest score

Supporting Offering:

Second highest (if threshold met)

Optional Capability:

Third (if threshold met)