Here is the updated **MSCFT v4.2 Guidebook – Step-by-Step Instructions** draft including your five-step chat setup example and reference to the GitHub screenshot for final documentation. Plain text, no ASCII, no inline citations, as requested:

MSCFT v4.2 Guidebook: Step-by-Step Forecasting Instructions
(For LLM Use and Forecasting Platform Integration)

Quick Setup Example – How to Structure Your LLM Chat

//* Add template (1)

　Note: (you only need to add the Template at the beginning of each new chat)

//* Add question (2)

//* Add parameters (3)
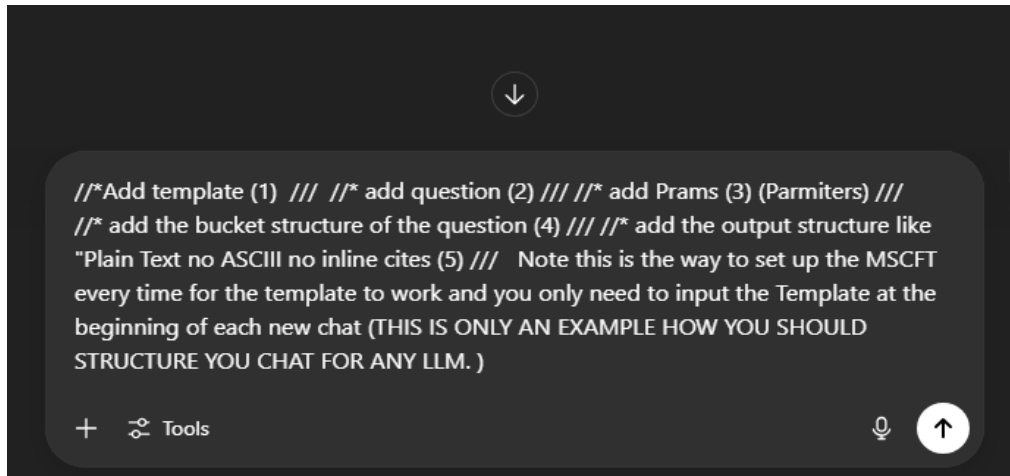
//* Add the bucket structure of the question (4)

//* Add the output structure like: "Plain Text, no ASCII, no inline cites" (5)

Note: This is how to set up MSCFT for any LLM.

You must input the full template at the beginning of each new chat for it to work properly.

This simple five-step sequence ensures that MSCFT logic activates and forecasting nodes execute in structured form. You only need to do this once at the start of a new session.

See Screenshot below :

//*Add template (1) /// //* add question (2) /// //* add Prams (3) (Parmiters) /// //* add the bucket structure of the question (4) /// //* add the output structure like "Plain Text no ASCIII no inline cites (5) /// Note this is the way to set up the MSCFT every time for the template to work and you only need to input the Template at the beginning of each new chat (THIS IS ONLY AN EXAMPLE HOW YOU SHOULD STRUCTURE YOU CHAT FOR ANY LLM.)

This screenshot shows the basic structure of the prompt style you should input into the chat box.

Platform Usage Note: Automation After Initial Setup

On platforms like Metaculus, Good Judgment Open, or RANGE, most of the downstream logic activates automatically once the following three steps are correctly entered:

1. Pose the forecasting question

2. Provide key parameters and clarifications

3. Define the bucket structure (or binary Yes/No) and assign percentage probabilities or the output structure you want to achieve

Note why — Metaculus's use of continuous sliders and distribution graphs necessitates flexible output structuring, which your updated Step 3 now accommodates. This differs from the fixed bucket formats used on GJO, RANGE, or RAND (RFI).

Once complete, the MSCFT engine handles the rest: synthesis, time series modeling, entropy and interpretation, retrieval, and meta-filtering—without further manual editing of nodes.

This is by design. MSCFT separates "human input" (Steps 1–3) from "structured AI reasoning" (Nodes A–G), enabling fast, reproducible forecasting with full auditability.

Full Step-by-Step Forecasting Procedure Using MSCFT v4.2

Step 1: Input the MSCFT Template into the Chat

Copy and paste the entire MSCFT template (v4.2) at the beginning of the session. Do not omit or truncate any nodes. This includes Nodes A through G.6 and all substructures (BIN, RAG, Entropy, Time Series, etc.).

Use the information retrieved to frame your reasoning and support structured forecasting as defined in the previously memorized MSCFT Template 4.0B — SWARM Nodes and BIN Integrated. No improvisation. No format deviation.

Step 2: Add the Parameters of the Question

Under **Node A**, fill in:

  • Forecast Title
• Forecaster handle or name
• Full forecast question
• Clarifications (e.g., dates, entities, thresholds, edge cases)
• Key sources or datasets used (at least 3 preferred)

This ensures the system has enough structural context to initialize Nodes B–G.

Step 3: Define the Bucket or Output Structure

Choose one of the following:

  • Binary structure: Yes / No with percentage for each
• Multi-bucket: Assign percentages across defined bins
• Metaculus-style continuous slider: Include cutpoints or quantiles explicitly
• Custom: Define the output structure you want to achieve (e.g., ranges, medians, thresholds)

Always specify clear numeric percentages. Ambiguous ranges are not allowed.

This step controls the output behavior of Nodes C and D.

Step 4: Let MSCFT Auto-Run Nodes B–G

Once Steps 1–3 are complete, downstream analysis is automatic and node-driven:

  • Node B handles probability estimation, Yes/No logic, and BIN decomposition
• Node C synthesizes final output and forecast summary
• Node D handles uncertainty quantification (Entropy, Markov, KL Divergence)
• Node E applies time series inference (ARIMA, ETS, Fourier, etc.)
• Node F governs external retrieval (if used) and validates content
• Node G.6 filters adversarial/noisy data prior to reasoning steps

You may manually enter content into these nodes, but you are not required to unless override or validation is needed.