

# ChatCRS v0 – Rule-Based Baseline (Detailed Explanation)

## 1. Purpose of v0 Baseline

The **v0 rule-based baseline** was developed as the first step in building a ChatCRS-style conversational recommender system. The primary objective of this version is **architectural validation**, not performance optimization.

Specifically, v0 is designed to:

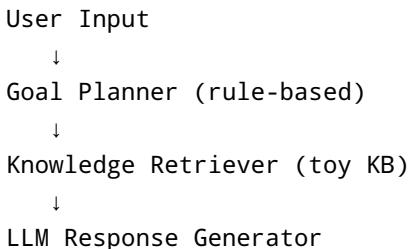
- Validate the end-to-end **pipeline structure** of ChatCRS
- Ensure correct **data flow** between components
- Provide a **fully controllable system** for later robustness analysis
- Serve as a stable reference point for incremental improvements

This version intentionally avoids complex models and large datasets so that each system component can be clearly understood, debugged, and modified.

---

## 2. High-Level System Architecture

The v0 system follows the same **conceptual architecture** as the original ChatCRS, but with simplified internal implementations:



Each module has a **single, well-defined responsibility**, which is critical for both research reproducibility and robustness experimentation.

---

## 3. Component-Level Explanation

### 3.1 Conversational Agent (LLM)

**Role:**

- Generate natural language responses
- Produce recommendations conditioned on external signals

**Implementation in v0:**

- A small instruction-tuned LLM (e.g., Qwen 1.5B Instruct)
- Used strictly as a *response generator*, not a decision-maker

**Key Design Choice:** The LLM is *guided* using structured prompts that include: - Predicted dialogue goal - Retrieved external knowledge

This mirrors the ChatCRS philosophy that LLMs should be **controlled**, not left to reason freely.

---

### 3.2 Goal Planning Module

**Role:** - Predict the system's next conversational action

Examples of dialogue goals: - greeting - ask\_preference - recommend - inform

**Implementation in v0:** - Deterministic, rule-based function - Uses keyword matching on user input

**Why Rule-Based First:** - Makes decision logic transparent - Eliminates training complexity - Allows easy analysis of failure cases

This module is a **placeholder** for later replacement with a learned (LoRA-finetuned) goal predictor.

---

### 3.3 Knowledge Retrieval Module

**Role:** - Supply factual, domain-specific information to the LLM

**Implementation in v0:** - A small, manually defined Python dictionary - Maps genres to movie titles

Example:

```
"sci-fi" → [Interstellar, Inception, Blade Runner 2049]
```

**Why This Matters:** - Separates factual knowledge from the LLM - Enables controlled corruption for robustness testing - Matches the function of the knowledge graph used in the original ChatCRS

This module will later be replaced with a **dataset-derived knowledge base**.

---

### 3.4 System Controller (Pipeline)

**Role:** - Orchestrate all components - Ensure consistent execution order

**Pipeline Steps:** 1. Receive user input 2. Predict dialogue goal 3. Retrieve relevant knowledge 4. Construct a structured prompt 5. Query the LLM 6. Return response and metadata

This controller is the **core of the system** and remains largely unchanged across future versions.

---

## 4. Design Principles Followed in v0

1. **Modularity** - Each component can be replaced independently
2. **Interpretability** - All decisions are explicit and inspectable
3. **Reproducibility** - No stochastic training steps
4. **Extensibility** - Clean hooks for datasets, noise, and learning-based models

These principles are essential for robustness-oriented research.

---

## 5. Limitations of v0 (Intentional)

The v0 baseline has several known limitations: - No real CRS dataset (toy knowledge only) - Rule-based goal prediction - No evaluation metrics - No robustness mitigation strategies

These limitations are **intentional** and define the scope of v0 as an architectural prototype rather than a final system.

---

## 6. Role of v0 in the Overall Research Roadmap

The v0 baseline serves as: - A **reference system** for all future experiments - A sanity check before introducing learning-based complexity - The foundation for robustness studies

Future versions will incrementally add: - Real datasets (DuRecDial / TG-ReDial) - Learned goal planning - Knowledge graph retrieval - Noise injection and robustness evaluation

Each future result will be compared against this v0 baseline to quantify system improvements and degradation.

---

## 7. Summary

The ChatCRS v0 rule-based baseline establishes a clear, functional, and controllable conversational recommender architecture. While simple by design, it faithfully reflects the structural principles of the original ChatCRS and provides a solid foundation for both academic research and practical system development.

This version marks the **starting point** of the project and will be preserved unchanged to ensure experimental traceability and reproducibility.