

# AIU Smart Café Assistant: Database Schema (ERD)

## NoSQL (MongoDB) Database Design

This document outlines the logical Entity-Relationship Diagram (ERD) design for the AIU Smart Café Assistant database. Because the database used is MongoDB Atlas (NoSQL), this design focuses on Collections (equivalent to tables) and uses Embedded Documents for efficiency, rather than using many relational join tables.

This design consists of 6 main Collections.

### 1. USER Collection

Stores all user accounts: students, café staff, and administrators.

Attribute	Data Type	Constraint	Purpose & Justification
<b>_id</b>	ObjectId	Primary Key	Unique MongoDB document identifier.
<b>name</b>	String	Mandatory	Full name of the user.
<b>email</b>	String	Mandatory, Unique	User's login email.
<b>passwordHash</b>	String	Mandatory	Hashed password for security.
<b>role</b>	String	Mandatory	e.g., "student", "staff", "admin". Controls permissions.

<b>dietaryPreferences</b>	Array of Strings	Optional	e.g., ["vegetarian", "nut_allergy"]. Key input for the Recommendation Engine.
<b>createdAt</b>	Timestamp	Mandatory	Automatic timestamp for account creation.

## 2. MENU\_ITEM Collection

The main list of all food and drink items. This is the "product catalog."

Attribute	Data Type	Constraint	Purpose & Justification
<b>_id</b>	ObjectId	Primary Key	Unique item identifier.
<b>name</b>	String	Mandatory, Unique	Name of the dish (e.g., "Nasi Lemak").
<b>description</b>	String	Optional	Brief description of the item.
<b>price</b>	Decimal	Mandatory	Price of the item.
<b>category</b>	String	Mandatory	e.g., "Main Course", "Drink", "Snack".
<b>imageUrl</b>	String	Optional	Link to the item's photo.
<b>isAvailable</b>	Boolean	Mandatory	true / false. Allows staff to change availability in real-time.

<b>nutritionalData</b>	Embedded Document	Optional	e.g., { "calories": 550, "allergens": ["peanuts"] }. Key input for the Recommendation Engine.
<b>lastUpdatedBy</b>	ObjectId (Ref: USER)	Optional	Tracks which admin/staff last modified this item.
<b>lastUpdatedAt</b>	Timestamp	Optional	Tracks when the item was last modified.

### 3. ORDER Collection

Stores every transaction. This collection embeds order items, which is a key NoSQL design choice for efficiency.

Attribute	Data Type	Constraint	Purpose & Justification
<b>_id</b>	ObjectId	Primary Key	Unique order identifier.
<b>userId</b>	ObjectId (Ref: USER)	Mandatory	Links the order back to the user.
<b>createdAt</b>	Timestamp	Mandatory	Automatic timestamp when the order is placed.
<b>totalAmount</b>	Decimal	Mandatory	The final price of the entire order.
<b>status</b>	String	Mandatory	e.g., "Pending", "Preparing", "Ready", "Completed", "Cancelled".

<b>staffId</b>	ObjectId (Ref: USER)	Optional	ID of the staff who prepared the order.
<b>timePrepared</b>	Timestamp	Optional	Timestamp when the status changed to "Preparing".
<b>timeCompleted</b>	Timestamp	Optional	Timestamp when the status changed to "Completed".
<b>items</b>	Array of Embedded Documents	Mandatory	<p>This replaces an ORDER_ITEM table.</p> <p>e.g., [{"itemId": "...", "name": "Nasi Lemak", "quantity": 1, "price": 5.50}]</p>

#### 4. FEEDBACK Collection

Stores all user-submitted reviews. This collection is the primary data source for your NLP sentiment analysis model.

Attribute	Data Type	Constraint	Purpose & Justification
<b>_id</b>	ObjectId	Primary Key	Unique feedback identifier.
<b>userId</b>	ObjectId (Ref: USER)	Mandatory	Links the feedback to a user.
<b>itemId</b>	ObjectId (Ref: MENU_ITEM)	Mandatory	Links the feedback to a specific item.

<b>orderId</b>	ObjectId (Ref: ORDER)	Mandatory	Links to the transaction for context.
<b>rating</b>	Number (1-5)	Mandatory	(As requested) Quantitative star rating.
<b>textReview</b>	String	Optional	(As requested) User's written comments. Key input for the NLP Model.
<b>createdAt</b>	Timestamp	Mandatory	Automatic timestamp when the review is submitted.
<b>sentimentScore</b>	Number	Optional	(ML Output) Precision score (e.g., 0.92) from your Logistic Regression/LSTM model.
<b>sentimentCategory</b>	String	Optional	(ML Output) Simple label (e.g., "Positive", "Negative") derived from the score.

## 5. DEMAND\_DATA Collection

A special collection to store aggregated sales data. This is the main input for your Demand Forecasting (Decision Tree/Random Forest) model.

Attribute	Data Type	Constraint	Purpose & Justification

<b>_id</b>	ObjectId	Primary Key	Unique data point identifier.
<b>date</b>	ISODate	Mandatory	The date of the data (e.g., 2025-11-07T00:00:00Z).
<b>hour</b>	Number (0-23)	Mandatory	The hour of the day. This granularity is important for predicting lunch rush hours.
<b>itemId</b>	ObjectId (Ref: MENU_ITEM)	Mandatory	The item being tracked.
<b>actualDemand</b>	Number	Mandatory	Total number of units sold in that hour. (ML Input)
<b>predictedDemand</b>	Number	Optional	(ML Output) The forecast generated by your model. Stored here for comparison.

## 6. ADMIN\_SETTINGS Collection

A "singleton" collection (will only contain one document) to store global settings for the café, allowing admins to make changes without touching the code.

Attribute	Data Type	Constraint	Purpose & Justification
<b>_id</b>	String	Primary Key	e.g., "global_settings" (a known, fixed ID).

<b>operatingHours</b>	Embedded Document	Mandatory	e.g., { "start": "08:00", "end": "17:00" }
<b>orderCutoffTime</b>	String	Mandatory	e.g., "11:30". The order cutoff time for the day.
<b>forecastModelParams</b>	Embedded Document	Optional	e.g., { "lookbackDays": 30 }. Allows admins to change ML model parameters.
<b>isCafeOpen</b>	Boolean	Mandatory	A main switch to open or close the entire online ordering system.