NON-RELATIONAL DATABASES FOR STORING JSON DATA

Sprint 1: BackLog

Full name: Angel Martín González García

NAO ID: 3313

Date: 08/10/2025

Pathway: IN-MEXICO PROGRAM BACKEND DEVELOPER CERTIFICATION

Backlog

User stories

1. As a diner, I want to search restaurants using filters (TBC) so that I can find relevant options.

2. As a diner, I want to view restaurants within a geographic area (TBC) so that I only see reachable options.

3. As a diner, I want to add comments and ratings so that I can share my experience and evaluate places.

4. As a Tattler content maintainer, I want to register new restaurants so that the directory stays current.

5. As a platform admin, I want a reproducible database backup and CSV import scripts so that the team can restore or reload data quickly.

Tracking Table 1 — Requirements List

#	USER STORY	REQUIREMENT
1	Search with filters (TBC)	Implement a **search endpoint** with **filtering (fields TBC)** and **pagination (TBC)**.
2	View restaurants within an area (TBC)	Implement a **geospatial area filter** (operator and parameters **TBC**) to return restaurants within the specified area.
3	Add comments and ratings	Implement **endpoints to add comments and ratings** and to **retrieve rating/comment data** for a restaurant.
4	Register new restaurants	Implement an **endpoint to create/register restaurants** with required fields and coordinates.
5	Backup + CSV imports	Provide **mongodump-based backup** and **CSV import scripts**, with **instructions** and **screenshots** of successful execution.

Tracking Table 2 — Prioritized List

#	REQUIREMENT	STAGE	TIME ESTIMATION	DELIVERABLES (VERIFIABLE EVIDENCE)
5	Backup + CSV imports	**Sprint 1**	3h	/db/backup/ (mongodump), /db/import/ (CSV + scripts), **README instructions**, **screenshots** of DB/collections (and indexes if applicable).
4	Register restaurants (fields/coords TBC)	**Sprint 2**	3h	Create endpoint implemented; minimal validation (**TBC**); **Postman/Insomnia** example; repo code + screenshots.
3	Comments & ratings	**Sprint 2**	3h	Endpoints to add and retrieve comments/ratings implemented; **tests/screenshots** in Postman/Insomnia; brief notes on data model (**TBC**).
1	Search with filters (fields TBC)	**Sprint 3**	3h	Search endpoint with **filters (TBC)** and **pagination (TBC)**; **tests/screenshots** in Postman/Insomnia.
2	Geospatial area filter (details TBC)	**Sprint 3**	3h	Area-based retrieval implemented per provided MongoDB example (**operator/params TBC**); any necessary geospatial index evidence (screenshot); **tests/screenshots**.