PocketMCP – Phase 1 Ingestion Test (DOCX)

Purpose: Verify Phase 1 ingestion for DOCX and text-based PDF files. This document contains structured headings, lists, and a small table so the extractor can validate layout-to-text conversion.

# 1. Context

PocketMCP is a lightweight, local-first MCP server that watches a folder, chunks content (~1000 chars with 120 overlap), embeds with Transformers.js (Xenova/all-MiniLM-L6-v2), and stores vectors in SQLite + sqlite-vec for semantic search.

# 2. What to Validate

**Extraction goals:**

* Headings and paragraphs are preserved as readable text.
* Bullet lists are flattened into clean lines.
* Table cells are included in plain text output.
* Unicode and punctuation are retained: “quotes”, en–dashes, and emojis 🙂.

# 3. Sample Content

## 3.1 Design Notes

Chunking should not cross page/section boundaries when possible. For DOCX, you may split by H1/H2 (feature-flag) or treat the entire document as one segment. Store segment metadata (e.g., heading) to improve result snippets.

## 3.2 Example Table

|  |  |  |
| --- | --- | --- |
| Feature | Expectation | Notes |
| Headings | Appear as plain text | H1/H2 preferred for splitting |
| Lists | Each item on its own line | No weird bullets |
| Table | Cells concatenated logically | Header row included |

# 4. Acceptance Criteria

1. Ingest status is OK; segments ≥ 1; total chars > 1500.
2. Search for “sqlite-vec” returns a relevant chunk from this document.
3. Search for “MiniLM-L6-v2” returns another chunk from this document.

# Appendix A: Sample Q&A

Q: Why page/section segments?  
A: Better locality and clearer source pointers in search results.

Q: Are scanned PDFs supported?  
A: Not in Phase 1; those should be marked needs\_ocr.