Sequential OCR & Text Extraction SOP (with Page Numbering + Alt Text)

Purpose OCR entire target file chunk by chunk to extract content from a single Target File (PDF, scanned PDF, DOC/DOCX, PPT/PPTX, HTML export, images of text, etc.) in strict sequential order, preserving all details. Output is chunked only to respect chatbot context window/processing smoothness, with ZERO reordering or omission of meaningful content. Page numbering is explicitly included. Features: OCR, image/icon alt text insertion, page number handling, and minimal cleaning (only remove cruft when blatant and obvious).

Files Used in Prompt

Target File

* Prompt the user to upload this file in chat.
* Only this file is processed. No other files are touched unless explicitly named.

Context Files

* Prompt user after target file is uploaded. If user uploads context files or specifies projec tfiles as context, use them to ensure correctness, contextual knowledge, and proper handling of field-specific terms and knowledge.

Page Numbering Rules

* Prefer printed page numbers from the text itself.
* If only odd/even pages are numbered, infer missing pages intelligently.
* If numbering restarts (front matter vs book matter), assign front matter negative numbers: e.g., [Page -3], [Page -2], [Page -1], then first numbered page = [Page 1].
* If no printed page numbers are visible, assign sequential logical numbers starting at 1.
* Each chunk begins with an explicit page range header, e.g.: “Pages 5–7”.
  + //TODO Input instructions about single tripe backtick fencing surrounding the parsed text output of each chunk, if that language isn’t in place already. No other fencing inside that single triple backtick fence–any structure/division/organization should use headings, line breaks, or labels instead of additional fencing. The content which should occur outside of this fencing is status information: 1. list which pages each chunk consists of 2. give status of chunk processing. OK and moving on to next chunk,k or issues encountered/suspected and pause processing to resolve in chat.

Auto-Advance

* After each chunk, proceed immediately to the next sequential chunk. Do not pause for permission. //TODO unless issues suspected, then resolve in chat before returning to regular SOP.
* At the end of the file, append a final line (outside fences): FINISHED: [Target File] fully extracted.

Pre-Processing Pipeline

1. File introspection • Detect: native vs image-only pages, page count, languages, multi-column layouts, footnotes, bibliography sections.
2. OCR (if needed) • Apply OCR to image-only pages; preserve reading order; attempt to detect columns. • Normalize ligatures and hyphenation only when clear; preserve content otherwise. //TODO Edit prompt to make it clear EVERY BIT of the target file must be OCR’d. We’re not relying onany pre-existing text here, we’re ensuring everything is processed form the ground up for best possible results.
3. Image/Icon pre-scan • Identify images, figures, diagrams, icons. • Create a simple symbol map if recurring icons represent meaning (e.g., heart = health in gaming books).
4. Alt text insertion • Inline brief bracketed descriptions: [Image: concise description]. Example [Image: Example of exploding dice being rolled. 1d6 comes up 2, 1d4 rolls 4 then 3, totla is 9.]• If decorative and meaningless, omit alt.

Chunking Workflow

* Process ~500–900 words per chunk for readability, but never break mid-page if avoidable. //TODO Make it so the chatbot decides the optimal chunk size, adjusting as it goes if needed. Balance speed of processing with making sure it never takes bites big enough to choke on or result in accuracy loss or other parsing issues.
* Each chunk includes: • Page range header (based on printed or inferred page numbers). //TODO THis goes outside fencing, make sure that’s clear.// • Sequential text exactly as in source. • Alt text for images/figures inline where they appear.
* Do not reorder or slot content into an external skeleton.

Cross-File Safety

* Process only the declared Target File. Stop if out-of-scope material is detected.

Quality Controls

* Preserve order and completeness above all else.
* Do not condense or reorganize examples.
* Maintain fidelity to source wording.
* Include alt text consistently.

Error Handling

* If OCR fails, insert [OCR issue: description]. //TODO Instead handle ocr issies in chat convo before returning to auto-advance SOP.
* For unclear numbering, explain in Notes.//TODO This is an issue to address in chat as described above before returning to audot advanc SOP

Completion Signal

* After final chunk, outside any fenced block: FINISHED: [Target File] fully extracted.

Kickoff Instruction

1. Confirm Target File name. //TODO Instead prompt for target file, then prompt fror context file upload/identification if user wants contxt files.
2. Run Pre-Processing Pipeline.
3. Immediately emit first sequential chunk with page numbering.
4. Continue automatically until file ends.
5. End with Completion Signal.