Absolutely, we can structure your progress report in a **professional, industry-style format**, showing clear milestones and achievements through Stage 5. Here's a detailed breakdown you can use for documentation, a presentation, or a portfolio:

Search Engine Project – Progress Report

Project Goal

Design and implement a full-text search engine capable of indexing documents, computing relevance scores using BM25, and returning query results with relevant snippets. The project demonstrates **end-to-end understanding** of Information Retrieval (IR) systems, data structures, and ranking algorithms.

Stage 1 – Lexicon Construction

Objective: Build a vocabulary dictionary mapping tokens to IDs and their document frequencies.

Work Completed:

- Processed 5,000 sample documents.
- Constructed **Lexicon** with 46,763 unique tokens.
- Implemented:
    - `LexiconEntry` → stores `term_id`, `term`, and document frequency (`df`).
    - Token-to-ID mapping for fast lookup.
- Verified sample entries successfully.

Outcome:
Lexicon loaded and verified. Tokens are ready for indexing.

Stage 2 – Forward Index

Objective: Build a document-centric index to store term occurrences and positions.

Work Completed:

- Created `ForwardDoc` objects storing:
    - `doc_id`
    - `term_ids`
    - `term frequencies`
    - `positions`
    - `length`
- Constructed `ForwardIndex` with all 5,000 documents.
- Verified document token counts for multiple samples.

Outcome:
Forward index successfully built and verified, allowing rapid access to document-specific term information.

Stage 3 – Inverted Index

Objective: Build a term-centric index for quick retrieval of document lists.

Work Completed:

- Constructed `InvertedIndex` mapping:
    - `term_id → vector<doc_id>`
- Verified postings for sample terms (`colvin`, `keratin`, `sba`, etc.).
- Sample verification confirmed correct document counts for each term.

Outcome:
Inverted index ready for efficient query-based document retrieval.

Stage 4 – BM25 Ranking

Objective: Compute term importance using BM25 ranking for relevance scoring.

Work Completed:

- Implemented `Stage4Ranking`:
    - Compute document lengths (`dl`) and average document length (`avgdl`).
    - Compute **IDF** for all terms.
    - Full BM25 verification across all documents confirmed **non-negative scores**.
- Verified sample term IDFs:
    - `colvin` → IDF=8.112
    - `keratin` → IDF=8.112

Outcome:
Ranking module successfully computes BM25 scores; system can rank documents accurately by query relevance.

Stage 5 – Query Engine

Objective: Provide query functionality returning top-K results with snippets.

Work Completed:

- Implemented `QueryEngine` attaching to lexicon, inverted index, and forward index.
- Tested sample queries: `"lime"`, `"carbon"`, `"ocean"`, `"technology"`.
- Returned:

    - Document IDs
    - BM25 scores
    - Snippets (first 200 characters for context)
- Verified results are consistent with expectations from document content.

Outcome:
Query engine fully functional, supporting relevance-based retrieval with context-aware snippets.

Summary of Progress

Stage	Status	Key Achievements
Stage 1: Lexicon	✅ Complete	46,763 tokens, verified mapping
Stage 2: Forward Index	✅ Complete	5,000 documents indexed, token frequencies and positions stored
Stage 3: Inverted Index	✅ Complete	Term-to-document mapping verified
Stage 4: Ranking (BM25)	✅ Complete	IDFs computed, full BM25 verified
Stage 5: Query Engine	✅ Complete	Top-K query results with snippet retrieval working

Overall Progress: All stages completed to industry standards, demonstrating a **robust end-to-end search engine prototype**. The system is modular, maintainable, and ready for further enhancements such as scalability, interactive queries, or advanced ranking features.

If you want, I can also **visualize this as a polished "professional progress chart"** with stages, verification results, and key metrics – perfect for a report, CV, or LinkedIn showcase.

Do you want me to create that visual summary?