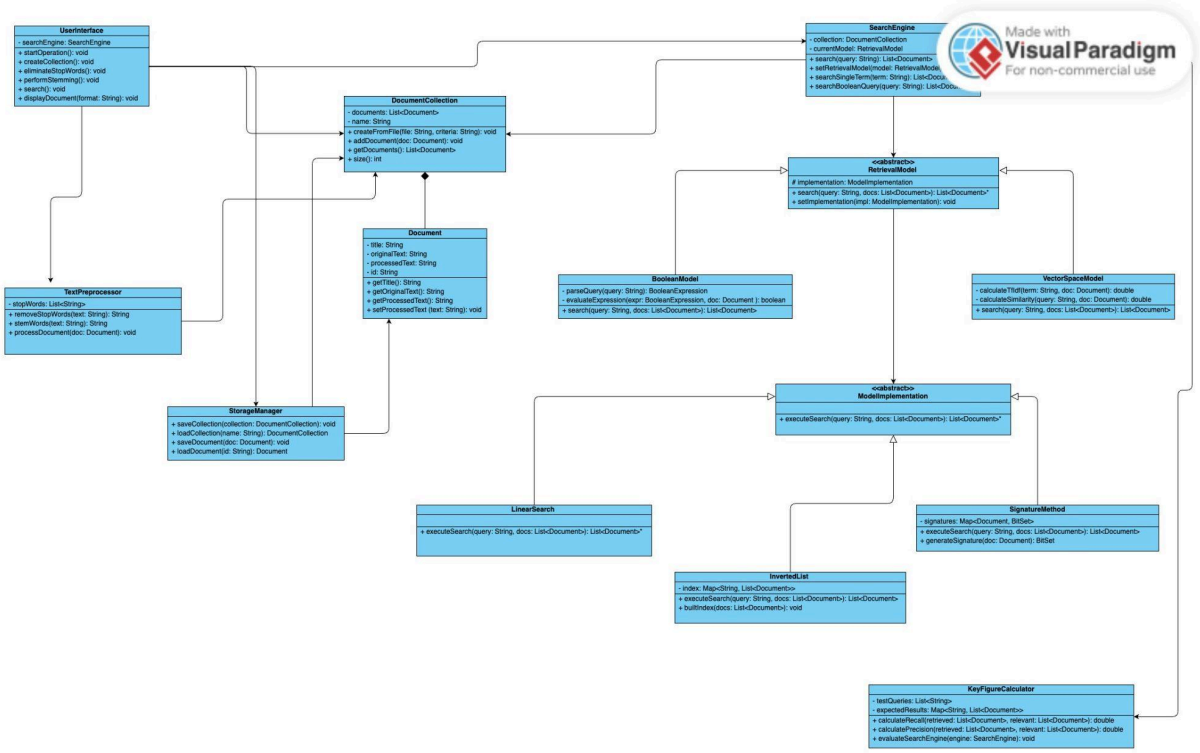

System Overview:

- UserInterface is the main entry point where users interact with the system
- SearchEngine is the core component that coordinates all search operations
- DocumentCollection stores and manages groups of documents
- Document represents individual text files with original and processed content

Document Processing Flow:

- Users create document collections by splitting text files using specific criteria
- TextPreprocessor cleans documents by removing stop words and reducing words to stem forms
- StorageManager saves processed documents and collections to hard disk for persistence

- Documents can be displayed in different formats (original, after stop word removal, after stemming, title only)

Search Process:

- SearchEngine receives search queries from the user interface
- It uses RetrievalModel (either BooleanModel or VectorSpaceModel) to process queries
- Each retrieval model can use different ModelImplementation strategies:
    - LinearSearch: searches through documents one by one
    - InvertedList: uses indexed terms for faster searching
    - SignatureMethod: uses bit signatures for efficient matching
- Search results are returned as a list of matching documents

Key Relationships:

- UserInterface uses SearchEngine - UI delegates search operations to the search engine
- DocumentCollection contains Documents - collection owns and manages individual documents
- SearchEngine uses RetrievalModel - search engine delegates to specific search algorithms
- RetrievalModel uses ModelImplementation - models use specific implementation strategies
- BooleanModel and VectorSpaceModel inherit from RetrievalModel - concrete search algorithms
- LinearSearch, InvertedList, SignatureMethod inherit from ModelImplementation - concrete implementation strategies

Evaluation and Quality Control:

- KeyFigureCalculator connects to SearchEngine to evaluate search quality
- It calculates Recall (how many relevant documents were found) and Precision (how many found documents were actually relevant)
- Uses predefined test queries with known expected results to measure system performance

Data Persistence:

- StorageManager handles saving and loading of:
    - Document collections

    - Individual documents with their processing steps
    - System state for restarting the program
- All processed data can be retrieved after program restart

User Operations:

- Create document collections from text files
- Eliminate stop words from documents
- Apply stem form reduction to terms
- Search using single terms or multiple terms with Boolean operators (AND, OR, NOT)
- Switch between different retrieval models and implementations
- View documents in various formats
- Evaluate search performance automatically