

# General Overview

The system developed for this project utilizes two distinct data models, showcasing the difference between a normalized document store (Task 1) and an embedded document store (Task 2), thereby highlighting the impact of indexing on query performance. Built using Python, the system interacts with MongoDB as its database.

Key Components:

- **MongoDB Database:** Acts as the backend to store information.
  - **Python Scripts:**
    - *task1\_build.py*: Handles the insertion of messages into the messages collection and senders into the senders collection.
    - *task1\_query.py*: Executes queries related to Task 1, such as counting messages with specific criteria and identifying senders with the most messages.
    - *task2\_build.py*: Inserts messages into the messages collection with embedded sender information.
    - *task2\_query.py*: Executes queries related to Task 2, including counting messages with specific criteria and updating sender credit.
- These scripts enable data loading, querying, and seamless interaction with the MongoDB database.

## User Guide(Code execution guide inside readme.md)

This guide provides instructions for setting up and running the MongoDB document store project, covering Task 1 (Normalized Document Store) and Task 2 (Embedded Document Store), using Python and MongoDB.

Prerequisites

Ensure that MongoDB and Python 3.x are installed on your system. To interact with MongoDB from Python, you will need the PyMongo library. Install it using the following command:

**pip3 install pymongo**

Setup MongoDB

### **On Lab Machines:**

- Start the MongoDB Server by creating a directory for MongoDB data files and launching the server with the commands:

- **mkdir ~/mongodb\_data\_folder**

- **mongod --port 27012 --dbpath ~/mongodb\_data\_folder &**

- To interact with MongoDB, open another terminal window and enter:

- **mongosh --port 27012**

### **On MacOS**

- For macOS Users with Intel Processors, Start MongoDB using: ○

**mongod --config /usr/local/etc/mongod.conf --fork**

- For macOS Users with Apple Silicon Processors, Start MongoDB using: ○

**mongod --config /opt/homebrew/etc/mongod.conf --fork**

After starting MongoDB, you can interact with it by typing: **mongosh**  
Running the Scripts

- To run Task 1, Execute the following script: (port number:27012)

- **python3 task1\_build.py portnumber**

- **python3 task1\_query.py portnumber**

- To run Task 2, Execute the following script: : (port number:27012)

- **python3 task2\_build.py portnumber**

○ **python3 task2\_query.py portnumber**

## **Outputs on Lab Machine: >python3**

### **task1\_build.py 27012**

Existing 'messages' collection found. Dropped existing 'messages' collection.

Time taken to insert messages: 12.96 seconds Existing 'senders' collection found.

Dropped existing 'senders' collection. Time taken to insert senders: 0.10 seconds

### **>python3 task1\_query.py 27012**

Q1 Result: 19551 messages have 'ant' in their text.

Time taken for Q1: 0.71 seconds (710.56 milliseconds)

Q2 Result: \*\*\*S.CC has sent the greatest number of messages of 98613. Time taken for Q2: 0.70 seconds (698.56 milliseconds)

Q3 Result: 15354 messages have senders with credit 0.

Time taken for Q3: 0.55 seconds (552.55 milliseconds)

Time taken for Q4: 0.01 seconds (13.55 milliseconds)

Q1 Result: 19551 messages have 'ant' in their text. Time taken for Q1: 0.69 seconds (688.45 milliseconds)

Q2 Result: \*\*\*S.CC has sent the greatest number of messages of 98613.

Time taken for Q2: 0.54 seconds (537.21 milliseconds)

Q3 Result: 15354 messages have senders with credit 0.

Time taken for Q3: 0.02 seconds (18.05 milliseconds)

### **>python3 task2\_build.py 27012**

Time taken to insert messages with embedded sender info: 14.12 seconds

### **>python3 task2\_query.py 27012**

Q1 Result: 19551 messages have 'ant' in their text.

Time taken for Q1: 0.69 seconds (688.62 milliseconds)

Q2 Result: \*\*\*S.CC has sent the greatest number of messages of 98613.

Time taken for Q2: 0.86 seconds (856.52 milliseconds)

Q3 Result: 15354 messages have senders with credit 0.

Time taken for Q3: 0.50 seconds (498.74 milliseconds)  
Time taken for Q4: 2.66 seconds (2658.18 milliseconds)