

CPSC 313 Chat APP (Lab5) Data Model

1. We have 3 main shared collections, then a collection per room.
 - a. The shared room list: name = "main".
 - i. The name is the name of the list in the class
 - ii. The name is also the name of the collection in Mongo
 - iii. The first document in the collection is the room_list metadata
 1. '_id': <autogenerated by Mongo>
 2. 'list_name': 'main'
 3. 'create_time'
 4. 'modify_time'
 5. rooms_metadata = list of room metadata
 - a. Each room metadata has:
 - i. 'room_name': str
 - ii. 'room_type': int
 - iii. 'owner_alias': str
 - iv. 'member_list': dict = {
"alias": <last_message_read: int>,
"alias": <last_message_read: int>
}
 - b. The reason we have this room metadata – **a subset of the metadata stored in the room collection** – is for the constructor of the Room class.
 - i. Since we cannot overload the constructor for a class in Python, we need to think through the constructor
 1. In Python, we approximate overloading by using default parameters in our single constructor
 - ii. The question is "what are the minimal data elements we need to construct an instance of the room class?"
 1. The answer I came up with is the four pieces of data below.
 2. The seeming duplication of room metadata is just to make it much easier to restore a room list and give the room constructor what it needs
 - b. The shared user list: name = 'global'
 - i. The name in the class instance is "global"
 - ii. The name of the collection is "users"
 - iii. There are two document types in the users collection:
 1. A single document containing the metadata for the user list
 - a. '_id': <autogenerated by Mongo>
 - b. 'list_name': '<list_name: str>'
 - c. 'create_time': <value: datetime>

- d. 'modify_time': <value: datetime>
- 2. Each user has a document:
 - a. '_id': <autogenerated by Mongo>
 - b. 'alias': "<alias_name: str>"
 - c. 'blacklist': list = [
 - "<alias_name: str>", "<alias_name: str>"
 - d. 'removed': <value: bool>
 - e. 'create_time': <value: datetime>
 - f. 'modify_time': <value: datetime>
- c. The shared sequence collection
 - i. The name of the collection is "sequence"
 - ii. There is only one document in this collection:
 - 1. '_id': 'userid'
 - 2. Each room has a key:value pair
 - a. Key = <room_name: str>
 - b. Value = <next_sequence_number: int>
- d. Each room has its own collection
 - i. The name of the room (room_name) is the name of the collection
 - ii. The collection has two document types:
 - 1. A single document that is the metadata for the room
 - a. '_id' : <autogenerated by Mongo>
 - b. 'room_name': '<room_name: str>'
 - c. 'owner_alias' : '<owner_alias: str>'
 - d. 'room_type': <value: int>
 - e. 'member_list': dict = {
 - "alias": <last_message_read: int>,
 - "alias": <last_message_read: int>
 - f. 'deleted': <value: bool>
 - g. 'create_time': <value: datetime>
 - h. 'modify_time': <value: datetime>
 - 2. Each message in the room is a document in the collection
 - a. '_id': <autogenerated by Mongo>
 - b. 'message': '<message_text: str>'
 - c. 'sequence_num': <value: int>
 - d. 'removed': <value: bool>
 - e. 'mess_props': dict = {
 - i. 'room_name': '<room_name: str>'
 - ii. 'mess_type': <room_type: int>
 - iii. 'to_user': '<alias: str>'
 - iv. 'from_user': '<alias: str>'

- v. 'sent_time': <value: datetime>
 - 1. The time the message was sent
- vi. 'rec_time': <value: datetime>
 - 1. The time the message was received