

# 1. Broadcast:

## a) send broadcast message

POST localhost:8000/image ws://localhost:8000/ws Overview + ... No Environment

Raw ws://localhost:8000/ws/62211e74de6da79873505623 Save Feedback

ws://localhost:8000/ws/62211e74de6da79873505623 Disconnect

Params Headers Settings

New message Save Message

```
1 {
2   "eventName": "broadcast",
3   "eventPayload": {
4     "fromUserID": "62211e74de6da79873505623",
5     "message": "broadcast test for sprint3"
6   }
7 }
8 }
```

Text Send

Messages CONNECTED

Search All Messages Clear Messages

↓ {"eventName": "message-response", "eventPayload": {"fromUserID": "62211e74de6da79873505623", "toUserID": "", "message": "broadcast te... 21:24:56

↑ {"eventName": "broadcast", "eventPayload": {"fromUserID": "62211e74de6da79873505623", "message": "broadcast test for sprint... 21:24:56

✓ Connected to ws://localhost:8000/ws/62211e74de6da79873505623 21:24:56

Local

5 DBS 9 COLLECTIONS FAVORITE

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 5.0.6 Community

Filter your data

BBChatTest

broadcasts

driftBottles

messages

users

BBchat

room

user

admin

config

local

BBChatTest.broadcasts Documents

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' }

ADD DATA VIEW

\_id: ObjectId("62477c3cfbb53d0a3a2f3533")  
fromUserID: "62476975f2b1f88baeffb354"  
message: "4 March Test"

\_id: ObjectId("62477c78fbb53d0a3a2f3534")  
fromUserID: "62211e74de6da79873505623"  
message: "4 March Broadcast1"

\_id: ObjectId("62477c90fbb53d0a3a2f3535")  
message: "4 March Broadcast2"  
fromUserID: "622261e5f61f9a3ee8b25101"

\_id: ObjectId("6247a5e8e543f23a5e569725")  
fromUserID: "62211e74de6da79873505623"  
message: "broadcast test for sprint3"

## b) get broadcast message

localhost:8000/getBroadcast/62477c78fbb53d0a3a2f3534

GET localhost:8000/getBroadcast/62477c78fbb53d0a3a2f3534 Send

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings Cookies

body Cookies Headers (4) Test Results Status: 200 OK Time: 13 ms Size: 663 B Save Response

Pretty Raw Preview Visualize JSON

```
1 1
2 {"code": 200,
3  "status": "OK",
4  "message": "Username is available.",
5  "response": [
6    {
7      "id": "62477c3cfbb53d0a3a2f3533",
8      "message": "4 March Test",
9      "fromUserID": "62476975f2b1f88baeffb354",
10     "image": ""
11   },
12   {
13     "id": "62477c78fbb53d0a3a2f3534",
14     "message": "4 March Broadcast1",
15     "fromUserID": "62211e74de6da79873505623",
16     "image": ""
17   },
18   {
19     "id": "62477c90fbb53d0a3a2f3535",
20     "message": "4 March Broadcast2",
21     "fromUserID": "622261e5f61f9a3ee8b25101",
22     "image": ""
23   },
24   {
```

## 2. DriftBottle

### a) send driftbottle message to a random user

POST localhost:8000/image ws://localhost:8000/ws Overview +

No Environment

Raw ws://localhost:8000/ws/622261e5f61f9a3ee8b25101 Save Feedback

ws://localhost:8000/ws/622261e5f61f9a3ee8b25101 Disconnect

Params Headers Settings

New message Save Message

```
1 {
2   "eventName": "driftBottle",
3   "eventPayload": {
4     "fromUserID": "622261e5f61f9a3ee8b25101",
5     "message": "test for sprint3",
6     "toUserID": "622261e5f61f9a3ee8b25101"
7   }
8 }
9
```

Text Send

Messages CONNECTED

Search All Messages Clear Messages

- ↓ {"eventName": "driftbottle-response", "eventPayload": {"fromUserID": "622261e5f61f9a3ee8b25101", "toUserID": "622261e5f61f9a3ee8b25101", "message": "test for sprint3", "toUserID": "622261e5f61f9a3ee8b25101"}, "image": ""} 21:16:01
- ↑ {"eventName": "driftBottle", "eventPayload": {"fromUserID": "622261e5f61f9a3ee8b25101", "message": "test for sprint3", "toUserID": "622261e5f61f9a3ee8b25101"}, "image": ""} 21:16:01
- ↓ {"eventName": "image-response", "eventPayload": {"fromUserID": "622261e5f61f9a3ee8b25101", "toUserID": "622261e5f61f9a3ee8b25101", "message": "test for sprint3", "toUserID": "622261e5f61f9a3ee8b25101"}, "image": ""} 21:05:29
- ↑ {"eventName": "image", "eventPayload": {"fromUserID": "622261e5f61f9a3ee8b25101", "image": "4e470d0a1a0a", "toUserID": "622261e5f61f9a3ee8b25101"}, "message": "test for sprint3", "toUserID": "622261e5f61f9a3ee8b25101"}, "image": ""} 21:05:29

BBChatTest.driftBottles

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' }

ADD DATA VIEW

Displaying documents

```

{ "_id": "62477782f2b1f88baeffb356",
  "toUserID": "62476a7af2b1f88baeffb355",
  "fromUserID": "62476975f2b1f88baeffb354",
  "message": "1 April Test",
  "image": "" }

{ "_id": "62477786f2b1f88baeffb357",
  "fromUserID": "62476975f2b1f88baeffb354",
  "message": "1 April Test",
  "toUserID": "62476a7af2b1f88baeffb355",
  "image": "" }

{ "_id": "62477832f2b1f88baeffb358",
  "fromUserID": "62476975f2b1f88baeffb354",
  "message": "1 April Test2",
  "toUserID": "62476a7af2b1f88baeffb355",
  "image": "" }

{ "_id": "6247a3d1e543f23a5e569724",
  "message": "test for sprint3",
  "toUserID": "622261e5f61f9a3ee8b25101",
  "fromUserID": "622261e5f61f9a3ee8b25101",
  "image": "" }

```

- b) user get driftbottle messages
- although here we could see the from and to user ID, but they will not be showed in front end. The ToUserID will be randomly picked up by the front side codes.

localhost:8000/getDriftBottle/62476a7af2b1f88baeffb355/62476975f2b1f88baeffb354

GET localhost:8000/getDriftBottle/62476a7af2b1f88baeffb355/62476975f2b1f88baeffb354

Params Authorization Headers (9) Body Pre-request Script Tests Settings

body Cookies Headers (4) Test Results Status: 200 OK

```

{
  "response": [
    {
      "id": "62477782f2b1f88baeffb356",
      "message": "1 April Test",
      "toUserID": "62476a7af2b1f88baeffb355",
      "fromUserID": "62476975f2b1f88baeffb354",
      "image": ""
    },
    {
      "id": "62477786f2b1f88baeffb357",
      "message": "1 April Test",
      "toUserID": "62476a7af2b1f88baeffb355",
      "fromUserID": "62476975f2b1f88baeffb354",
      "image": ""
    },
    {
      "id": "62477832f2b1f88baeffb358",
      "message": "1 April Test2",
      "toUserID": "62476a7af2b1f88baeffb355",
      "fromUserID": "62476975f2b1f88baeffb354",
      "image": ""
    }
  ]
}

```

### 3. image

frontend send image as binary files and backend stores image in the message collection in binary format.

The screenshot shows a WebSocket client interface. At the top, the URL is `ws://localhost:8000/ws/622261e5f61f9a3ee8b25101`. The interface includes tabs for Params, Headers, and Settings. A "New message" section shows a JSON payload being sent:

```
{  "eventName": "image",  "eventPayload": {    "fromUserID": "622261e5f61f9a3ee8b25101",    "image": "4e470d0a1a0a",    "toUserID": "622261e5f61f9a3ee8b25101"  }}
```

Below the message input, there is a "Send" button. The "Messages" section shows a list of messages. The first message is an "image-response" from the server. The second message is an "image" from the client, which is the payload shown above. The status bar at the bottom indicates "Connected to ws://localhost:8000/ws/622261e5f61f9a3ee8b25101".

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure, with the `BBChatTest` database selected. The `messages` collection is highlighted. The main panel shows the `BBChatTest.messages` collection with 4 documents. The documents are listed in a table view, showing fields like `_id`, `fromUserID`, `toUserID`, `message`, and `image`. A red arrow points to the `image` field in the last document, which contains a binary value.

_id	fromUserID	toUserID	message	image
ObjectId("622272ecf61f9a3ee8b25104")	62211e70de6da79873505622	62211d78de6da79873505621	4 March Test	
ObjectId("622279fbf61f9a3ee8b25105")	62211d78de6da79873505621	62211e70de6da79873505622	4 March Test	
ObjectId("62227a17f61f9a3ee8b25106")	62211d78de6da79873505621	62211e70de6da79873505622	4 March Test	
ObjectId("624784e9fbb53d0a3a2f3536")	622261e5f61f9a3ee8b25101	622261e5f61f9a3ee8b25101	4e470d0a1a0a	
ObjectId("624784e9fbb53d0a3a2f3537")	622261e5f61f9a3ee8b25101	622261e5f61f9a3ee8b25101	4e470d0a1a0a	