

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

**Інститут КНІТ
Кафедра ПЗ**

ЗВІТ

До лабораторної роботи № 6

З дисципліни: *“Програмування в Інтернет”*

На тему: *“Основні типи та функції доступу до БД MongoDB”*

Лектор:

викл. каф. ПЗ
Степанов Д. С.

Виконав:

ст. гр. ПЗ-22
Чаус Олег

Прийняла:

ст. викл. каф. ПЗ
Грицай О. Д.

« ____ » _____ 2023 р.

Σ= _____.

Львів – 2023

Тема роботи: Основні типи та функції доступу до БД MongoDB.

Мета роботи: Засвоїти елементи створення, модифікації, читання та занесення даних з таблиць БД засобами Node.js.

ЗАВДАННЯ

Розробити 2 складові частини проекту :

- Створюємо ДБ тут <https://www.mongodb.com>
- Використати бібліотеку socket.io
- Сайт з формою, що надсилає на сервер дані чату
- Серверна частина перевіряє чи повідомлення конкретному студенту чи декільком/всім студентам і повертає відповідь (response) всім кому це повідомлення адресовано
- Серверна частина має зберегти повідомлення у бд щоб в подальшому показати їх користувачам
- Сайт реагує на відповіді. І в потрібних чатах показує повідомлення

ХІД ВИКОНАННЯ

1. Створена база даних.

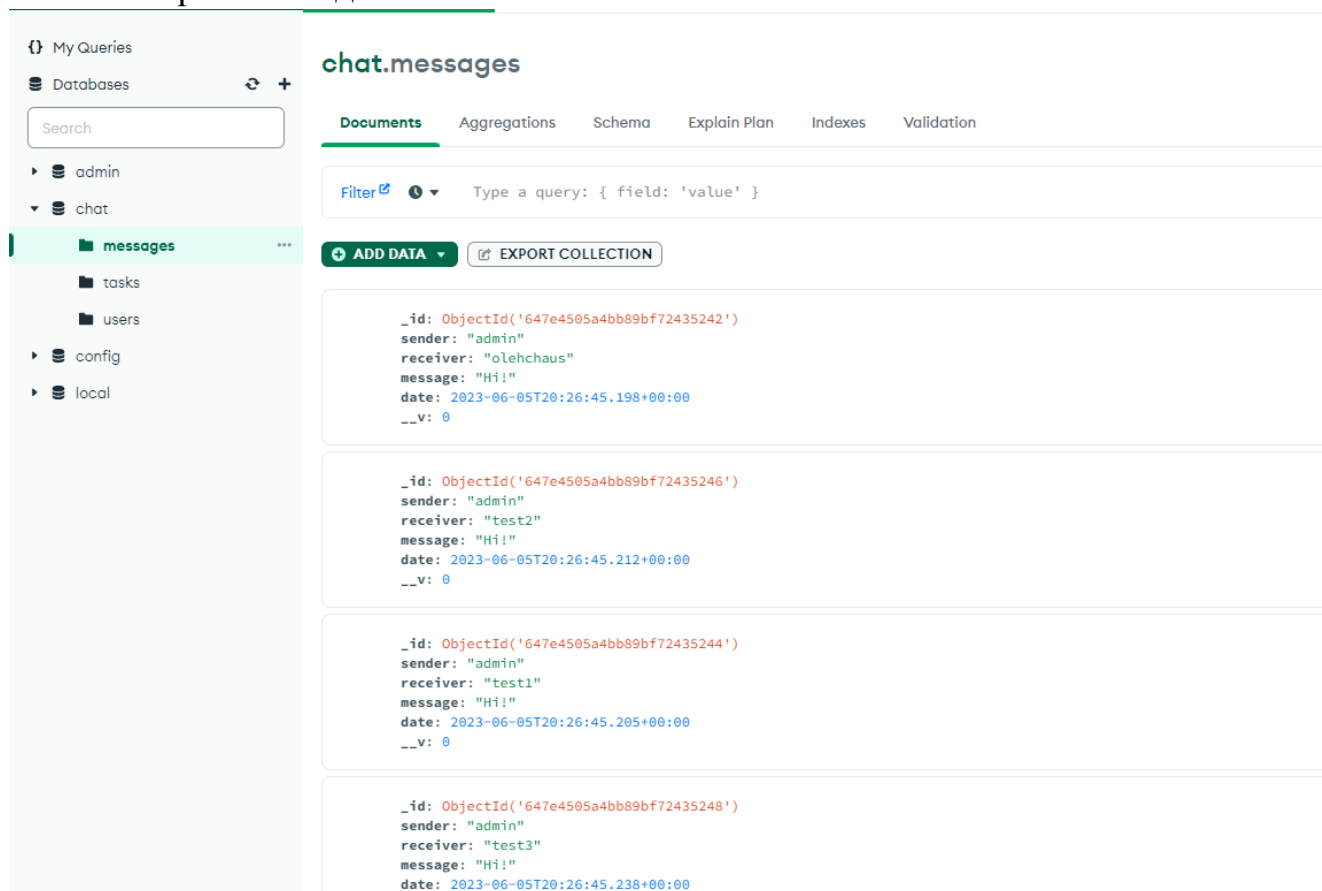


Рис. 1. База даних та колекція *messages*.

2. Створена за допомогою mongoose модель та API.

```
const mongoose = require('mongoose');

const messageSchema = new mongoose.Schema({
  sender: {
    type: String,
    required: true
  },
  receiver: {
    type: String,
    required: false
  },
  message: {
    type: String,
    required: true
  },
  date: {
    type: Date,
    required: false
  },
  global: {
    type: Boolean,
    required: false
  }
});

module.exports = mongoose.model("Message", messageSchema);
```

```
const express = require('express');
const router = express.Router();
const Message = require('../model/messageModel');
module.exports = router;

// POST
router.post('/send', async (req, res) => {
  try {
    if(!req.session.userId) {
      res.status(401).json({message: 'You are not logged in.'});
      return;
    }
    if(req.body.receiver === req.session.username) {
```

```

        res.status(400).json({message: 'You cannot send a message to
yourself.'});
        return;
    }
    const data = new Message({
        sender: req.session.username,
        receiver: req.body.receiver,
        message: req.body.message,
        global: req.body.global,
        date: Date.now()
    });

    const savedData = await data.save();
    res.status(200).json(savedData);
}
catch (error) {
    res.status(400).json({ message: error.message })
}
});

router.get('/fetchMessages', async (req, res) => {
    try {
        if(!req.session.userId) {
            res.status(401).json({message: 'You are not logged in.'});
            return;
        }
        const username = req.session.username;

        const data = await Message.find({
            $or: [
                {sender: username},
                {receiver: username},
                {global: true}
            ]
        }).sort({date: -1});

        //console.log(data);

        res.status(200).json({messages: data, username: username});
    }
    catch (error) {
        res.status(400).json({ message: error.message })
    }
});

```

[Посилання на репозиторій в GitLab](#)

ВИСНОВКИ

Під час виконання лабораторної роботи засвоїв елементи створення, модифікації, читання та занесення даних з таблиць БД засобами Node.js.