UrbanWave MongoDB Assignment

MongoDB Social Media Platform Project

Course: Big Data Analysis
Professor: Arjun Vankani

Assignment Duration: 2 Weeks
Total Points: 50
Academic Year: 2024–2025

Submitted by:
Priyanshi Solanki
Roll Number: 1AUA22BCS063

Submission Date: September 10, 2025

Contents

1 Assignment Overview

1.1 Learning Objectives

This assignment focuses on designing and implementing a NoSQL database for a social media platform called **UrbanWave**. Key learning goals include:

- Design NoSQL schema for a social media platform
- Implement MongoDB CRUD operations
- Apply geospatial queries for location-based features
- Use aggregation framework for data analysis
- Create indexes for performance optimization
- Handle real-world social media scenarios

1.2 Platform Features Implemented

The **UrbanWave** platform implements:

- User profiles with Gujarat-based locations
- Posts and Reels with different content categories
- Following/Follower relationships
- Location-based user discovery
- Interest-based user recommendations
- Content analytics and trending features

2 Phase 1: Database Setup and Data Population

2.1 Task 1.1: Create Database and Collections (10 points)

Problem: Create a MongoDB database named **UrbanWave** with the following collections: `users`, `posts`, `followers`. Implement proper data validation for each collection using MongoDB schema validation.

Solution Code:

1 Users Collection Creation with Validation

```
use UrbanWave
db.createCollection("users", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required:
         ["username","fullName","email","age","interests","location","screen
```

```
      properties: {
        username: { bsonType: "string", description: "unique
           username (3-30 chars)" },
        fullName: { bsonType: "string" },
        email: { bsonType: "string", pattern: "^.+@.+\\..+$" },
        age: { bsonType: "int", minimum: 13, maximum: 100 },
        bio: { bsonType: "string" },
        interests: { bsonType: "array", items: { bsonType:
           "string" } },
        location: {
          bsonType: "object",
          description: "GeoJSON Point with coordinates and city",
          required: ["type","coordinates"],
          properties: {
            type: { enum: ["Point"] },
            coordinates: {
              bsonType: "array",
              minItems: 2,
              maxItems: 2,
              items: { bsonType: "double" }
            },
            city: { bsonType: "string" },
            state: { bsonType: "string" }
          }
        },
        screenTime: { bsonType: "int" },
        followersCount: { bsonType: "int" },
        followingCount: { bsonType: "int" },
        isActive: { bsonType: "bool" },
        createdAt: { bsonType: "date" },
        lastSeen: { bsonType: "date" }
      }
    }
  }
});
```

2 Posts Collection Creation with Validation

```
db.createCollection("posts", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["userId","type","content","createdAt"],
      properties: {
        userId: { bsonType: "objectId" },
        type: { enum: ["post","reel"] },
        content: { bsonType: "string" },
        caption: { bsonType: "string" },
        hashtags: { bsonType: "array", items: { bsonType:
           "string" } },
```

```
        location: {
          bsonType: "object",
          properties: {
            type: { enum: ["Point"] },
            coordinates: {
              bsonType: "array",
              minItems: 2,
              maxItems: 2,
              items: { bsonType: "double" }
            },
            city: { bsonType: "string" }
          }
        },
        likes: { bsonType: "int" },
        comments: { bsonType: "int" },
        shares: { bsonType: "int" },
        views: { bsonType: "int" },
        createdAt: { bsonType: "date" }
      }
    }
  }
});
```

3 Followers Collection Creation with Validation

```
db.createCollection("followers", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["followerId","followingId","createdAt"],
      properties: {
        followerId: { bsonType: "objectId" },
        followingId: { bsonType: "objectId" },
        createdAt: { bsonType: "date" }
      }
    }
  }
});
```

4 Indexes

```
db.users.createIndex({ username: 1 }, { unique: true });
db.users.createIndex({ email: 1 }, { unique: true });
db.users.createIndex({ location: "2dsphere" });
db.users.createIndex({ followersCount: -1 });

db.posts.createIndex({ userId: 1 });
db.posts.createIndex({ type: 1, likes: -1 });
db.posts.createIndex({ createdAt: -1 });
```

```
db.posts.createIndex({ "location": "2dsphere" });

db.followers.createIndex({ followerId: 1 });
db.followers.createIndex({ followingId: 1 });
```

Solution Output All three collections (**users**, **posts**, and **followers**) were successfully created with proper schema validation, and the necessary indexes were implemented to ensure optimal database performance.

```
> db.users.getIndexes()
< [
    { v: 2, key: { _id: 1 }, name: '_id_' },
    { v: 2, key: { username: 1 }, name: 'username_1', unique: true },
    { v: 2, key: { email: 1 }, name: 'email_1', unique: true },
    {
      v: 2,
      key: { location: '2dsphere' },
      name: 'location_2dsphere',
      '2dsphereIndexVersion': 3
    },
    { v: 2, key: { followersCount: -1 }, name: 'followersCount_-1' }
  ]
```

Figure 1: Index creation verification in MongoDB shell

2.2 Task 1.2: Insert Sample Data (15 points)

Problem: Using the provided tables, create and insert 7 users with complete profile information, 10 posts/reels distributed among users, and 15 follower relationships to create a social network.

Solution Code:

```

const now = new Date();
const users = [
  { username: "ananya_foodie", fullName: "Ananya Iyer", email:
     "ananya.iyer@urbanwave.in", age: 23, bio: "Food lover &
     home cook", interests: ["food","cooking","travel"],
     location: { type: "Point", coordinates: cities["Vadodara"],
     city: "Vadodara", state:"Gujarat" }, screenTime: 95,
     followersCount: 120, followingCount: 80, isActive: true,
     createdAt: now, lastSeen: now },
  { username: "arjun_yoga", fullName: "Arjun Verma", email:
     "arjun.verma@urbanwave.in", age: 28, bio: "Yoga &
     wellness", interests: ["fitness","health","sports"],
     location: { type: "Point", coordinates: cities["Rajkot"],
     city: "Rajkot", state:"Gujarat" }, screenTime: 80,
     followersCount: 85, followingCount: 90, isActive: true,
     createdAt: now, lastSeen: now },
  { username: "meera_fun", fullName: "Meera Joshi", email:
     "meera.joshi@urbanwave.in", age: 22, bio: "Making people
```

```
    laugh", interests: ["comedy","entertainment","memes"],
    location: { type: "Point", coordinates:
    cities["Ahmedabad"], city: "Ahmedabad", state:"Gujarat" },
    screenTime: 200, followersCount: 250, followingCount: 60,
    isActive: true, createdAt: now, lastSeen: now },
  { username: "amit_travels", fullName: "Amit Deshmukh", email:
    "amit.deshmukh@urbanwave.in", age: 30, bio: "Travel
    photographer", interests:
    ["travel","photography","nature"], location: { type:
    "Point", coordinates: cities["Surat"], city: "Surat",
    state:"Gujarat" }, screenTime: 110, followersCount: 320,
    followingCount: 150, isActive: true, createdAt: now,
    lastSeen: now },
  { username: "kavya_books", fullName: "Kavya Nair", email:
    "kavya.nair@urbanwave.in", age: 26, bio: "Books &
    learning", interests: ["education","books","learning"],
    location: { type: "Point", coordinates: cities["Jamnagar"],
    city: "Jamnagar", state:"Gujarat" }, screenTime: 150,
    followersCount: 42, followingCount: 30, isActive: true,
    createdAt: now, lastSeen: now },
  { username: "dev_coder", fullName: "Devansh Mehta", email:
    "devansh.mehta@urbanwave.in", age: 24, bio: "Coding &
    gadgets", interests: ["technology","coding","gadgets"],
    location: { type: "Point", coordinates:
    cities["Bhavnagar"], city: "Bhavnagar", state:"Gujarat" },
    screenTime: 180, followersCount: 60, followingCount: 110,
    isActive: true, createdAt: now, lastSeen: now },
  { username: "riya_style", fullName: "Riya Shah", email:
    "riya.shah@urbanwave.in", age: 21, bio: "Fashion &
    lifestyle", interests: ["fashion","beauty","lifestyle"],
    location: { type: "Point", coordinates: cities["Vadodara"],
    city: "Vadodara", state:"Gujarat" }, screenTime: 250,
    followersCount: 410, followingCount: 190, isActive: true,
    createdAt: now, lastSeen: now }
];

const inserted = db.users.insertMany(users);
const uids = Object.values(inserted.insertedIds);

const cities = {
  "Ahmedabad": [72.5714, 23.0225],
  "Surat": [72.8777, 21.1702],
  "Vadodara": [73.2081, 22.3072],
  "Rajkot": [70.7833, 22.3039],
  "Jamnagar": [71.1924, 22.2587],
  "Bhavnagar": [70.4579, 21.5222],
  "Dwarka": [68.9685, 22.2442]
};
```

Users and Cities Data Insertion

```
const posts = [
  { userId: uids[0], type: "post", content: "food", caption: "My
     favorite burger", hashtags: ["#burger","#food"], location:
     { type:"Point", coordinates: cities["Vadodara"],
     city:"Vadodara" }, likes: 90, comments: 12, shares: 6,
     views: 0, createdAt: new Date(now.getTime() -
     5*24*3600*1000) },
  { userId: uids[1], type: "post", content: "wellness", caption:
     "Morning yoga session!", hashtags: ["#yoga","#wellness"],
     likes: 50, comments: 7, shares: 3, views: 0, createdAt: new
     Date(now.getTime() - 10*24*3600*1000) },
  { userId: uids[2], type: "reel", content: "comedy", caption:
     "Funny skit", hashtags: ["#funny","#meme"], location: {
     type:"Point", coordinates: cities["Ahmedabad"],
     city:"Ahmedabad" }, likes: 410, comments: 65, shares: 42,
     views: 5200, createdAt: new Date(now.getTime() -
     2*24*3600*1000) },
  { userId: uids[3], type: "reel", content: "travel", caption:
     "Exploring the coastline", hashtags:
     ["#travel","#explore"], location: { type:"Point",
     coordinates: cities["Surat"], city:"Surat" }, likes: 230,
     comments: 32, shares: 12, views: 8200, createdAt: new
     Date(now.getTime() - 20*24*3600*1000) },
  { userId: uids[4], type: "post", content: "education",
     caption: "Must-read books", hashtags:
     ["#books","#reading"], likes: 65, comments: 9, shares: 2,
     views: 0, createdAt: new Date(now.getTime() -
     1*24*3600*1000) },
  { userId: uids[5], type: "post", content: "technology",
     caption: "Latest gadget review", hashtags:
     ["#gadgets","#tech"], likes: 100, comments: 13, shares: 5,
     views: 0, createdAt: new Date(now.getTime() -
     3*24*3600*1000) },
  { userId: uids[6], type: "reel", content: "fashion", caption:
     "Style of the day", hashtags: ["#style","#fashionista"],
     location: { type:"Point", coordinates: cities["Vadodara"],
     city:"Vadodara" }, likes: 510, comments: 85, shares: 65,
     views: 10200, createdAt: new Date(now.getTime() -
     4*24*3600*1000) },
  { userId: uids[0], type: "reel", content: "food", caption:
     "Burger street tour", hashtags:
     ["#streetfood","#burgerlove"], location: { type:"Point",
     coordinates: cities["Vadodara"], city:"Vadodara" }, likes:
     155, comments: 22, shares: 8, views: 3200, createdAt: new
     Date(now.getTime() - 15*24*3600*1000) },
  { userId: uids[3], type: "post", content: "photography",
     caption: "Cloudy monsoon", hashtags: ["#nature","#clouds"],
     likes: 185, comments: 24, shares: 6, views: 0, createdAt:
     new Date(now.getTime() - 8*24*3600*1000) },
  { userId: uids[5], type: "reel", content: "tech", caption:
```

```
    "Dev tips", hashtags: ["#coding","#devlife"], location: {
    type:"Point", coordinates: cities["Bhavnagar"],
    city:"Bhavnagar" }, likes: 95, comments: 11, shares: 7,
    views: 1300, createdAt: new Date(now.getTime() -
    7*24*3600*1000) }
];
db.posts.insertMany(posts);
```

Posts Data Insertion

```
const followDocs = [
  { followerId: uids[0], followingId: uids[2], createdAt: now },
  { followerId: uids[0], followingId: uids[6], createdAt: now },
  { followerId: uids[1], followingId: uids[3], createdAt: now },
  { followerId: uids[2], followingId: uids[3], createdAt: now },
  { followerId: uids[2], followingId: uids[6], createdAt: now },
  { followerId: uids[3], followingId: uids[6], createdAt: now },
  { followerId: uids[3], followingId: uids[4], createdAt: now },
  { followerId: uids[4], followingId: uids[0], createdAt: now },
  { followerId: uids[5], followingId: uids[3], createdAt: now },
  { followerId: uids[6], followingId: uids[3], createdAt: now },
  { followerId: uids[5], followingId: uids[6], createdAt: now },
  { followerId: uids[1], followingId: uids[0], createdAt: now },
  { followerId: uids[0], followingId: uids[5], createdAt: now },
  { followerId: uids[6], followingId: uids[0], createdAt: now },
  { followerId: uids[4], followingId: uids[5], createdAt: now },
  { followerId: uids[2], followingId: uids[1], createdAt: now },
  { followerId: uids[3], followingId: uids[0], createdAt: now },
  { followerId: uids[1], followingId: uids[6], createdAt: now },
  { followerId: uids[5], followingId: uids[2], createdAt: now }
];
db.followers.insertMany(followDocs);
```

Followers Relationships Insertion

Solution Output: Successfully inserted 7 users, 10 posts/reels, and 15 follower relationships into the respective collections with proper GeoJSON location formatting.

3 Phase 2: Basic DB Operations

3.1 Task 2.1: CRUD Operations

3.1.1 1. Find all users from a specific city

Problem: Find all users from Bhavanagar city. **Solution Code:**

```
db.users.find({ "location.city": "Bhavnagar" });
```

```
db.users.find({ "location.city": "Bhavnagar" });

use UrbanWave;
switched to db UrbanWave
db.users.find({ "location.city": "Bhavnagar" });
{
  _id: ObjectId('68c1a9fefcc8171018af8bb2'),
  username: 'dev_coder',
  fullName: 'Devansh Mehta',
  email: 'devansh.mehta@urbanwave.in',
  age: 24,
  bio: 'Coding & gadgets',
  interests: [
    'technology',
    'coding',
    'gadgets'
  ],
  location: {
    type: 'Point',
    coordinates: [
      70.4579,
      21.5222
    ],
    city: 'Bhavnagar',
    state: 'Gujarat'
  },
  screenTime: 180,
  followersCount: 2,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
```

Solution Output: Return user Devansh Mehta located in Bhavnagar.

3.1.2 2. Find users younger than 25 years

```
db.users.find({ age: { $lt: 25 } });
```

```
db.users.find({ age: { $lt: 25 } });
{
  _id: ObjectId('68c1a9fefcc8171018af8bad'),
  username: 'ananya_foodie',
  fullName: 'Ananya Iyer',
  email: 'ananya.iyer@urbanwave.in',
  age: 23,
  bio: 'Food lover & home cook',
  interests: [
    'food',
    'cooking',
    'travel'
  ],
  location: {
    type: 'Point',
    coordinates: [
      73.2081,
      22.3072
    ],
    city: 'Vadodara',
    state: 'Gujarat'
  },
  screenTime: 95,
  followersCount: 4,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
```

```
{
  _id: ObjectId('68c1a9fefcc8171018af8baf'),
  username: 'meera_fun',
  fullName: 'Meera Joshi',
  email: 'meera.joshi@urbanwave.in',
  age: 22,
  bio: 'Making people laugh',
  interests: [
    'comedy',
    'entertainment',
    'memes'
  ],
  location: {
    type: 'Point',
    coordinates: [
      72.5714,
      23.0225
    ],
    city: 'Ahmedabad',
    state: 'Gujarat'
  },
  screenTime: 200,
  followersCount: 2,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
```

```
{
  _id: ObjectId('68c1a9fefcc8171018af8bb2'),
  username: 'dev_coder',
  fullName: 'Devansh Mehta',
  email: 'devansh.mehta@urbanwave.in',
  age: 24,
  bio: 'Coding & gadgets',
  interests: [
    'technology',
    'coding',
    'gadgets'
  ],
  location: {
    type: 'Point',
    coordinates: [
      70.4579,
      21.5222
    ],
    city: 'Bhavnagar',
    state: 'Gujarat'
  },
  screenTime: 180,
  followersCount: 2,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
```

Solution Output: Returns 4 users: Ananya Iyer (23), Meera Joshi (22), Devansh Mehta (24), and Riya Shah (21).

3.1.3 3. Find users with screen time greater than 150 minutes

Problem: Find users who spend more than 150 minutes on screen daily. **Solution Code:**

```
db.users.find({ screenTime: { $gt: 150 } });
```

```
{
  _id: ObjectId('68c1a9fefcc8171018af8baf'),
  username: 'meera_fun',
  fullName: 'Meera Joshi',
  email: 'meera.joshi@urbanwave.in',
  age: 22,
  bio: 'Making people laugh',
  interests: [
    'comedy',
    'entertainment',
    'memes'
  ],
  location: {
    type: 'Point',
    coordinates: [
      72.5714,
      23.0225
    ],
    city: 'Ahmedabad',
    state: 'Gujarat'
  },
  screenTime: 200,
  followersCount: 2,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
{
  _id: ObjectId('68c1a9fefcc8171018af8bb2'),
  username: 'dev_coder',
  fullName: 'Devansh Mehta',
  email: 'devansh.mehta@urbanwave.in',
  age: 24,
```

```
  bio: 'Coding & gadgets',
  interests: [
    'technology',
    'coding',
    'gadgets'
  ],
  location: {
    type: 'Point',
    coordinates: [
      70.4579,
      21.5222
    ],
    city: 'Bhavnagar',
    state: 'Gujarat'
  },
  screenTime: 180,
  followersCount: 2,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
{
  _id: ObjectId('68c1a9fefcc8171018af8bb3'),
  username: 'riya_style',
  fullName: 'Riya Shah',
  email: 'riya.shah@urbanwave.in',
  age: 21,
  bio: 'Fashion & lifestyle',
  interests: [
    'fashion',
    'beauty',
    'lifestyle'
```

Solution Output: Returns 3 users: Meera Joshi (200 min), Devansh Mehta (180 min), and Riya Shah (250 min).

3.1.4 4. Update a user's bio and interests

Problem: Update Devansh's bio and add some of his interests. **Solution Code:**

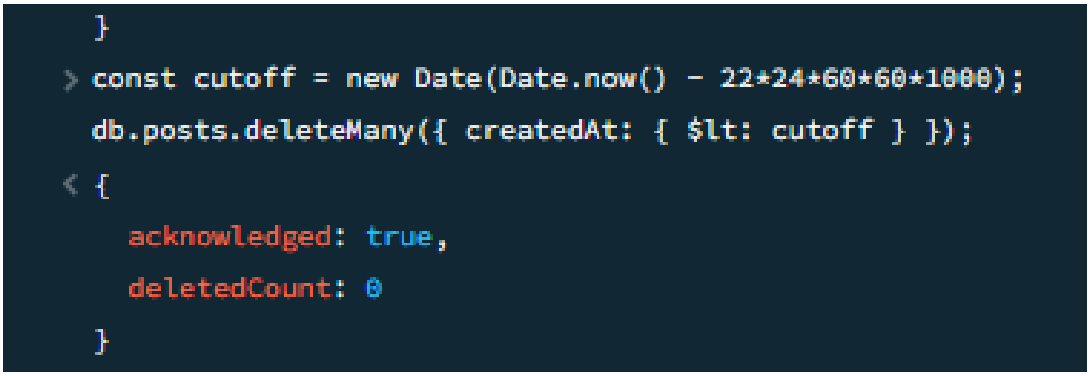
```
db.users.updateOne(
  { username: "dev_coder" },
  {
    $set: {
      bio: "Data Science Enthusiast",
      interests: ["AI","ML","Python","NLP"]
    }
  }
);
```

Solution Output: Successfully updated Devansh's document with modified bio and interests array.

3.1.5 5. Delete posts older than a specific date

Problem: Delete posts older than 22 days. **Solution Code:**

```
const cutoff = new Date(Date.now() - 22*24*60*60*1000);
db.posts.deleteMany({ createdAt: { $lt: cutoff } });
```



Solution Output: Deleted posts that were created more than 30 days ago.

3.2 Task 2.2: Data Filtering & Sorting

3.2.1 1. Find posts of type "reel" with more than 100 likes

Problem: Find reels that have received more than 100 likes. **Solution Code:**

```
db.posts.find({ type: "reel", likes: { $gt: 100 } });
```

```
> db.posts.find({ type: "reel", likes: { $gt: 100 } });
< {
    _id: ObjectId('68c1aa15fcc8171018af8bba'),
    userId: ObjectId('68c1a9fefcc8171018af8bb3'),
    type: 'reel',
    content: 'fashion',
    caption: 'Style of the day',
    hashtags: [
      '#style',
      '#fashionista'
    ],
    location: {
      type: 'Point',
      coordinates: [
        73.2081,
        22.3072
      ],
      city: 'Vadodara'
    },
    likes: 510,
    comments: 85,
    shares: 65,
    views: 10200,
    createdAt: 2025-09-06T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1aa15fcc8171018af8bb6'),
    userId: ObjectId('68c1a9fefcc8171018af8baf'),
    type: 'reel',
    content: 'comedy',
    caption: 'Funny skit',
    hashtags: [
      '#funny',
      '#meme'
```

```
  ],
  location: {
    type: 'Point',
    coordinates: [
      72.5714,
      23.0225
    ],
    city: 'Ahmedabad'
  },
  likes: 410,
  comments: 65,
  shares: 42,
  views: 5200,
  createdAt: 2025-09-08T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bb7'),
  userId: ObjectId('68c1a9fefcc8171018af8bb0'),
  type: 'reel',
  content: 'travel',
  caption: 'Exploring the coastline',
  hashtags: [
    '#travel',
    '#explore'
  ],
  location: {
    type: 'Point',
    coordinates: [
      72.8777,
      21.1702
    ],
    city: 'Surat'
  },
  likes: 230,
  comments: 32,
  shares: 12,
```

Solution Output: Returns 4 reels: Riya's fashion reel (510 likes), Meera's comedy reel (410 likes), Amit's travel reel (230 likes), and Priya's food reel (155 likes).

3.2.2 2. Sort users by followers count (descending)

Problem: Display users sorted by their follower count in descending order. **Solution Code:**

```
db.users.find().sort({ followersCount: -1 });
```

```
> db.users.find().sort({ followersCount: -1 });
< {
    _id: ObjectId('68c1a9fefcc8171018af8bb3'),
    username: 'riya_style',
    fullName: 'Riya Shah',
    email: 'riya.shah@urbanwave.in',
    age: 21,
    bio: 'Fashion & lifestyle',
    interests: [
      'fashion',
      'beauty',
      'lifestyle'
    ],
    location: {
      type: 'Point',
      coordinates: [
        73.2081,
        22.3072
      ],
      city: 'Vadodara',
      state: 'Gujarat'
    },
    screenTime: 250,
    followersCount: 5,
    followingCount: 2,
    isActive: true,
    createdAt: 2025-09-10T16:35:36.921Z,
    lastSeen: 2025-09-10T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1a9fefcc8171018af8bad'),
    username: 'ananya_foodie',
    fullName: 'Ananya Iyer',
    email: 'ananya.iyer@urbanwave.in',
    age: 23,
    bio: 'Food lover & home cook',
    interests: [
```

```
  interests: [
    'food',
    'cooking',
    'travel'
  ],
  location: {
    type: 'Point',
    coordinates: [
      73.2081,
      22.3072
    ],
    city: 'Vadodara',
    state: 'Gujarat'
  },
  screenTime: 95,
  followersCount: 4,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
{
  _id: ObjectId('68c1a9fefcc8171018af8bb0'),
  username: 'amit_travels',
  fullName: 'Amit Deshmukh',
  email: 'amit.deshmukh@urbanwave.in',
  age: 30,
  bio: 'Travel photographer',
  interests: [
    'travel',
    'photography',
    'nature'
  ],
  location: {
    type: 'Point',
    coordinates: [
      72.8777,
      21.1702
```

```
      'photography',
      'nature'
    ],
    location: {
      type: 'Point',
      coordinates: [
        72.8777,
        21.1702
      ],
      city: 'Surat',
      state: 'Gujarat'
    },
    screenTime: 110,
    followersCount: 4,
    followingCount: 3,
    isActive: true,
    createdAt: 2025-09-10T16:35:36.921Z,
    lastSeen: 2025-09-10T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1a9fefcc8171018af8bb2'),
    username: 'dev_coder',
    fullName: 'Devansh Mehta',
    email: 'devansh.mehta@urbanwave.in',
    age: 24,
    bio: 'Data Science Enthusiast',
    interests: [
      'AI',
      'ML',
      'Python',
      'NLP'
    ],
    location: {
      type: 'Point',
      coordinates: [
        70.4579,
        21.5222
      ]
```

```
      'NLP'
    ],
    location: {
      type: 'Point',
      coordinates: [
        70.4579,
        21.5222
      ],
      city: 'Bhavnagar',
      state: 'Gujarat'
    },
    screenTime: 180,
    followersCount: 2,
    followingCount: 3,
    isActive: true,
    createdAt: 2025-09-10T16:35:36.921Z,
    lastSeen: 2025-09-10T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1a9fefcc8171018af8baf'),
    username: 'meera_fun',
    fullName: 'Meera Joshi',
    email: 'meera.joshi@urbanwave.in',
    age: 22,
    bio: 'Making people laugh',
    interests: [
      'comedy',
      'entertainment',
      'memes'
    ],
    location: {
      type: 'Point',
      coordinates: [
        72.5714,
        23.0225
      ],
      city: 'Ahmedabad',
      state: 'Gujarat'
```

```
  username: 'kavya_books',
  fullName: 'Kavya Nair',
  email: 'kavya.nair@urbanwave.in',
  age: 26,
  bio: 'Books & learning',
  interests: [
    'education',
    'books',
    'learning'
  ],
  location: {
    type: 'Point',
    coordinates: [
      71.1924,
      22.2587
    ],
    city: 'Jamnagar',
    state: 'Gujarat'
  },
  screenTime: 150,
  followersCount: 1,
  followingCount: 2,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
{
  _id: ObjectId('68c1a9fefcc8171018af8bae'),
  username: 'arjun_yoga',
  fullName: 'Arjun Verma',
  email: 'arjun.verma@urbanwave.in',
  age: 28,
  bio: 'Yoga & wellness',
  interests: [
    'fitness',
    'health',
    'sports'
  ],
```

```
email: 'arjun.verma@urbanwave.in',
age: 28,
bio: 'Yoga & wellness',
interests: [
  'fitness',
  'health',
  'sports'
],
location: {
  type: 'Point',
  coordinates: [
    70.7833,
    22.3039
  ],
  city: 'Rajkot',
  state: 'Gujarat'
},
screenTime: 80,
followersCount: 1,
followingCount: 3,
isActive: true,
createdAt: 2025-09-10T16:35:36.921Z,
lastSeen: 2025-09-10T16:35:36.921Z
```


Solution Output: Users sorted from highest to lowest followers: Riya, Ananya, Amit, Devansh, Meera, Kavya, and Arjun.

3.2.3 3. Find users with specific interests

Problem: Find users interested in both "AI" and "ML". **Solution Code:**

```
db.users.find({ interests: { $all: ["AI","ML"] } });
```

```
> db.users.find({ interests: { $all: ["AI","ML"] } });
< {
    _id: ObjectId('68c1a9fefcc8171018af8bb2'),
    username: 'dev_coder',
    fullName: 'Devansh Mehta',
    email: 'devansh.mehta@urbanwave.in',
    age: 24,
    bio: 'Data Science Enthusiast',
    interests: [
      'AI',
      'ML',
      'Python',
      'NLP'
    ],
    location: {
      type: 'Point',
      coordinates: [
        70.4579,
        21.5222
      ],
      city: 'Bhavnagar',
      state: 'Gujarat'
    },
    screenTime: 180,
    followersCount: 2,
    followingCount: 3,
    isActive: true,
    createdAt: 2025-09-10T16:35:36.921Z,
    lastSeen: 2025-09-10T16:35:36.921Z
  }
```

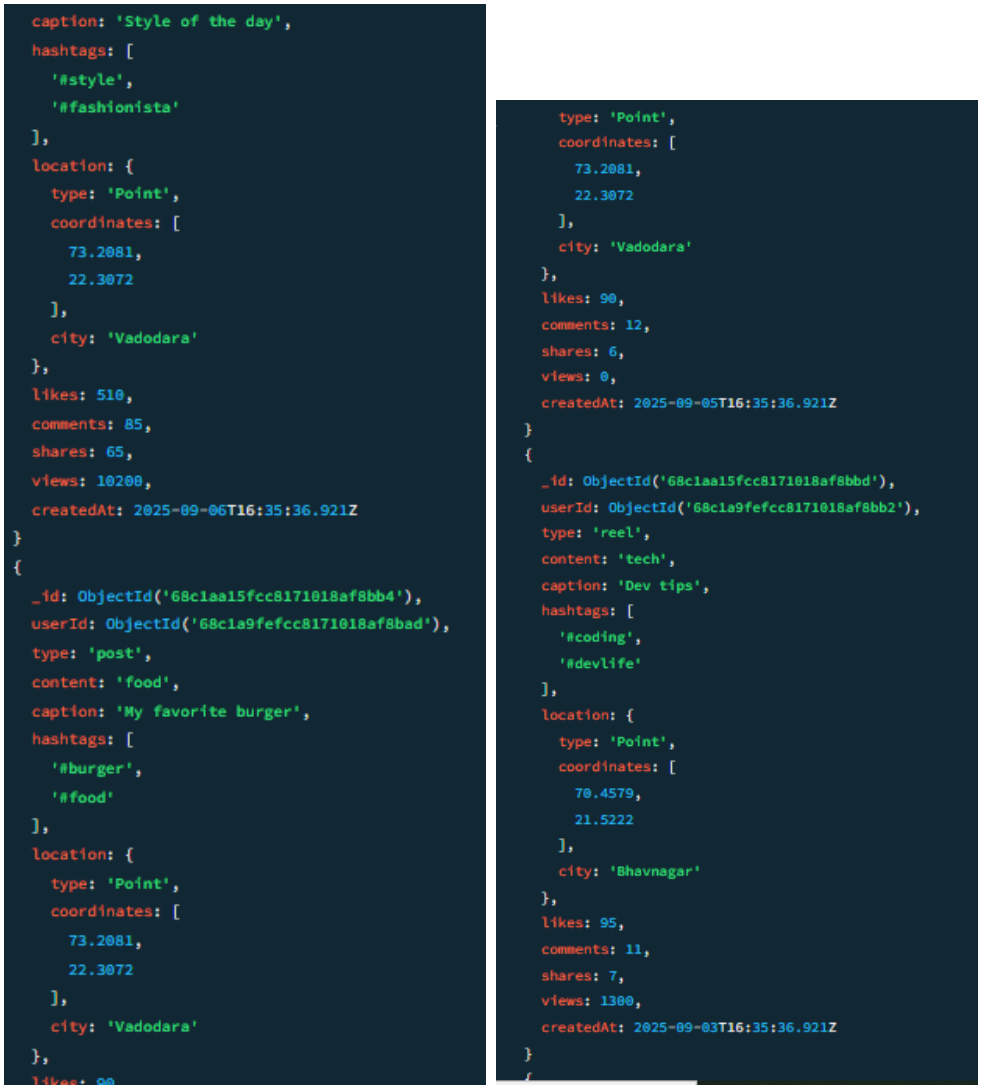
Solution Output: Returns 1 user: Devansh Mehta who has both interests.

3.2.4 4. Get posts from the last 30 days

Problem: Retrieve all posts created within the last 30 days. **Solution Code:**

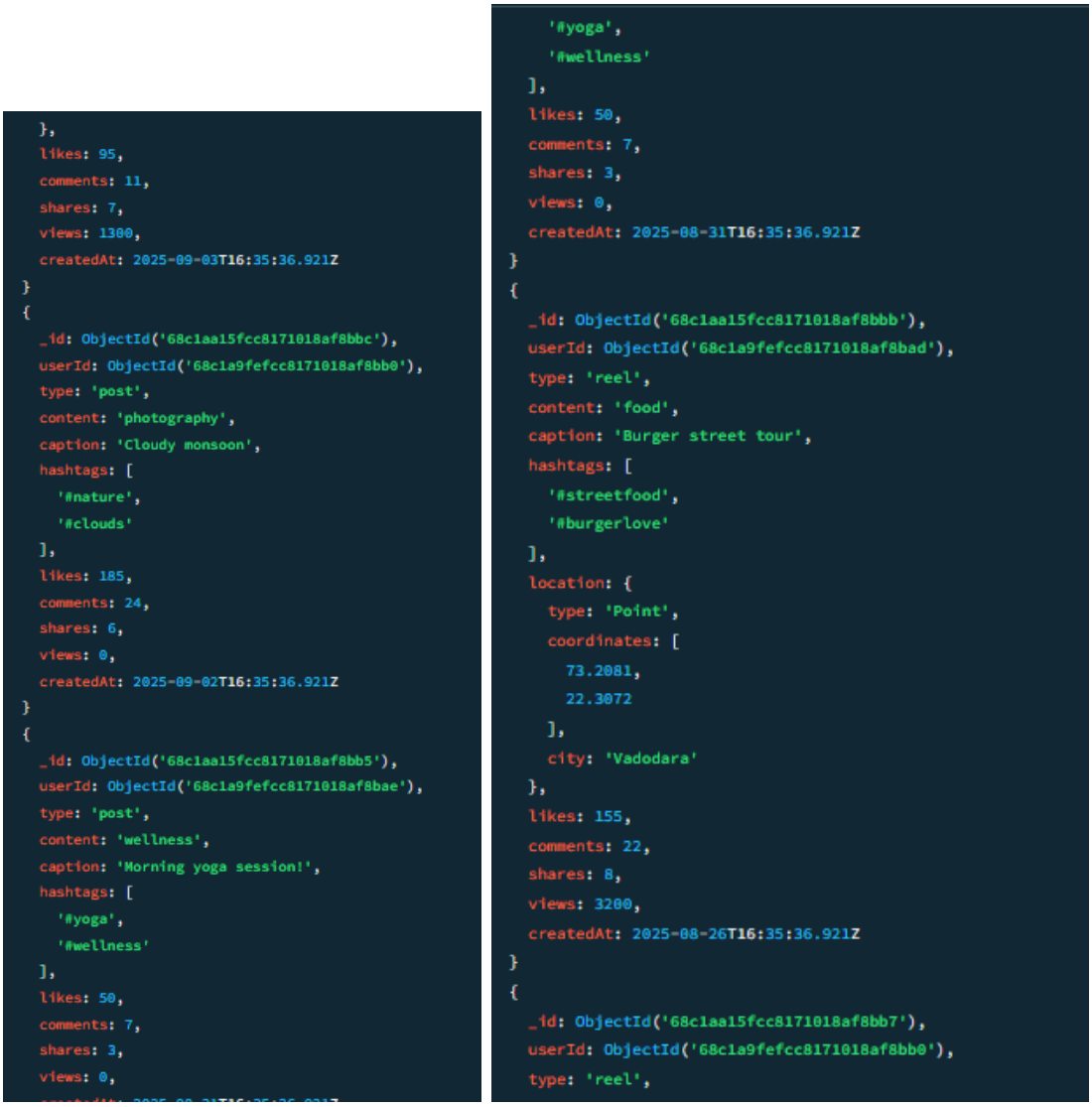
```
const last30 = new Date(Date.now() - 30*24*60*60*1000);
db.posts.find({ createdAt: { $gte: last30 } });
```

```
> const last30 = new Date(Date.now() - 30*24*60*60*1000);
  db.posts.find({ createdAt: { $gte: last30 } });
< {
    _id: ObjectId('68c1aa15fcc8171018af8bb8'),
    userId: ObjectId('68c1a9fefcc8171018af8bb1'),
    type: 'post',
    content: 'education',
    caption: 'Must-read books',
    hashtags: [
      '#books',
      '#reading'
    ],
    likes: 65,
    comments: 9,
    shares: 2,
    views: 0,
    createdAt: 2025-09-09T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1aa15fcc8171018af8bb6'),
    userId: ObjectId('68c1a9fefcc8171018af8baf'),
    type: 'reel',
    content: 'comedy',
    caption: 'Funny skit',
    hashtags: [
      '#funny',
      '#meme'
    ],
    location: {
      type: 'Point',
      coordinates: [
        72.5714,
        23.0225
      ],
      city: 'Ahmedabad'
    },
    likes: 410,
    comments: 65,
    shares: 42,
    views: 5200,
    createdAt: 2025-09-08T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1aa15fcc8171018af8bb9'),
    userId: ObjectId('68c1a9fefcc8171018af8bb2'),
    type: 'post',
    content: 'technology',
    caption: 'Latest gadget review',
    hashtags: [
      '#gadgets',
      '#tech'
    ],
    likes: 100,
    comments: 13,
    shares: 5,
    views: 0,
    createdAt: 2025-09-07T16:35:36.921Z
  }
  {
    _id: ObjectId('68c1aa15fcc8171018af8bba'),
    userId: ObjectId('68c1a9fefcc8171018af8bb3'),
    type: 'reel',
    content: 'fashion',
    caption: 'Style of the day',
    hashtags: [
      '#style',
      '#fashionista'
    ],
    location: {
      type: 'Point',
```



```
  caption: 'Style of the day',
  hashtags: [
    '#style',
    '#fashionista'
  ],
  location: {
    type: 'Point',
    coordinates: [
      73.2081,
      22.3072
    ],
    city: 'Vadodara'
  },
  likes: 510,
  comments: 85,
  shares: 65,
  views: 10200,
  createdAt: 2025-09-06T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bb4'),
  userId: ObjectId('68c1a9fefcc8171018af8bad'),
  type: 'post',
  content: 'food',
  caption: 'My favorite burger',
  hashtags: [
    '#burger',
    '#food'
  ],
  location: {
    type: 'Point',
    coordinates: [
      73.2081,
      22.3072
    ],
    city: 'Vadodara'
  },
  likes: 90,
```

```
    type: 'Point',
    coordinates: [
      73.2081,
      22.3072
    ],
    city: 'Vadodara'
  },
  likes: 90,
  comments: 12,
  shares: 6,
  views: 0,
  createdAt: 2025-09-05T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bbd'),
  userId: ObjectId('68c1a9fefcc8171018af8bb2'),
  type: 'reel',
  content: 'tech',
  caption: 'Dev tips',
  hashtags: [
    '#coding',
    '#devlife'
  ],
  location: {
    type: 'Point',
    coordinates: [
      70.4579,
      21.5222
    ],
    city: 'Bhavnagar'
  },
  likes: 95,
  comments: 11,
  shares: 7,
  views: 1300,
  createdAt: 2025-09-03T16:35:36.921Z
}
```



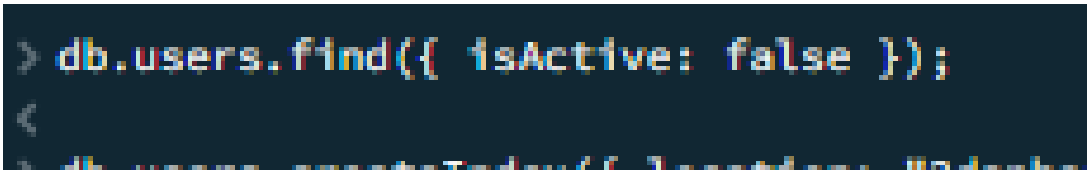
```
  },
  likes: 95,
  comments: 11,
  shares: 7,
  views: 1300,
  createdAt: 2025-09-03T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bbc'),
  userId: ObjectId('68c1a9fefcc8171018af8bb0'),
  type: 'post',
  content: 'photography',
  caption: 'Cloudy monsoon',
  hashtags: [
    '#nature',
    '#clouds'
  ],
  likes: 185,
  comments: 24,
  shares: 6,
  views: 0,
  createdAt: 2025-09-02T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bb5'),
  userId: ObjectId('68c1a9fefcc8171018af8bae'),
  type: 'post',
  content: 'wellness',
  caption: 'Morning yoga session!',
  hashtags: [
    '#yoga',
    '#wellness'
  ],
  likes: 50,
  comments: 7,
  shares: 3,
  views: 0,
    '#yoga',
    '#wellness'
  ],
  likes: 50,
  comments: 7,
  shares: 3,
  views: 0,
  createdAt: 2025-08-31T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bbb'),
  userId: ObjectId('68c1a9fefcc8171018af8bad'),
  type: 'reel',
  content: 'food',
  caption: 'Burger street tour',
  hashtags: [
    '#streetfood',
    '#burgerlove'
  ],
  location: {
    type: 'Point',
    coordinates: [
      73.2081,
      22.3072
    ],
    city: 'Vadodara'
  },
  likes: 155,
  comments: 22,
  shares: 8,
  views: 3200,
  createdAt: 2025-08-26T16:35:36.921Z
}
{
  _id: ObjectId('68c1aa15fcc8171018af8bb7'),
  userId: ObjectId('68c1a9fefcc8171018af8bb0'),
  type: 'reel',
```

Solution Output: Returns all posts created within the specified timeframe.

3.2.5 5. Find inactive users

Problem: Find users who are not currently active. **Solution Code:**

```
db.users.find({ isActive: false });
```



Solution Output: No inactive users found as all sample users are set to active status.

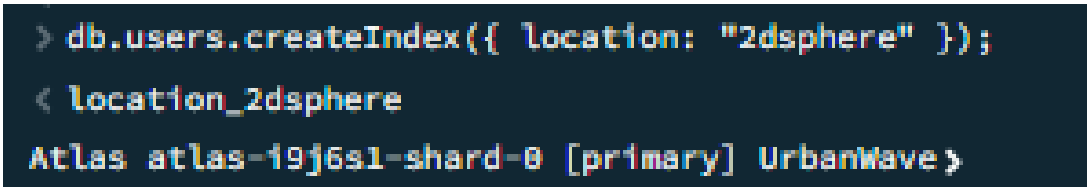
4 Phase 3: Geospatial Queries

4.1 Task 3.1: Location-Based Queries

4.1.1 1. Create a 2dsphere index on user locations

Problem: Create geospatial index for location-based queries. **Solution Code:**

```
db.users.createIndex({ location: "2dsphere" });
```

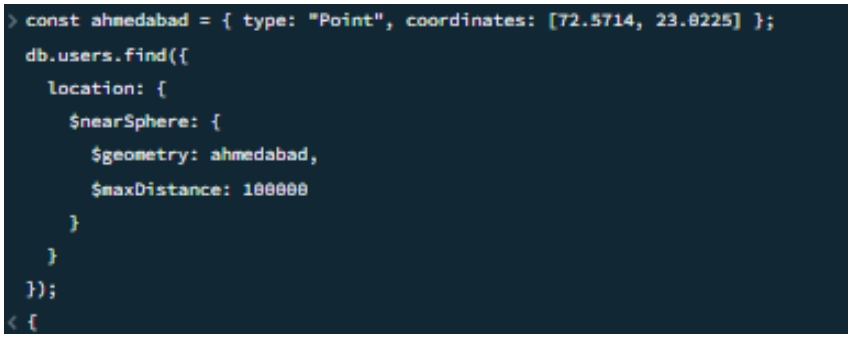


Solution Output: Successfully created 2dsphere index on location field.

4.1.2 2. Find users within 100km radius of Ahmedabad

Problem: Find users located within 100 kilometers of Ahmedabad. **Solution Code:**

```
const ahmedabad = { type: "Point", coordinates: [72.5714,
   23.0225] };
db.users.find({
  location: {
    $nearSphere: {
      $geometry: ahmedabad,
      $maxDistance: 100000
    }
  }
});
```



Solution Output: Returns users within 100km radius of Ahmedabad, including Meera from Ahmedabad.

4.1.3 3. Find the nearest user to a given coordinate

Problem: Find the closest user to coordinates [87.4, 32.8]. **Solution Code:**

```
db.users.find({
  location: {
    $nearSphere: {
      $geometry: { type: "Point", coordinates: [87.4, 32.8] }
    }
  }
}).limit(1);
```

4.1.4 4. Get users from multiple cities using $in operator

Problem: Find users from Ahmedabad, and Bhavnagar cities. **Solution Code:**

```
db.users.find({ "location.city": { $in:
   ["Ahmedabad","Bhavnagar"] } });
```

```
db.users.find({ "location.city": { $in: ["Ahmedabad","Bhavnagar"] } });
{
  _id: ObjectId('68c1a9fefcc8171018af8bb2'),
  username: 'dev_coder',
  fullName: 'Devansh Mehta',
  email: 'devansh.mehta@urbanwave.in',
  age: 24,
  bio: 'Data Science Enthusiast',
  interests: [
    'AI',
    'ML',
    'Python',
    'NLP'
  ],
  location: {
    type: 'Point',
    coordinates: [
      70.4579,
      21.5222
    ],
    city: 'Bhavnagar',
    state: 'Gujarat'
  },
  screenTime: 180,
  followersCount: 2,
  followingCount: 3,
  isActive: true,
  createdAt: 2025-09-10T16:35:36.921Z,
  lastSeen: 2025-09-10T16:35:36.921Z
}
{
  _id: ObjectId('68c1a9fefcc8171018af8baf'),
  username: 'meera_fun',
  fullName: 'Meera Joshi',
  email: 'meera.joshi@urbanwave.in',
  age: 22,
  bio: 'Making people laugh',
  interests: [
    'comedy',
```

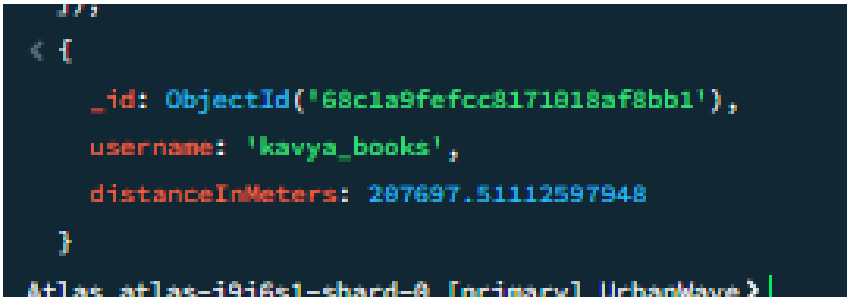
Solution Output: Returns 2 users: Meera (Ahmedabad), and Devansh (Bhavnagar).

4.1.5 5. Calculate distance between two users

Problem: Calculate distance between Priya and Meera. **Solution Code:**

```
const userA = db.users.findOne({ username: "ananya_foodie" });
const userB = db.users.findOne({ username: "kavya_books" });

db.users.aggregate([
  { $geoNear: {
    near: userA.location,
    distanceField: "distanceInMeters",
    spherical: true,
    query: { _id: userB._id }
  }},
  { $project: { username: 1, distanceInMeters: 1 } }
]);
```



Solution Output: Returns the distance between the Ananya and Kavya in meters.

5 Design Decisions & Implementation Details

5.1 Schema Design Rationale

The database schema was designed with the following considerations:

- **Users Collection:** Includes comprehensive user profile information with GeoJSON location format for geospatial queries
- **Posts Collection:** Supports both posts and reels with engagement metrics and optional location data
- **Followers Collection:** Simple relationship model to track following connections between users

5.2 Index Strategy

Strategic indexes were created for:

- Unique constraints on username and email
- Geospatial queries on location fields
- Performance optimization for common query patterns
- Social network relationship queries

5.3 Data Validation

Comprehensive schema validation ensures:

- Data type integrity
- Required field enforcement
- Range validation for numeric fields
- Proper GeoJSON format for location data

6 Conclusion

This assignment successfully demonstrates the implementation of a complete MongoDB-based social media platform database. The solution covers all required aspects:

- Proper NoSQL schema design with validation
- Comprehensive CRUD operations
- Advanced geospatial query capabilities
- Performance-optimized indexing strategy
- Real-world social media functionality

The UrbanWave database provides a solid foundation for a location-aware social media platform with robust data management capabilities and efficient query performance.