Todd Gavin

27 March 2023

DSCI551: Foundations of Data Management

**Project Midterm Progress Report**

* Project: Firebase Emulator using Flask, WebSockets, and MongoDB
* Team: Solo Project

Midterm Progress Report:

* MongoDB Structure
  + Collection Data:
    - Each collection can contain multiple documents (similar to Firebase nodes)
    - Can be accessed using their unique identifier (similar to Firebase keys)
  + Nested documents for hierarchical data:
    - Each document contains other documents (represent nested nodes in Firebase)
    - Use a unique identifier for each document (represent a key in Firebase)
  + Arrays to store ordered data:
    - Array index to represent the order of the data (similar to how Firebase stores ordered data using arrays)
  + sub-collections for complex data:
    - sub-collection can contain documents (represent nested nodes in Firebase)
  + References for related data:
    - Can use references in MongoDB to link the data.
    - Use a unique identifier to link the documents (similar to how Firebase stores related data using references)
* Code Progress:
  + So far the functions GET, POST, PATCH, DELETE and PUT are complete and working.
  + Currently half filter functions of orderBy, limitToFirst/Last, equalTo, startAt/endAt are complete.
  + Therefore, CRUD to and from the MongoDB is working as normal with PyMongo library.
  + As far as setting up the RESTful server, I still need to develop that so it is constantly listening for commands to push and pull data from the MongoDB.
  + The next step of my development is setting up the proper indexing for my MongoDB to match Firebase.
  + And finally, the last thing I will complete is developing the command line interface to properly accept curl commands.
* As far as meeting milestones…
  + I believe I am on track, but I do wish I was little farther ahead on developing the filtering functions completely.
  + The server and setup and developing the CLI will be a very time-consuming part of the project.
* Difficulties:
  + The hardest parts I foresee is indexing, setting up server, and developing CLI.