Lab 2 – Chirp (graded)

205.2 – Beyond relational databases

Goals and information regarding this lab

The goals of this lab are:

1. Learn to model a practical data-intensive application as a key-value database.
2. Be able to translate requirements into implementation tasks and data modeling.
3. Implement and interact with a key-value store from a programmatical point of view.

The estimated duration of this lab is **4 periods**. If you have not completed the laboratory in the time allowed, you are invited to complete it at home.
Credits: the CHIRP acronym meaning was generated with an AI-powered tool.

Submission:

Your submission will consist of two elements:

1. A **link** to a repository in Github or Gitlab holding a proper clean, professional version of your solution. Concretely it should contain:
    a. `readme.md` file explaining how to compile/run your solution, as well as who the authors are.
    b. the source code files.
    c. you don't need to include the original raw data. If any other intermediate data has been generated as an input for your solution, please include it as well.
2. A report including various analysis needed:
    a. introduction and motivation
    b. requirements clarification, assumptions made, decisions and reasons
    c. data modelling as key-value
    d. software architecture and functionalities
    e. conclusions & future work (if applicable)

The submission of these two elements should be done by using ISC moodle no later than **30th of April 2025**.

Part 1 - Setup preparation & data wrangling

Task 1- Install necessary packages

The provided data is in json format. You can use jq to query it. You can install it in Linux by doing:

```
sudo apt-get install jq
```

For the last part you will need the streamlit python library installed. This can be easily achieved by doing:

```
pip install streamlit
```

Task 2- Download -and get familiar with- the data

You can get the data from ISC moodle. You can use the newly installed `jq` command to understand the data given, familiarize yourself with the format and content as well as what can be done with it.

Part 2 - CHIRP: Compact Hub for Instant Real-time Posting

For this laboratory you will implement a lightweight simplified version of Twitter/now X (https://twitter.com/) called Chirp (Compact Hub from Instant Real-time Posting).

The minimal information that your solution should store and display:

- Following/followers → each use can have other users as followers and can follow other users
- Chirps → users can post small text-only messages (in English)
- Keep top 5 users with higher number of followers
- List of latest 5 chirps
- Top 5 users with higher number of chirps

Task 3- Make sure you have all the information needed

Just like when working for a company, sometimes the requirements need refinement, make sure that you understand what is needed and you have all the data necessary to do it, by asking the teacher. You can go on to task 4 and reiterate over this one.

Task 4- Key-value data modeling in Redis

Before starting to code anything, you should model the way you will store data. According to:

1. the functionalities
2. the available data structures in Redis
3. the available *or missing* data twitter (you would need to make up for the missing data somehow)
4. what would be needed in terms of performance

The output of this task would be a design of how the data will be organized as key-values.

Task 5- Implementation of features

In this task you will use a programming language like Python to implement the functionalities. ***Note***: is not mandatory to do this task in Python, you can use any other language of your choice, provided it has a library compatible with Redis. However, you would need to search for an alternative for part 3.

You do not need to implement the following/followers part, only include the number of followers per user.

Task 6- Import relevant data to Redis

Now that you have the data model and the program that goes with it, you should import relevant data from the existing dataset and adapt it to the specifications and needs.

Hint: use tweets written only in English. You can filter by 'lang' == 'en'.

Part 3 - A tiny web app for Chirp (bonus 10%)

Task 7- Installing and getting started with Streamlit

You will be using Streamlit and Python for creating a very simple webapp based on your Python code.
Before integrating your data try out a hello world web application locally.

Task 8- Make a tiny web app for Chirp

Add streamlit components to your main python program to make it interactive from the web.