1. The general intent was clear. Do some data processing and build these endpoints. However, if I received this as Jira ticket I would have many questions, which would help me refine the specs. Ex: Error handling &Logging philosophy, what are valid country inputs, do we want to accept two letters country codes AND full country names, and many more. I do not think anything was missing per-say, but I liked how I was left to make some executive decisions on my own.
2. Cumulatively I’m not quite sure how much time I spent on this project, as I was having fun with it. I know I had 2-3 cups of coffee over the course of a couple of sessions, so that would translate to 3-4 hours. Which includes looking up documentation and things I forgot about Flask and such.
3. I went with Python because it is what I’ve used primarily in a professional environment. In my opinion it is also aesthetically pleasing.
4. Sqlite is nice and lightweight as well.
5. I used the Flask framework. I utilized it once before, on a smallish blog type project ~8 months ago. I remembered it was lightweight and easy to work with so I went with it again.
6. Okay here is an awesome question. With more time and resources, I would add more eloquent error handling, class based controllers, query parameter cleansing/sanitation (aka type/value checking), I would ditch the ORM and use hardcoded SQL for performance gains where applicable, A caching layer would also be a must have, and finally TESTS TESTS TESTs.
7. To be honest the whole thing would not survive any sort of production load. It would blow up and die. If the above suggestions were implemented along with tossing this into an auto-scaling group with a load balancer, then I think it would have a chance. Depending on how frequently repeat requests come in, that cache layer would be super valuable.
8. For a larger dump I could not get away with how I laid out the relationships. The transaction table would have to separated into Watch, Like, and Upload tables. This binning technique would help alleviate some of the burden but not all that is necessary. To handle more complex queries, I’ve been instructed that there are two ways to accomplish this. Either write the query as optimal as you can or reduce the search space by breaking up your tables into a more granular layout.
9. For RMM I had to do a quick google. But I’d rate it at a 2.
10. I’m confident in the functionality of my solution, as I have traced the logic of each route through my mind many times, and I keep concluding that it makes sense.(Test cases to verify solutions would make me feel even better : P).What I am more interested in is how I could do it better.
11. I had fun with it. I was a little daunted by the word SENIOR in the coding challenge but I had some coffee and gave it my best shot. I hope the code is clear concise, documented, and makes sense. I’d love to get some feedback on possible improvements and or any thoughts you all may have.