



Software Engineer, Full Stack Developer

Reference:

Technology / O3b mPOWER

Date Issued:

11 October 2021

Software Engineer, Full Stack Developer

1 Questionnaire

Respond to the following prompts in each of the categories below.

▲ Database Exercises:

- The following table schema is meant to link the person with the city in which he/she currently resides. Given this current schema, could you illustrate the main pitfalls of such a design?
 - Note: The only primary key is the table ID.
 - How would you re-arrange such design so that:
 - Each person is unique?
 - The same person can live in different cities at different times?

| ID (PK) | City of residence | City coordinates | Name | Surname | Person ID number | Current Job Role | Date of registration |
|---------|-------------------|------------------------|---------|------------|------------------|-------------------|----------------------|
| 1 | Milan | 45.4642° N, 9.1900° E | Marc | Zuckerberg | 00001111 | CEO | 01-01-2021T13:00 UTC |
| 2 | Rome | 41.9028° N, 12.4964° E | Patrick | Smith | 00001234 | Software engineer | 01-02-2021T13:00 UTC |
| 3 | Rome | 41.9028° N, 12.4964° E | Marc | Zuckerberg | 00001111 | CEO | 01-03-2021T13:00 UTC |

Table 1 – Sample Database Schema

- Define the following database schema:
 - Satellites are uniquely identified by a string called 'satellite_id' (e.g., "F01") beside other data (like "launch_date", "in_use", "orbit"); every Satellite has a set of parameters, but we would like to expand/add more in the future; the type of the parameters might change over time.
- Query:
 - Given the above design, write a query that will retrieve the satellite ID's of all 'MEO' orbit satellites that have a parameter called 'tm1_frequency_mhz' equal to 19008.



Software Engineer, Full Stack Developer

▲ Coding Exercises:

- Find the most common character in a sequence of strings. E.g., given the input ["a", "bb", "a", "cc"], your program should output: "a".
 - Given that:
 - (1) You may assume that there is only one of such character.
 - (2) Select a data structure to store the input.
 - (3) Explain why you have used that particular data structure.
 - (4) Analyze the time and space complexity of your algorithm.
 - (5) Use any language you prefer (even pseudo-code).
- The following code is not written very well. It is a mix of pseudocode and Golang and it computes the satellite visibility based on an ephemeris file type ("A" or "B") and returns it to the user. Analyze the following code and explain why and how you would refactor it using pseudo-code. Highlight the main changes and mistakes in the current design.

```
package router
•
• // ComputeVisHandler computes the satellite visibility
• func(s SatelliteRouter) ComputeVisHandler(start, stop time.Time, ephemerisType string) {
•     // queries the DB for all satellites
•     satellites := db.ListAllSatellites()
•     s := make(map[string]int64, 0)
•     g := make(map[string]int64, 0)
•     for _, ss := range s {
•         satMap[ss.Name] = ss.ID
•     }
•     // queries the DB for all gateways
•     gateways := db.ListAllGateways()
•
•     for _, gg := range g {
•         g[gg.Name] = gg.ID
•     }
•     // queries the DB for all satellite parameters
•     sp := db.ListAllsatelliteParams()
•     // zzzzz...
•     LongLastingOperation()
•     v := make(map[string][[]])
•     if fileType == "A"{
•         for _, ss := sp{
•             // do something with ss
•             v := computeVis(ss, start, stop)
•         }
•     }
•     if fileType == "B"{
•         for _, ss := sp{
•             // do something with ss
•             v := computeVis(ss, start, stop)
•         }
•     }
• }
```

Software Engineer, Full Stack Developer

```
•     }  
•     }  
•     return v  
•     }
```

▲ System Design Exercises:

- Sketch the system design modules for a distributed GIT repository in two different regions (e.g., US and EU).
 - Address at least redundancy and replication aspects.
 - Chose the granularity of the design but try to keep it at a high-level perspective.

2 Project

Background: Our cloud-deployed systems must be able to scale on-demand. For each new incoming request, we want to redirect it to just **one backend server at a time** according to a customizable URL mapping. However, the same incoming URL at a different time can be redirected to another server (you should decide according to which policy; think about the best approach and the pro and cons of it). E.g.,

Time T_0

From: GET /api/v1.1/satellite/MP01

Redirect to: GET hostname.com/api/v1.1/satellite/MP01

Time T_1

From: GET /api/v1.1/satellite/MP01

Redirect to: GET secondary.com/api/v1.1/satellite/MP01

You need to code this proxy, preferably in Golang, so that a client can be served without relying on just one server.

In case no mapping can be found the client shall return a 404 error.

You do not have to account for:

- Authentication/authorization
- Sender IP



Software Engineer, Full Stack Developer

Your minimum implementation must support the following features:

- Allow URL mapping in a file
- Post your solution on a private repo
- Make a presentation explaining at least:
 - Design choices
 - Clearly explain your thought process on approaching this problem
 - Explain the challenges you had while implementing it
 - Explain in which context this software would make sense and in which use-cases you would use it for
 - How would you improve it in the future?
 - Which security feature(s) you would add?
 - (NOT REQUIRED TO ADD ANY, just explain which one(s))

Nice to have features:

- Allow dynamic rule customization while the program is running
- Filter on method
- Use a DB for saving the rules
- Build a UI using your preferred technology

---END OF DOCUMENT---