**System design—Problem and solution**

I needed to design a system that should hold the following items:

* Accommodate increased load
* Managing complex data processing
* Ensuring high availability
* Maintaining security under strict constraints

I would performed the following actions for each item described above.

**Accommodate increased load**

* Open database, design tables/fields and normalize database tables so that it can be easy reading and fast access onto records.

**Managing complex data processing**

* Use libraries like Spark framework or similar kind, in order to give data the meaning or shape that user is expecting.
* Use intrinsic functions and/or tools from the motor of database, in order to give data the meaning or shape that the user is expecting.

**Ensuring high availability**

* Use data caching when the application is alive.
* Use threading to get the user request and give the response to the user by making small requests (based on the original request) in an asynchronous manner.
* Use server load balancing as a strategy

**Maintaining security under strict constraints**

* Create Primary Key, Foreign Key, Check constraints for database tables’ fields.
* Create API(s) that may connect to database.
* Create user authentication by user-password, google authenticator factor, or face recognition.
* Create roles, profiles, permissions for application' users.