Question 61

Not yet answered

Marked out of 1.00

You have been given a set of requirements for a new system that will take a set of status updates and save them in a database. These status updates will be defined as a date/time + message. Each status update will be submitted as a set of two strings, one for the date/time and one for the message. While the statuses will be posted fairly infrequently, they will be heavily queried for. These queries commonly search for status messages within a certain time period, or for messages that contain a certain keyword.
If you were to design this system to include a robust cache, what part of a caching system would best improve performance?

Select one:

● By using a proper hash table that ensures as few collisions as possible given the dataset

○ By distributing the cache across a global CDN (such as Amazon's), so that requests are properly routed to a nearby datacenter

○ Ensuring that there is as little processing as possible between status submission and saving in the database + cache updates

○ By doing pre-processing on the date/time string to convert to a standard dateTime to optimize date operations

○ Building a deep learning algorithm and training it across the common status message requests to build a prediction engine that pre-caches requests before clients even request them

Question 62

Not yet answered

Marked out of 1.00

Which guideline should NOT be considered when designing an user interface?

Select one:

○ a. Cached data is in a ready to use format instead of the raw format in which it was retrieved.

○ b. Device-dependant input, such as ink or speech, is considered in the design.

○ c. Navigation and UI processing are integrated and act as one.

● d. Validation routines are centralized, where possible, to improve maintainability and reuse.

Question **63**

Not yet answered

Marked out of 1.00

You're working on a new business critical service that for a set of customer repositories, as part of a CI/CD process, evaluates every check-in against 3 quality checks before ultimately deciding if the check-in is GOOD or BAD. Each of these quality checks have already been implemented as a webapp/service. Each time a new check-in is built by the CI/CD system, your component orchestrates the quality analysis; this is done by notifying the analysis sub-systems of any new check-in, and waits for them to respond with their individual evaluation. As this is a business critical app, you want to add some monitoring to the production deployment, along with alarms when certain thresholds are met. You have come up with several indicators you can monitor and have had your dev team put estimates for each one. Considering in this release you have one week (5 days) of dev-time allocated for monitoring, which monitoring indicators do you select to implement?

Select one or more:

☑ (.5d) Infrastructure indicators: CPU/Memory/Disk space

□ (1d) Time from when your service is notified of a new check-in and each sub-service is notified

□ (1d) Time from when a sub-service replies with its evaluation and the local service's DB/Data structure is updated

☑ (1d) Check-in evaluations requested per hour

☑ (1d) Indicators for each subservice: Time to evaluate, time out occurrences, pass/fail rates

☑ (1d) Total evaluation time from check-in notification through final recommendation

□ (1d) The % of new metric: "Complete/Incomplete evaluations", which is the number of evaluations that successfully completed vs evaluations that had one or more sub-service evaluations fail

☑ (1d) The pass/fail/error evaluation fail rate for each individual repository, to determine if any individual repo is having systemic issues

☑ (.5d) The Queries Per Second (QPS) rate for the overall service

Question 64

Not yet answered

Marked out of 1.00

You need to implement HTTP session management without using server-side storage of any kind, so you opted to use JSON Web Tokens. All goes smoothly and the required scalability is achieved, but a new problem appears. Which one?

Select one:

○ a. CPU load has increased too much on each request

○ b. It is impossible to log out all users at once

● c. It is impossible to log out a specific user

Question 65

Not yet answered

Marked out of 1.00

What are the most important characteristics of a system that uses service-oriented architecture (SOA)?

Select one or more:

☑ Performance of the entity classes

☑ Loose coupling

□ XML web services

□ Stateful session handling in a load-balanced cluster

☑ Well-defined contracts between message producers and consumers