SWE 4537 - Quiz II

Topics: Push Notifications, System Designing, Load Balancing, CDN, Deployment, etc. [Everything covered after mid]
Total number of questions: 30
Quiz Duration: 30 minutes
Quiz Link: [I will edit this field and add quiz link at 1:38 PM]
Quiz Guidelines:
1. If there are multiple correct answers to a question, select all of them.
2. Each question is responsible for 1 point.
3. Quiz will be started at 1:40 PM. Try to join the meeting by 1:35 PM.
4. Submit your response by 2:10 PM.
5. Exam rules are same as mid/final semester examination.
6. You have to keep your camera on during the quiz. Make sure that you have a backup source of internet.
7. Since this is an MCQ exam, you can use Laptop, PC, Tablet or Mobile as your answering device.
8. Closed book exam. You can only use keyboard for writing your name and student ID at the beginning of the
exam.

Zoom Meeting Link:
1. ID (1 - 25): https://bdren.zoom.us/j/66729547608
2. ID (26 - Rest): https://zoom.us/j/5995656166?pwd=SlAwWjl3MzJNMVlrN0VTczJhZ1pmUT09

Email *

tasnimahmed@iut-dhaka.edu

Full Student ID

154407

Name

Tasnim Ahmed

Questions

What are the advantages of saving images as BLOBs if you have to deal with millions of images? 1 point

- [x] Query through images
- [ ] Takes less space
- [ ] Less resource hungry than saving images as a file
- [x] Image based search

In case of saving image as a BLOB, which of the followings are false? 1 point

- [ ] Query through images
- [x] Takes less space
- [x] Less resource hungry than saving images as a file
- [ ] Image based search

In case of news feed generation which of the following will result in less latency for the users? 1 point

- [x] Pre-calculating News Feed based on followees' uploads
- [ ] For each refresh request, pull 50 images and pass it through ranking algorithm.
- [ ] For each refresh request, pull 30 images and pass it through ranking algorithm.
- [ ] Performance will be the same

For sending notifications to the instagram followers which method would perform better? 1 point

- [ ] Pull
- [ ] Push
- [x] Hybrid
- [ ] Probably the same.

In case of Instagram, which database sharding mechanism is more likely to perform better in the long run? 1 point

- [ ] Consistent hashing on userID
- [ ] Consistent hashing on PhotoID (incremental-based on time)
- [x] Based on photo creation (upload in this case) time
- [ ] Probably the same

De-coupling business logic tier results in microservice architecture. 1 point

- [ ] False
- [x] True

For massive scale image delivery, a system like instagram needs - 1 point

- [x] Caching
- [x] Content Delivery Network
- [ ] Optimized Notification System
- [ ] None

Losing images for a system like Instagram is not an option. What should you do? keep in mind that you have minimum 500 million users and at least 1 million active users. 1 point

- [ ] Caching
- [ ] Load Balancing
- [x] Storing multiple copies of same image in different servers
- [ ] None

Multiple instance of same service running on different servers helps you in - 1 point

- [x] Balancing your load
- [x] Helps you to get rid of single point of failure
- [x] More cached data
- [ ] None

Which of the followings might be good caching strategy for Instagram? 1 point

- [x] Caching celebrity images
- [x] Caching instagram influencer images
- [x] Caching celebrity/influencer stories
- [ ] None

Instagram uses 'cache-first rendering'. It basically shows you data saved on client side (from your previous uses) till you receive a response from the server side while you open the app. When the fresh - server side data reaches cached data is replaced. Should you store 'interaction functions - like, comment' for the cached data? 1 point

- [x] Yes
- [ ] No

For one to one messaging we usually use - 1 point

- [ ] Client - Server protocol
- [x] Peer to Peer protocol
- [ ] HTTP
- [x] XMPP

To increase security of user data tech giants usually use - 1 point

- [ ] HTTPS
- [x] Their own protocol for internal communication
- [x] Firewall for internal servers
- [ ] XMPP

Which of the followings are feasible for UBER? 1 point

- [ ] Sending request from server to all cabs of a geo location and cabs respond if they're active or not.
- [x] Every cab which is active keep on sending lat-long to the server every 5 sec once.
- [ ] Hybrid
- [ ] None

Dispatch Optimization at UBER (is known as DISCO) uses nodeJS and web sockets. Why? 1 point

- [x] Asynchronous functionality
- [x] Both user (Passenger and Cab) and server can send messages whenever it wants to.
- [ ] Client Server Mechanism
- [ ] None

For UBER, sharding database based on _____ seems to be a better option. 1 point

- [x] Longitude-Latitude
- [ ] UserID
- [ ] Cab types - Sedan, Luxury Sedan, 7 seater etc.
- [ ] Driver Rating
- [ ] User Rating

Invocation API - 1 point

- [ ] Sends notifications from client to server
- [x] Controls how notification should appear on your device
- [ ] Controls how user interacts with it
- [ ] None

Notification API - 1 point

- [ ] Sends notifications from client to server
- [x] Controls how notification should appear on your device
- [x] Controls how user interacts with it
- [ ] None

Push service uses separate endpoints for each client and this endpoint is generated upon client's subscription. 1 point

- [x] True
- [ ] False

Service worker - 1 point

- [x] Receives the message
- [x] Shows message as notification
- [x] Listens for event
- [ ] None

Using multiple servers attached to a load balancer, where each of the servers contains a copy of business logic tier is 'Microservice Architecture'. 1 point

- [x] False
- [ ] True

Permanent storage tier is where we keep our html, css, js etc. 1 point

- [x] False
- [ ] True

Which of the followings help you to reduce load on your servers? 1 point

- [x] Code Refactoring
- [x] Horizontal Scaling
- [x] Vertical Scaling
- [ ] None

Which can result in expired content on the client side? 1 point

- [ ] Push CDN
- [x] Pull CDN
- [ ] Direct calling main servers for data
- [ ] None

It is always wiser to roll-back when an error occurs immediately after deployment. 1 point

- [ ] True
- [x] False

Fixing your new feature deployment time should depend on - 1 point

- [x] Business case scenario
- [x] Amount of user load on server
- [x] Amount of engineers available at that time
- [ ] Arbitrary

CI is responsible for pushing every update to the users as soon as they are available. 1 point

- [ ] True
- [x] False

Deployment permission should be available to every engineers so that the cay push updates as soon as they are available. 1 point

- [ ] True
- [x] False

Instagram story expires every 24 hours. Also, Facebook, Instagram - these platforms are read-heavy systems. Not write-heavy systems. For implementing instagram story which of the following will you chose? 1 point

- [ ] Pull based - When a user logs into the instagram, server will collect his/her followee stories from DB and send it to user.
- [ ] Precomputed Pull based - Server will keep a queue for every user to store story queue. When one of my followee posts a story my queue will be updated and response will be sent back to me when I log in or refresh the page
- [ ] Push based - When one of my followee posts a story, server will push that story to my device
- [x] Push based for regular user, Pull based for celebrity like Eminem, Kylie Jenner, etc.

Cache eviction policy is very important. If you do not optimize this your cache is of no use. Also you have to keep in mind that you can not keep everything in cache. Because cache storage (usually high speed SSDs) are a lot more expensive than where you store your normal database (Usually HDD). A better cache eviction policy might help you to - 1 point

- [x] reduce network calls
- [x] reduct DB load
- [ ] expired content
- [ ] more accurate data