Project 5 - Quora

1. Individual profiles

i. Public profiles, anyone can search and follow

ii. Private profile, anyone can search but approval is needed before showing the content

2. Business profiles to promote a brand or an organization like Amazon etc.. i. This accounts should have a moderator

ii. Only moderators are allowed to publish the content

iii. Only moderators are allowed to configure the list of ads to show up

iv. Moderator can start a new category and after only new posts are allowed

iv. Participants should be allowed to comment, like, and dislike the content

3. This system should allow publishing the following content

i. Text posts

4. Comments/answers support

i. Nested commenting should be allowed a minimum of 5 levels - 1 level

ii. Should be able to add links from other locations

iii. Should have the ability to write code snippets and this snippet should be shown with different style from other comments

5. Comments visibility

i. Private profiles, a moderator or owner should approve before the comments appear for others and notification to be sent

ii. For public profiles comments should be made visible without any need for approval

iii. Include profanity filter and block a user from adding offensive comments

6. Each post should have Like, Dislike and emojis

7. Events post by individuals or business should be allowed to share within Quora and on Pagebook platform

8. Once the moderator/individual closes a thread, no more comments are allowed on the thread and the comments section

9. Users landing should have the following content

i. Feed from the friends and business profiles he/she is following

ii. Feed should be shown based on the sequence in which they are published iii. Option to search for friends on the system

iv. Public profile entire content should be shown while searching

v. Private profile only the user profile should be visible, not any posts. Once the approval happens the post needs to be made visible to the follower

vi. Notifications of the posts and comments from the list of followers

vii. Business profiles follower requests should be approved by a moderator

10. User profile

i. Should have an option to set up an image - current

ii. Should have the ability to choose the list of interested domains

11. Ads display

i. Feeds should also include ads based on the categories user have chosen as part of the profile

ii. If the user has not selected any interests, needs to show random ads across different categories - minimum 3 interests

12. This functionality should be supported from both Mobile and Web version

13. Data tracking requirements

i. User clicks on likes and dislikes should be captured in a common analytics system ii. User comments on the post should be captured (primarily the category of the post belongs to)

iii. If there are dislikes in the post related to a business profile, user details should be captured in the CRM system

14. Notifications support

i. Mobile notification for iOS and Android

ii. Web notifications if possible

15. User Ranking

i. Each question posted by user should fetch him 10 points

ii. Each answers like should fetch him 1 point

iii. Each dislike should fetch him -1 point

iv. Each accepted answer should fetch him 5 points

16. User classification

i. Between 0 - 1000 points beginner

ii. Between 1001 - 2500 silver level

iii. Between 2501 - 6000 gold level

iv. Above 6K should be platinum level user

Databases

Account (Based on input from CommonInfra)

String id

String profileImage

String userName

String emai~~l~~

String description

String phoneNumber

String gender

Enum gender -> male, female, prefer not to say

String password

String scoreBadge

Int score

Enum accountType -> public, private, corporate

– refer to this document for your need https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html

String isOwner -> (true for public,private, owner of corporate profile) <- Boolean

FollowersAndFollowing

String accountId

List<String>followers - accountId - figure out how to append to list in mongo query and how to delete one value from the inner list

List<String>following - accountId

List<String>followRequests- accountId

Question

String description

String codeEmbed - can be empty string

String uRLEmbed - can be empty string

String accountId

Int numberOfUpvotes

Int numberOfDownvotes

Date date

Boolean isLive

String category

Answer

String questionId

String AnswerDescription

Int NumberOfUpvotes

Int NumberOfDownvotes

Date DateModified

String accountId

String CodeEmbed - can be empty string

String URLEmbed - can be empty string

Boolean isAccepted

Comments

String accountId

String ParentId

Int level (if level id is 0 then parent Id is is answer table else parentId is in comments table )

String CommentText

String CodeEmbed - can be empty

String URLEmbed - can be empty

MicroServices

1. Profile
2. Post - Questions-Answer-Comments
3. Search

1. Profile

Entity:

Account✅

Moderators

UserPrefCategories

FollowersAndFolllowing✅

DTO:

AccountDto

ModeratorsDto

UserPrefCategoriesDto

FollowersAndFolllowingDto

PostDto

FeedsDto

ResponseDto

For Internal Purpose Only

**finById(String id)**

**AccountDto createAccount(AccountDto acc ) - post ✅**

* Check accountType
  + If business - Update List of Moderators
* Send to commonInfra
* Set isOwner - true or false

**AccountDto viewAccount(String id ) - get ✅**

* String checkVisibility (String Id) - internal

**ResponseDto FollowRequest(String requesterId, String requestingId) - post✅**

* Push notification
* findByid(requestingId) - > FollowersAndFollowing(table) followRequests add - requesterId
* // success if a notification has been sent to requesting account else failure

**ResponseDto AcceptRequest(String requesterId, String requestingId) - post✅**

* If accepted
  + Add requesting Id in requester’s followers List inFollowersAndFollowing(table)
* Else
  + findByid(requestingId) - > FollowersAndFollowing(table) followRequests remove - requesterId
* // success if a requesting account has responded

**ResponseDto DeclineRequest(String requesterId, String requestingId) - post✅**

* If accepted
  + Add requesting Id in requester’s followers List inFollowersAndFollowing(table)
* Else
  + findByid(requestingId) - > FollowersAndFollowing(table) followRequests remove - requesterId
* // success if a requesting account has responded

**List<QuestionDto> showFollowingFeeds (String accountId) - get -**

* Create A FeedList(Mega List)
* Find the FollowingList
* Collect Questions posted by the FollowingList merge of FeedList
* Desc Sort the FeedList based on date posted
* Return List<QuestionId>/<QuestionDto> FeedQuestion from Feeds table
* If list is null
  + Return random category feed

2. Posts

Entity:

Question✅

Answer✅

Comments✅

For Internal Purpose Only

**finById(String id)**

**QuestionDto AddQuestion (QuestionDto question, String accountId) - Post ✅**

- SendNotification

- Post Collection - > Add to List<QuestionPosted> - for this accountId

- User table score+10

**AnswerDto AddAnswer (AnswerDto answer, String accountId) - Post✅**

* SendNotification
* Post Collection - > Add to List<QuestionAnswered> - for this accountId
* isQuestionLive
* UserTable score+10
* UserTabel - score-1

**ResponseDto addComment(String ParentType, String ParentID,String QuestionId, CommentDto commentDto) - Post✅**

* isQuestionLive

**ResponseDto upVoteQuestion(String Id) - Post**

- if(type== question) - score+1

**ResponseDto downVoteQuestion(String Id) - Post**

* if(type== question) - score-1

**ResponseDto upVoteAnswer(String Id) - Post**

- if(type== question) - score+1

**ResponseDto downVoteAnswer(String Id) - Post**

* if(type== question) - score-1

3. Search

For Internal Purpose Only

finById(String id)

SearchByUserName

References:

Quinstion

| Collections.sort(toReturn, Comparator.comparing(s -> s.getLikes())); |
| --- |

// Date should be set at the back end not in the front end

// remove date from dto

//Add userName and profile Image in question and answerDto

/// moderator - private profiles - accept answer - show question thread as of now when clicked - api

1. Answer / comment to be accepted
2. Question and answers as of now