Commander of Full Stack

Quiz Organizing Platform

Create a platform to **conduct quizzes** where an **admin** can **configure** a **quiz** and **share** with **quiz takers**.

- Create a mobile or web app that allows a user to create a quiz and invite participants to the quiz. This user is called Quiz Admin.
- Anyone should be able to register and then create a quiz by naming it and adding questions to it. Registration is mandatory only for someone who wants to be a quiz admin and create a new quiz.
- Quiz Admin can share links with **Quiz Takers** to take the quiz.
- A Quiz Taker can open the link, enter their name, and attempt questions and submit the attempt. They need not register. They can just provide any name and attempt the quiz.
- Quiz Admin specify start date and time, end date and time. For example, admin can say 'Participants can attempt quiz anytime between *18 June 10 am* and **19 June 10 pm**.
- Quiz admin can manually end the quiz any time.
- Admin should be able to view the scores.

Essential features which are expected

Without these, an entry won't be considered valid. After all, what fun is a quiz app that can't conduct any quiz :)

1. Quiz Admin Register, and Sign In flow
2. Quiz Admin can publish a quiz.
3. Quiz taker can attempt a quiz using a shared link.
4. Score of all participants for a quiz to be available to quiz admin at the end of quiz

What will make your software a winning candidate?

The essentials described above constitute a barebones quiz organizing app. A winning candidate will be one that is richer and powerful in functionality, and of course performance!

Robustness and basic exception handling will play the biggest role in the winning candidate's score. This means really paying attention to obvious aspects like validations, handling edge cases, and preventing unexpected output!

We would want you to form your own opinion on what a winning entry would look – we have all been taking Quizzes for a long time. At the same time, time is of the essence (in what Quiz it is not!) and we would urge you to keep in mind the time available to attempt the problem statement. Three very well implemented features are much better than eleven half-baked ones. With this caution, we will share some ideas on what you can evaluate from and pick to create a winning entry –

- **Signup**
  a. What options you allow for signup/registration? Can I signup using common methods such as using Google Account etc.?
- **Quiz taking design choices**
  a. How do you prevent one person from taking quiz multiple times? Do you provide a unique URL per quiz taker?
- **When quiz admin sets rules for questions and scoring, what flexibility and power do you provide to them?**
  a. Can admin set timer for the entire quiz?
  b. Can admin set timer per question?
  c. Can admin give fixed scoring only (each question has same weightage) or also allow variable scoring?
  d. Can admin provide a mix of questions such as MCQ, free text answers etc.?
  e. Can admin give other parameters such as speed of answering for scoring?
  f. Can admin specify that questions should appear in random order? What about options for MCQs?
  g. Can admin export quiz report and scores?
- **Experience of quiz taker**
  a. Once quiz is over, does quiz taker get a summary of their responses? What kind of summary do they get?
  b. Does quiz taker get an instant response (correct or incorrect answer), or they get it at the end of quiz?
  c. Does quiz taker get a live score (their own score) or they get it only at the end of quiz?
- **Statistics**
  a. Do you provide any statistics to the quiz master? Somethings like 'most correctly answered question', 'most incorrectly answered question' are interesting. You can add more fun by telling things like distribution of browsers used to take the quiz? What else? Think!!
  b. What does quiz score dashboard refresh? Does it refresh every second? Does it refresh with a manual button? Does it refresh only at end of quiz?
- **Types of questions**
  a. Do you allow audio or visual questions?
  b. Do you allow questions where more than 1 option is correct answer? How do you score them?

*All above have a common theme – attempt to go the extra mile in whatever you do. These are just **areas to explore**, and **a direction to think into – neither a comprehensive list** nor **representative of what can be done in a weekend.***

Pick and choose, mix and match as you like and can achieve!

Any other factors I need to know? What else would we consider while evaluating your solution

There are a few things we want to call out explicitly

- You may want to strike a fine balance in what all you construct. For example, when giving signup, giving signing using one social media account is good, but providing all possible options may not be the most effective usage of your time.
- Operations such as ability to add Quiz questions are needed for sure. **However, operations such as delete questions, update questions etc. may have a lower priority if they come at cost of some other feature, such as Statistics etc.**
- *Originality of implementation and understanding of systems* plays a crucial role. Copy pasting code or picking up readymade solutions is not something that will help you qualify. Quizizz already exists. The spirit of this event is to create original solutions, from scratch and highlight your skills. It will be great if all participants stick to it!