

## Model

## Schema

## Role

User

- id, phoneNo, progress, avatar  
 societyName, societyId, emailId  
 name, admissionNo, token  
 GoogleId

admin

society 3/4

editor → 1/2

society

user

public

arena

username

editor

admin

editor

user

p. role: admin | manager | user

type = individual | societyMember | societyHead

Verified

flag

Society

id

members: [Users]

societyName

enCs

Riders: [Users]

description

verified

flag

l

Logo, coverImage

est.

ind

current add → Add edition: for the com  
 we

MVP

MCQ  
One word  
(level based) → ring or not

options

events

- id, enquiryDetails : name → phoneNo
- name
- createdTime
- startTime ✓
- endTime ✓
- updateTime
- isMediaLive paused
- state running, ended, restart, blocked
- createdBy individual/group
- description
- rules : [rules]
- prizes : [ ]
- verified

winner  
showLeaderboard

requireToken : true/false

google Id

Score

100

MVP

~~one word~~  
~~level based~~

~~ring or not~~

options

✓  
✓

events

- id, enquiryDetails : name → ✓  
- name, phoneNo ✓
- createdTime : type → ~~start~~ ~~end~~ ~~panoramic~~ ~~key~~
- start Time ✓
- end Time ✓
- updateTime
- ~~is~~ in ~~mediafile~~ paused
- state : running, ended, restart, blocked
- createdBy : ~~individual~~ type, individual/group
- description
- rules : [rule]
- prizes : [ ]
- verified
- editors : [ ]
- requireToken : true/false
- quickAnswers : [ ]
- images : [img]

winner  
showLeaderboard

Question

id

text

mediatype

~~image~~

media

1

type:

~~short answer~~

~~multiple options~~

~~long paragraph~~

~~file upload~~

options: [

text

image

}

]

flag

answer:

p ~~long~~ =  
weightage

medi

{ [

medi-type:

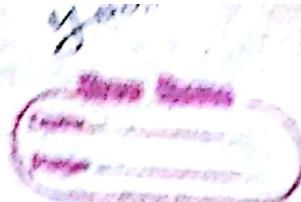
med

~~youtube~~

~~mp3~~

Server - user  
ex

opp



Arena

level

- id

= event ID

- user ID

- score :

- questions : [redacted]

type:  
level  
points

[ }

question ID

state : submitted | unattempted | correct  
correct | incorrect | unattempted

? answer :

Question Index : 0

submitted Time

Answer flag

Score

- id

- question ID

- text

badges

points

level

level

randomizing

skipping

anonymity

level

level

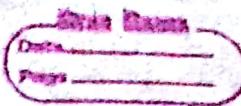
if randomized and skipable  
then variable weightage

if level, then no randomizing,  
skipping, no ~~variable~~ variable  
weightage

if same weightage, then  
skipping optional, randomization  
optional.

10 10 5 5 ?  
10  
5  
4 5 7 10  
10

home /



app /

classboard /

admin /

[done]

[same - weightage, randomized, skipable]

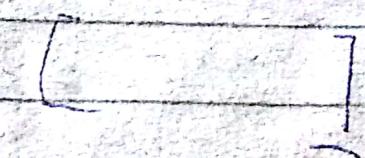
[Variable - weightage, randomized, skipable]. \*

[Variable - weightage, skipable]

[same - weightage, randomized]

[same weightage, skipable]

[same weightage]



plexus - 17

package.json

client /

package.json

src /

build /

development

server - 1

3000

backend

server - 2

5000

frontend

auth/google  
(api)

env vars

production

server

3000

5000

env vars

colleges

- 1) Seed data for admin
- 2) User register and login
- 3) Click on settings button to get option to create event.
- 4) On create event be individual or ~~make~~ as society.
- 5) Choose ~~society~~ If as society then enter it's name, then will check if society exists then ask him to request society to add him.
- 6) If as society then add new society name then ~~is~~ redirect to create ~~old~~ society page.
- 7) On create society page fill details save in db but unverified. Send a notif to admin to verify.
- 8) Ask user to either contact us or mail the association proof on mailId of admin.

- 9). Create society document in db and add user Id to the exec's key array.
- 10). Admin should flag the society if it's fake or ~~does not~~ no contact in 1 week.
- 11). Send mail to user if society approved.
- 12). Now ~~no~~ society can create Event, after creation, notification.
- 13). Event will be verified by admin in all cases.
- 14). User will submit answer by socket.
- 15). When user participates in event an arena document is created for him.
- 16). In arena, the questions will be setup once only & if randomizing then order the questions.

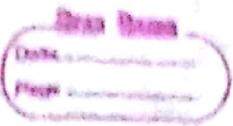
*correct*

- (7) When he gives answer, we update questionIndex, score, & question[questionIndex].answer.
- (8) To check answer, ~~only~~ only the value needs to be checked, in case of options also.
- (9) If questionType is file upload, then add the file URL in answer key.
- (10) Media -  
type = youtube  
then extract the video URL query parameter and make an embedding.  
type = image  
make an img tag and add src.  
type = file

make a download file button.

## (1). Dashboard →

- a) After creating event, you can add images in a batch.
- b) Then for each question, select image and stored the url in mailto.
- c) All the urls of images are in images key of Event document.
- d) After creating event, an exec can add users to be the editors.
- e) Four roles - admin - user  
manager - society admin  
(if event createdBy individual) editor - event admin  
user - event player
- f) When editor goes to events page he gets his event to test.



## Role

for

public

- view events
- register/login

• user

- view create event
- ~~create~~ but unverified
- if event gets verified  
update his role
- submit answers
- leaderboard

• editor

- manage event
- manage questions  
except delete question  
if event selected.
- block questions (update)

• manager

• manage society

• add members

• add exec

• assign editors to events

• admin

King, block users, society  
Event, block question  
send mail, get requests

- 50
- Date \_\_\_\_\_  
Page \_\_\_\_\_
- Groups chat
  - Scan for explicit content

## Dashboard -

(g) ~~on off~~ When add questions to event, there are some preset options to use -

- weight of each question if Event type has same weightage feature
- ask if images in questions and options are needed
- give event type preset from frontend
- preset for question type