

Uite cum am gandit eu aplicatia:

- Se logheaza userul
- **localhost:8000/api/contest**
 - o sa ii apara doar concursurile la care este jurat
 - *id": 3,*
"name": "Concurs de dans 2",
"type": "Battle",
"password": "test",
"start_time": "2019-11-07T19:29:12.964000Z",
"end_time": "2019-11-07T19:29:12.943000Z",
 - din ce primesti tu, o sa ai nevoie de urmatoarele lucruri:
 - ◆ de start_time si end_time ca sa stii cand incepe concursul si cand se termina; daca concursul nu a inceput nu il lasi sa intre mai departe
 - ◆ nume si tip ca sa afisezi sa arate frumos
 - ◆ si de id pentru a putea merge mai departe
- **localhost:8000/api/round/<id contest> POST**
 - aici trimiti parola cand o introduci, doar parola ca sa poti intra in concurs
- **localhost:8000/api/round/<id contest> GET**
 - aici vizual ma gandeam ca dupa ce apasa pe concurs sa intre intr-o lista cu toate rundele din acel concurs, unde se va afisa start time si end time, iar cea curent va fi colorata
 - la rundele ce inca au start_time > time_now nu poti sa intrii
 - *"id": 2,*
"round_no": 1,
"start_time": "2019-11-11T14:35:40Z",
"end_time": "2019-11-11T14:35:42Z",
 - ce te intereseaza aici este doar
 - ◆ start_time si end_time la fel ca si la contest
 - ◆ round_no e numarul rundei
 - ◆ si de id pentru a putea merge mai departe
- **localhost:8000/api/series/<id round>**
 - aceeasi idee ca mai sus, sa afisezi toate seriile din runda respectiva
 - *"id": 2,*

```
"jseries_no": 1,  
"start_time": "2019-11-11T15:16:31Z",  
"end_time": "2019-11-11T15:16:34Z",
```

- ce te intereseaza:
 - ◆ start_time si end_time la fel ca si la contest
 - ◆ jseries_no e numarul seriei
 - ◆ de id pentru a putea merge mai departe

- **localhost:8000/api/participant/<id series>**

- aici gasesti lista cu participantii la seria respectiva
- in functie de timp si de battle afisezi ce concurenti danseaza in momentul ala
- "id": 3,
"first_name": "er",
"last_name": "er",
"start_time": "2019-11-10T20:26:00Z",
"end_time": "2019-11-10T20:26:03Z",
"vote": 4,
- ce te intereseaza:
 - ◆ start_time si end_time ca sa stii cand danseaza
 - ◆ first_name si last_name
 - ◆ vote este numarul concursului la care votezi (o sa ai nevoie de el la urm pas)
 - ◆ id ai nevoie ca sa stii pe cine votezi

- **localhost:8000/api/note**

- aici dai post
- post-ul are urmatorul format:
- ['contest_id', 'participant_id', 'ritm', 'coregrafie', 'corectitudine', 'componentaArtistica']
- la contest_id pui numarul de la vote de la participant
- la participant_id e id-ul participantului pe care vrei sa il votezi
- si la urmatoarele trebuie sa dai notele de la 1 la 10

=====NOU=====

- **localhost:8000/api/rezultatserie/<id serie>**

- aici dai get
- trebuie ca la finalul fiecarei serii sa se afiseze rezultatele per serie
- fields = ['id', 'first_name', 'last_name', 'nota', 'status'] - asa arata json-ul
- eu ii trimit ordonati descrescator
- asta trebuie sa apara la fiecare serie in parte
- el va oferi datele de mai sus, doar dupa ce au votat toti juratii pe toti concurentii
- daca nu au votat toti, se trimite [] (cred ca e un json gol)

- aici nu e nimic legat de statusul jucatorului, pentru ca nu se elimina nimic

- **localhost:8000/api/rezultat/<id concurs>**

- aici dai get
- dupa ce s-au terminat toate seriile se face clasamentul final
- fields = ['id', 'first_name', 'last_name', 'nota', 'status'] - asa arata json-ul
- eu ii trimit ordonati descrescator
- daca are status=1 inseamna ca avanseaza in runda urmatoare
- daca are status=0 inseamna ca a fost descalificat
- daca nu au votat toti, se trimite [] (cred ca e un json gol)
- altfel apar rezultatele
- eu ii trimit si pe aia cu status=0 (descalificati) ce vor fi mereu la coada
- alegi tu daca vrei sa ii arati si pe cei descalificati

ordinea ca sa se vada corect:

- prima data se acceseaza rezultate serie si apoi rezultate finale
- dupa putem reaccesa rezultatele serie, fie ca am accesat rezultatele finale

USER:

- **user:** Neny **password:** Mancare10100.
- **user:** Ebru **password:** Mancare1010
- **user:** Resul **password:** Mancare1010.

Observatii:

- deci in baza de date avem contest(test1) ce e mai lung si pe care am testat in principal
- are 3 runde
- prima runda are 3 serii a cate 2 participanti adaugati manual de noi acolo din promptul de admin
- a doua runda are 2 serii a cate 2 participanti adaugati automat acolo
- a treia runda are o serie a cate 2 participanti ce se adauga automat