

**Tehnologii și platforme Java pentru aplicații distribuite**  
***Master semestrul 1, 2021-2022:***

Proiect nr. 16

## Teste online

Colectivul:

Dunca Laura  
Suciu Alexandra  
Grupa 244

Aplicația de “Teste online” este alcătuită din două aplicații: una server și una client, plus un engine de Postgres. Un utilizator poate să își genereze un test care conține 5 întrebări, fiecare întrebare având un răspuns de tip Adevărat/Fals, iar la final se afișează rezultatul.

Așadar, acțiunile pe care le poate face un utilizator sunt:

- Generează test
- Răspunde la întrebare
- Treci la următoarea întrebare
- Finalizează test

Tehnologiile utilizate în aplicația server sunt:

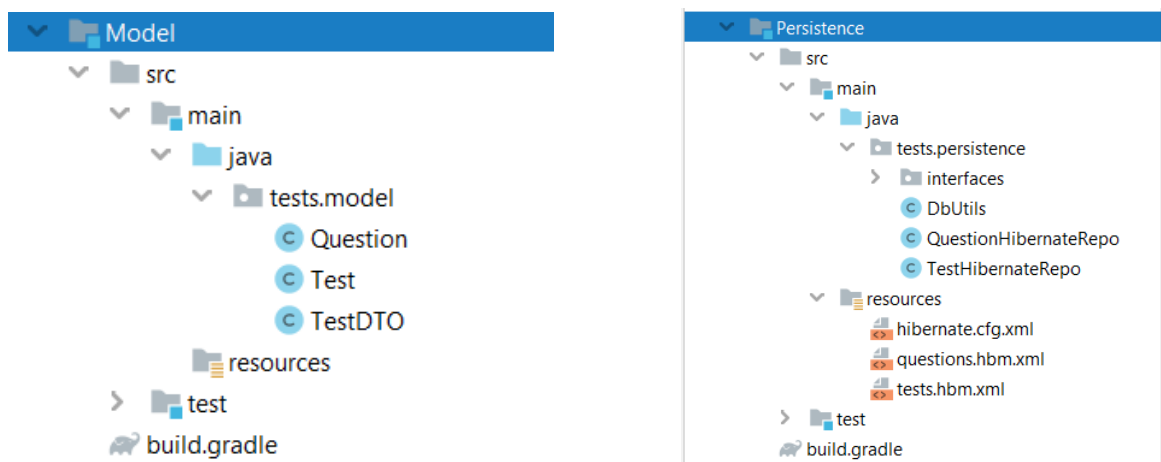
- Gradle 6.2
- Spring Boot (Starter data JPA) 2.6.3
- JDK 11
- Hibernate

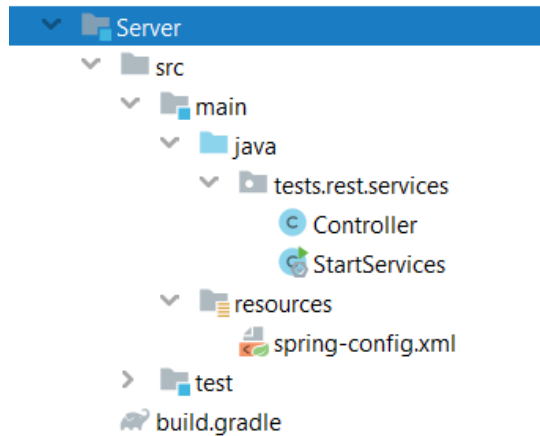
Tehnologiile utilizate în aplicația client sunt:

- Angular 11.2.3
- Node 16.13
- Typescript 4.1.5
- Material UI 11.2.0

## Serverul

Serverul este alcătuit din mai multe proiecte Gradle, care comunică între ele și sunt dispuse pe layere: modele, persistență și server.





Proiectul **Model** conține clasele modelate în aplicație:

- Question(ID, description, answer) – modelează o întrebare cu cerință și răspuns true/false.
- Test(ID, q1, q2, q3, q4, q5) – modelează un test care conține referințe către cele 5 întrebări.
- TestDTO(testId, questions) – modelează obiectul ce va fi trimis către client la generarea testului.

Proiectul **Persistence** conține clasele modelate pentru persistență și comunicarea cu baza de date:

- DbUtils – conține obiecte de tip Properties și Connection și se ocupă de furnizarea unei conexiuni către baza de date.
- QuestionHibernateRepo – conține metode care fac apeluri către baza de date pe tabela pentru întrebări; folosește DbUtils.
- TestHibernateRepo – conține metode care fac apeluri către baza de date pe tabela pentru teste; folosește DbUtils.
- In folderul resources se află fișierele XML pentru configurările ce țin de Hibernate, modelarea tabelor și a relațiilor dintre ele.

Proiectul **Server** conține:

- Controllerul Rest care primește apelurile REST de la client și comunică cu clasele de repo din Persistence
- StartServices pornește aplicația SpringBoot.
- În folderul resources se află fișierul XML pentru configurările ce țin de injectările beanurilor Spring și proprietățile injectate pentru baza de date.

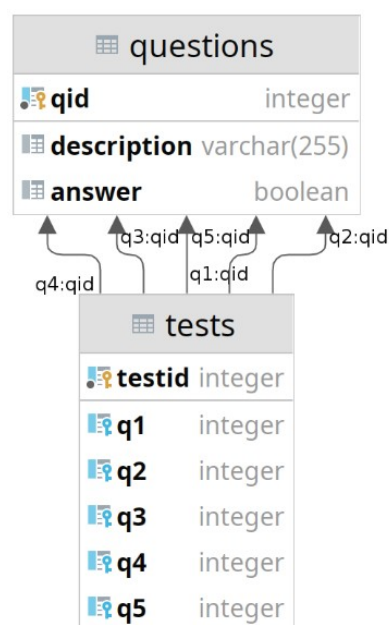


## Baza de date

Pentru a porni baza de date s-a rulat comanda *docker-compose up* pe un fișier YAML care pornește un serviciu docker ce încapsulează engine-ul de Postgres.

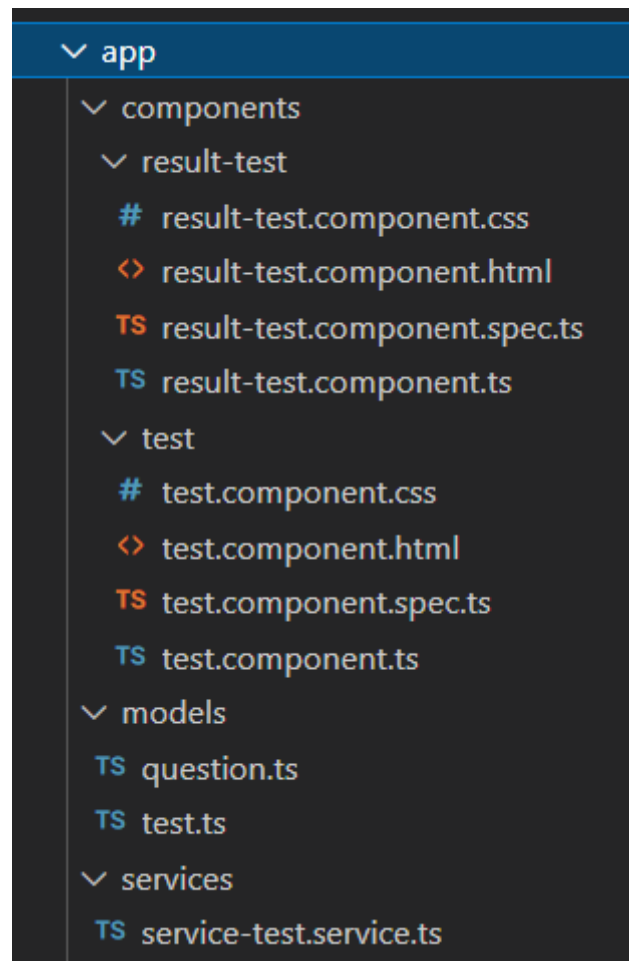
```
C:\Users\alexa\Downloads>docker-compose up
[+] Running 2/2
 - Container downloads-postgres-1 Recreated
 - Container downloads-pgadmin-1 Recreated
Attaching to downloads-pgadmin-1, downloads-postgres-1
downloads-postgres-1 | PostgreSQL Database directory appears to contain a database; Skipping initialization
downloads-postgres-1 |
downloads-postgres-1 | 2022-01-25 07:33:57.538 UTC [1] LOG: starting PostgreSQL 12.3 on x86_64-pc-linux-musl, compiled by gcc (Alpine 9.3.0) 9.3.0, 64-bit
downloads-postgres-1 | 2022-01-25 07:33:57.538 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
downloads-postgres-1 | 2022-01-25 07:33:57.538 UTC [1] LOG: listening on IPv6 address ":::", port 5432
downloads-postgres-1 | 2022-01-25 07:33:57.548 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
downloads-postgres-1 | 2022-01-25 07:33:57.569 UTC [21] LOG: database system was shut down at 2022-01-25 07:33:51 UTC
downloads-postgres-1 | 2022-01-25 07:33:57.574 UTC [1] LOG: database system is ready to accept connections
downloads-pgadmin-1 | sudo: setrlimit(RLIMIT_CORE): Operation not permitted
downloads-pgadmin-1 | [2022-01-25 07:34:02 +0000] [1] [INFO] Starting gunicorn 19.9.0
downloads-pgadmin-1 | [2022-01-25 07:34:02 +0000] [1] [INFO] Listening at: http://[::]:80 (1)
downloads-pgadmin-1 | [2022-01-25 07:34:02 +0000] [1] [INFO] Using worker: threads
downloads-pgadmin-1 | /usr/local/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, the default buffer size will be used
downloads-pgadmin-1 | return io.open(fd, *args, **kwargs)
downloads-pgadmin-1 | [2022-01-25 07:34:02 +0000] [88] [INFO] Booting worker with pid: 88
```

De asemenea, baza de date este creată Code-First, după care aceasta doar se updatează.



## Client

Clientul este realizat pe baza frameworkului Angular, fiind alcătuit din 2 componente (componenta de test și componenta result), un service și modelele utilizate.



Serviceul conține două metode care modelează prin HttpClient cererile la server, două metode POST: `getTest` care trimite numărul testului și primește testul corespunzător, și `sendAnswers` care trimite o listă cu răspunsurile utilizatorului și primește rezultatul obținut de acesta. Pentru ușurința conturării designului am folosit componentele din Material UI.

Componenta Test generează un număr random în momentul în care utilizatorul dorește să genereze un test, număr care va reprezenta id ul testului primit de la server.

Generate test number:

După ce acesta generează un număr se așteaptă 3 secunde pentru primirea răspunsului.

Generate test number:

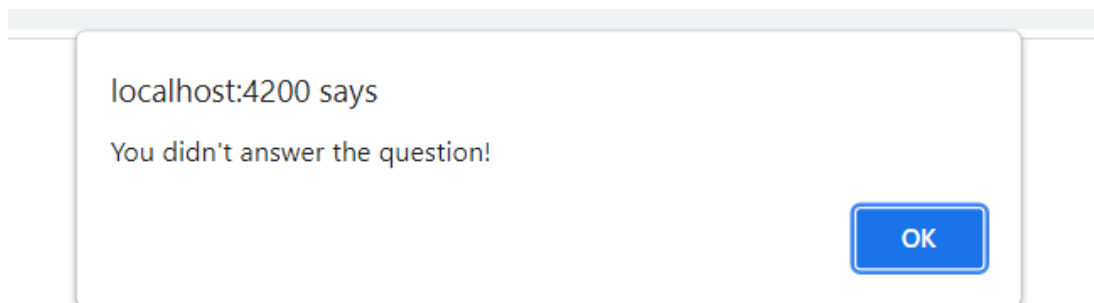
Răspunsul va conține o listă de întrebări corespunzătoare testului generat care vor fi afișate într-un mat-card împreună cu 2 butoane ca variante de răspuns (Adevărat și Fals).

Generate test number:

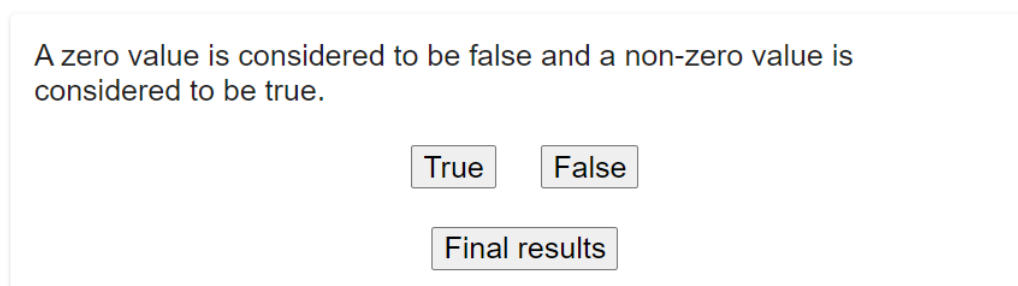
 

It is necessary that a loop counter must only be an int. It cannot be a float.

Dacă utilizatorul dorește să treacă la următoarea întrebare fără să răspundă la aceasta, primește un mesaj de alert obligându-l astfel să răspundă la întrebare și să nu poată trece mai departe.



Când a ajuns la ultima întrebare apare butonul de finalizare al testului, care de asemenea afișează un mesaj de alert dacă nu s-a răspuns la întrebare, în caz contrar se trimite la server lista cu rezultatele întrebărilor și se redirectează către pagina de result, unde componenta Result afișează numărul de răspunsuri corecte.



Rezultatul primit este depus în session storage de unde este apoi preluat.

**Result: 1/5**