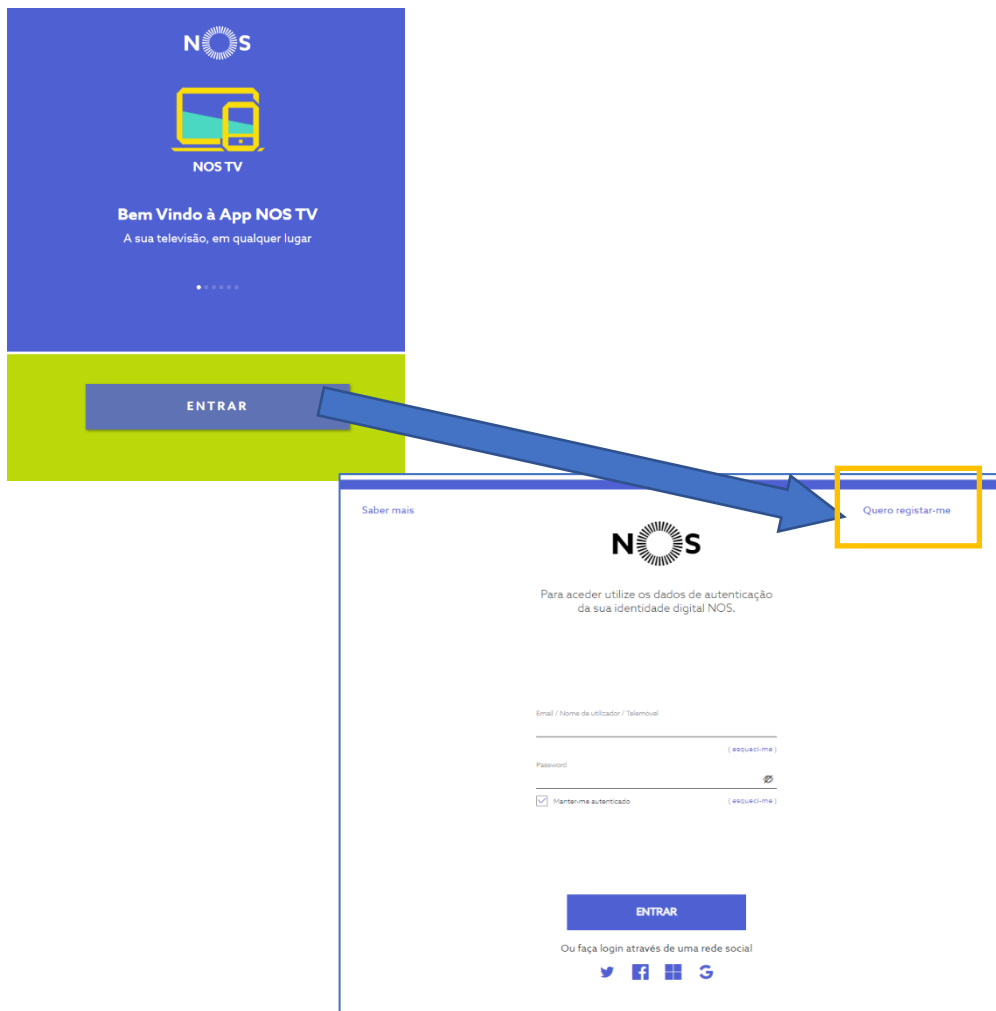




## Automation Testing Questions

### 1. Registration flow tests automation

Access to [nostv.pt](http://nostv.pt) , select “ENTRAR” option and entry page will be available with login/registration options. Implement the automation tests for registration (“Quero registar-me”) flow, using Selenium, Katalon or other tool. The tool can be freely choose.



NOTE: Registration flow finishes when confirmation codes sent for each contact are validated. To help this implementation you can get the codes manually and fill inputs in automation tests.

## 2. Login flow tests automation

Using the user registered above, access again to nostv.pt and automate the login flow.

[Saber mais](#)

[Quero registar-me](#)



Para aceder utilize os dados de autenticação da sua identidade digital NOS.

Email / Nome de utilizador / Telemóvel

(esqueci-me)

Password

☒ Manter-me autenticado

(esqueci-me)

ENTRAR

Ou faça login através de uma rede social



## 3. API Tests

The next examples of REST API requests describes the CRUD operations for create, update and delete an user with the following model:

```
{
  "_id": "1234567890",
  "name": "User Name",
  "age": "25"
}
```

For each operation, propose a response example and the sequence of assertions to validate the result.

- **Create new user request**

```
POST /api/1.0/users

Content-Type: application/json

{
  "name": "User Name",
  "age": "25"
}
```

**Response example and assertions for create user request?**

- **Update user request**

```
PUT /api/1.0/users/1234567890

Content-Type: application/json

{
  "name": "User Name New",
  "age": "25"
}
```

**Response example and assertions for create user request?**

- **Delete user request**

```
DELETE /api/1.0/users/1234567890
```

**Response example and assertions for create user request?**

**Example for get user request**

```
GET /api/1.0/users/1234567890
```

Response:

```
200 OK

{
  "_id": "1234567890",
  "name": "User Name",
  "age": "25"
}
```

Assertions:

- statusCode = 200
- user schema validation is ok
- user id is equals to request id = 1234567890
- name and age fields are not empty.