

Pruebas para el formulario de Registro

Test 1: Se prueba que el formulario no permita enviarse si tiene datos inválidos, además se prueba que la detección de coincidencia de contraseña funcione correctamente.

- Se mockea nombre, email y contraseñas con datos erróneos, además se asignan los términos y condiciones en aceptados.
- Se mockea la función `isInvalid` para que devuelva un `true`.
- Se guardan los elementos que contienen el mensaje de error para el email y `password2`
- Se prueba que el formulario sea inválido.
- Se prueba que el mensaje de email incorrecto se esté mostrando correctamente.
- Se testea que el mensaje de que las contraseñas no coinciden se esté mostrando en correctamente.

Test 2: Se prueba que el formulario no permita enviarse si tiene datos inválidos, además de que se tengan que aceptar los términos y condiciones.

- Se mockea nombre, email y contraseñas con datos correctos y se dejan sin aceptar los términos y condiciones.
- Se mockea la función `isInvalid` para que devuelva un `true`.
- Se guardan los elementos que contienen el mensaje de error para el email y `password2`
- Se prueba que el formulario sea inválido.
- Se prueba que el mensaje de que se deben aceptar los términos y condiciones se esté mostrando correctamente.

Test 3: Ejecutar solicitud HTTP y que sea exitosa.

- Se mockea nombre, email y contraseña.
- Se mockea la solicitud HTTP para que devuelva un observable con valor exitoso.
- Se prueba que el servicio que realiza la solicitud este definido.

Test 4: Ejecutar solicitud HTTP y que sea incorrecta.

- Se mockea nombre, email y contraseña.
- Se mockea la solicitud HTTP para que devuelva un observable con valor fallido.
- Se prueba que el servicio que realiza la solicitud este definido.