

Historia 1 – Formulario de Contacto

Campos obligatorios validados.

1. Email válido requerido.
2. Mensaje mínimo 10 caracteres.
3. Respuesta 200/201 al enviar correctamente.
4. Error controlado en campos incorrectos.

The screenshot shows the Postman interface for a workspace named 'stageGreentex'. The selected collection is 'Formulario de contacto EXITOSO'. The request is a POST to '[[base_url]]/[[contact_endpoint]]' with a raw JSON body. The body contains contact information for 'pedro', 'López', 'Sustainable Fabrica', including a phone number and email. The response is a 200 OK status with a response time of 1.57s and a body size of 66.82 KB. The test results show that all required fields are present, and the email, phone number, and location are valid.

```
1 {
2   "firstName": "pedro",
3   "lastName": "López",
4   "interestedIn": "Sustainable Fabrica",
5   "company": "pruebas",
6   "phoneNumber": "+57 3861234567",
7   "email": "red@example.com",
8   "location": "South America",
9   "message": "Estoy interesado en conocer más detalles sobre sus productos y tiempos de entrega."
10 }
```

Test Results (10/12):

- PASSED All required fields are present
- FAILED Email has valid format [AssertionError: expected 'red@example.com' to match /^(?!(\w|@|+|_|\$)/
- PASSED Phone number has valid format
- PASSED First name is not empty
- PASSED Last name is not empty
- PASSED Email has appropriate length
- PASSED Phone number has appropriate length
- PASSED Location is from valid list

Historia 2 – Listado y selección de producto

1. El servicio debe devolver status 200 OK.
2. El tiempo de respuesta debe ser menor a 3000 ms.
3. El encabezado Content-Type debe estar presente e indicar contenido HTML.

The screenshot shows the Postman interface for a workspace named 'stageGreentex'. The selected collection is 'Lista de productos OK'. The request is a POST to '[[base_url]]/[[products_list]]'. The test script includes assertions for the status code (200 OK), response time (under 3000ms), and the presence of the 'Content-Type' header (text/html). The response is a 200 OK status with a response time of 2.23s and a body size of 79.1 KB. The test results show that all assertions passed.

```
1 // 1. Debe responder 200 OK
2 pm.test("Status code is 200 OK", function () {
3   pm.response.to.have.status(200);
4 });
5
6 // 2. Tiempo de respuesta < 3000 ms
7 pm.test("Response time is under 3000ms", function () {
8   pm.expect(pm.response.responseTime).to.be.below(4000);
9 });
10
11 // 3. Content-Type debe existir y ser HTML
12 pm.test("Content-Type header is present and is HTML", function () {
13   const contentType = pm.response.headers.get("Content-Type");
14   pm.expect(contentType).to.exist;
15   pm.expect(contentType.toLowerCase()).to.include("text/html");
16 });
17
```

Test Results (3/3):

- PASSED Status code is 200 OK
- PASSED Response time is under 3000ms
- PASSED Content-Type header is present and is HTML

Historia 3 – Listado y selección de producto invalido

1. Si la ruta, el endpoint o el recurso está mal construido, debe retornar un error controlado (400 o 404), NO un 200 con página HTML genérica.
2. El tiempo de respuesta NO debe exceder los 1000 ms.
Si supera ese tiempo, se considera degradación de servicio.
3. El encabezado Content-Type NO debe indicar HTML cuando se espera una API.

The screenshot displays the Postman interface for a workspace named 'My Workspace'. The left sidebar shows a collection of API tests under the 'stageGreentex' folder, including 'POST Lista de productos Fallido', which is currently selected. The main panel shows the details of this test, which is a POST request to '[[base_url]]/products_list'. The test script contains three assertions: 1. The status code is 200 OK. 2. The response time is under 3000ms. 3. The Content-Type header is present and is HTML. The test results at the bottom show three failed assertions: 'Status code is 200 OK | AssertionError: expected response to have status code 400 but got 200', 'Response time is under 3000ms | AssertionError: expected 2035 to be below 1000', and 'Content-Type header is present and is HTML | Error: Invalid Chai property: exclude. Did you mean "include"?'. The interface also shows the 'Send' button and various tabs for the request details.