

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 a Postman collection named 'stageGreentex' with several test cases. One test case, 'Formulario de contacto EXITOSO', is highlighted. It sends a POST request to '({base_url})/({contact_endpoint})' with a JSON body containing required fields like 'firstName', 'lastName', and 'interestedIn'. The response is a 200 OK status with a message indicating success.

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 a Postman collection named 'stageGreentex' with several test cases. One test case, 'Lista de productos OK', is highlighted. It sends a POST request to '({base_url})/({products_list})' with an empty body. The response status is 200 OK, indicating a successful list retrieval.

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 shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' containing various collections and environments. The main area displays a test collection named 'stageGreentex / Lista de productos Fallido'. A specific test case for a POST request to '{{base_url}}/((products_Url))' is selected. The 'Tests' tab contains the following Chai assertions:

```
// 1. Debe responder 200 OK
pm.test("Status code is 200 OK", function () {
  expect(pm.response.to.have.status(200));
});

// 2. Tiempo de respuesta < 3000 ms
pm.test("Response time is under 3000ms", function () {
  expect(pm.response.responseTime).to.be.below(3000);
});

// 3. Content-Type debe existir y ser HTML
pm.test("Content-Type header is present and is HTML", function () {
  const contentType = pm.response.headers.get("Content-Type");
  expect(contentType).to.exist;
  expect(contentType.toLowerCase()).to.exclude("text/html");
});
```

The 'Test Results' section shows three failed assertions:

- FAILED Status code is 200 OK | AssertionError: expected response to have status code 400 but got 200
- FAILED Response time is under 3000ms | AssertionError: expected 2035 to be below 1000
- FAILED Content-Type header is present and is HTML | Error: Invalid Chai property: exclude. Did you mean "include"? | Fix test

At the bottom, the status bar indicates '200 OK - 2.04 s - 78.98 KB'.