

17	Verificar que el sistema rechaza la solicitud cuando el cuerpo de la solicitud no contenga la estructura correcta por ejemplo falta el campo id y devuelve el código de error apropiado	Path params: id=7 { "productsList": [{ "quantity": 1 }] }	1. El usuario selecciona el método POST y hace una solicitud a: URL + api/v1/kits/:id/products 2. El usuario envía una petición sin un ID de producto existente en los parámetros y el cuerpo de la solicitud, y con valor 1 en el campo quantity 3. No se añade productos existentes al kit. 4. El usuario obtiene un código de estado: error y un mensaje.	Código de respuesta: 400 Bad Request Mensaje de error que indique que Falta el Producto El producto no se agrega al kit	Código de respuesta: 200 OK El producto se agrega al kit con cantidad: 1 y sin información del producto.	Failed	https://axelburrito.atlassian.net/browse/UG-12
18	Verificar que el sistema rechaza la solicitud cuando el cuerpo de la solicitud no contenga la estructura correcta por ejemplo falta el campo quantity y devuelve el código de error apropiado	Path params: id=7 { "productsList": [{ "id": 34 }] }	1. El usuario selecciona el método POST y hace una solicitud a: URL + api/v1/kits/:id/products 2. El usuario envía una petición usando un ID de producto existente en los parámetros y el cuerpo de la solicitud, y sin el campo quantity 3. No se añade productos existentes al kit. 4. El usuario obtiene un código de estado: error y un mensaje.	Código de respuesta: 400 Bad Request Mensaje de error que indique que quantity falta El producto no se agrega al kit	Código de respuesta: 500 Internal Server Error Mensaje : "invalid input syntax for type integer: \"NaN\"" El producto no se agrega al kit	Failed	https://axelburrito.atlassian.net/browse/UG-13
19	Verificar que el sistema rechaza la solicitud cuando el cuerpo de la solicitud no contenga la estructura correcta por ejemplo productsList no es un array	Path params: id=7 { "productsList": [{ "id": 34, "quantity": 1 }] }	1. El usuario selecciona el método POST y hace una solicitud a: URL + api/v1/kits/:id/products 2. El usuario envía una petición usando un ID de producto existente en los parámetros y quantity pero el campo productsList no es un array 3. No se añade productos existentes al kit. 4. El usuario obtiene un código de estado: error y un mensaje.	Código de respuesta: 400 Bad Request Mensaje de error El producto no se agrega al kit	Código de respuesta: 500 Internal Server Error Mensaje : "productsList.find is not a function" El producto no se agrega al kit	Failed	https://axelburrito.atlassian.net/browse/UG-14
	Verificar que el sistema rechaza la solicitud cuando el cuerpo de la solicitud contiene id duplicados	Path params: id=7 { "productsList": [{ "id": 1, "quantity": 2 }, { "id": 1, "quantity": 2 }] }	1. El usuario selecciona el método POST y hace una solicitud a: URL + api/v1/kits/:id/products 2. El usuario envía una petición usando un ID de producto existente en los parámetros y id duplicado 3. No se añade productos existentes al kit. 4. El usuario obtiene un código de estado: error y un mensaje.	{ "code": 400, "message": "Duplicate product id: 1" }	Código 200 OK { "id": 7, "name": "Kit de Prueba", "productsList": [{ "id": 1, "name": "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free", "price": 2, "weight": 473, "units": "ml", "quantity": 2 }], "productsCount": 2 }	Failed	https://axelburrito.atlassian.net/browse/UG-15
REQUISITO 2							
1	Verificar que el servicio Order and Go está disponible cuando se cumplen los tres requisitos obligatorios.	{ "deliveryTime": 9, "productsCount": 10, "productsWeight": 6 }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando los campos obligatorios en el body, todos de tipo numérico 3. La respuesta muestra el tiempo de entrega, el costo de entrega y da un código 200 OK	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 5, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 5 }	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 5, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 5 }	Passed	
2	Verificar que el servicio Order and Go no está disponible cuando le falta el body en la solicitud	{ }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando en el body sin los campos obligatorios 3. La respuesta muestra el código 400 bad request	Código de respuesta: 400 Bad Request Mensaje de error No se genera una orden	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 3, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 0 }	Failed	https://axelburrito.atlassian.net/browse/SEUG-1
3	Verificar que el servicio Order and Go no está disponible cuando se falta el campo deliveryTime.	{"productsCount": 10, "productsWeight": 6 }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando en el body sin el campo deliveryTime 3. La respuesta muestra el código 400 bad request	Código de respuesta: 400 Bad Request Mensaje de error No se genera una orden	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 5, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 0 }	Failed	https://axelburrito.atlassian.net/browse/SEUG-2
4	Verificar que el servicio Order and Go no está disponible cuando se falta el campo productsCount	{"deliveryTime": 9, "productsWeight": 6 }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando en el body sin el campo productsCount 3. La respuesta muestra el código 400 bad request	Código de respuesta: 400 Bad Request Mensaje de error No se genera una orden	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 3, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 0 }	Failed	https://axelburrito.atlassian.net/browse/SEUG-3
5	Verificar que el servicio Order and Go no está disponible cuando se falta el campo productsWeight	{"deliveryTime": 9, "productsCount": 10, }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando en el body sin el campo productsWeight 3. La respuesta muestra el código 400 bad request	Código de respuesta: 400 Bad Request Mensaje de error No se genera una orden	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 3, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 0 }	Failed	https://axelburrito.atlassian.net/browse/SEUG-4
6	Verificar que el servicio de entrega devuelva un false cuando esta por fuera de la hora de servicio ejemplo 7	{"deliveryTime": 7, "productsCount": 10, "productsWeight": 6 }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando los campos obligatorios en el body, todos de tipo numérico con valor en el campo deliveryTime = 7 3. La respuesta muestra un codigo 200 OK	{ "name": "Order and Go", "isItPossibleToDeliver": false, "hostDeliveryCost": 5, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 5 }	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 5, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 5 }	Failed	https://axelburrito.atlassian.net/browse/SEUG-5
7	Verificar que el servicio de entrega devuelva un "isItPossibleToDeliver":true , hostDeliveryCost=3 y clientDeliveryCost=0 cuando se cumplen las condiciones	{"deliveryTime": 8, "productsCount": 1, "productsWeight": 0.1 }	1. El usuario selecciona el método POST y hace una solicitud a: URL + /order-and-go/v1/delivery 2. El usuario envía una petición usando los campos obligatorios en el body, todos de tipo numérico 3. La respuesta muestra un codigo 200 OK	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 3, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 0 }	{ "name": "Order and Go", "isItPossibleToDeliver": true, "hostDeliveryCost": 3, "toBeDeliveredTime": { "min": 20, "max": 25 }, "clientDeliveryCost": 0 }	Passed	

