Universidad ORT Uruguay

Facultad de Ingeniería

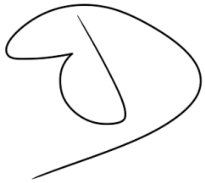
OBLIGATORIO 1

DISEÑO DE APLICACIONES 2

-

Evidencia de la ejecución de las pruebas de la API con Postman

Nicolás Edelman
251363



Juan Diego Etcheverry
252443

Mateo Goldwasser
239420

Tutores: Francisco Bouza, Juan Irabedra, Santiago Tonarelli

2023

Indice:

Video: **3**
USER: **3**
Signup: 3
Update: 4
Logout: 5
Login: 6
ADMIN: **7**
Create user: 7
Update user: 8
Delete user: 9
Purchase History: 10
Promotions: **11**

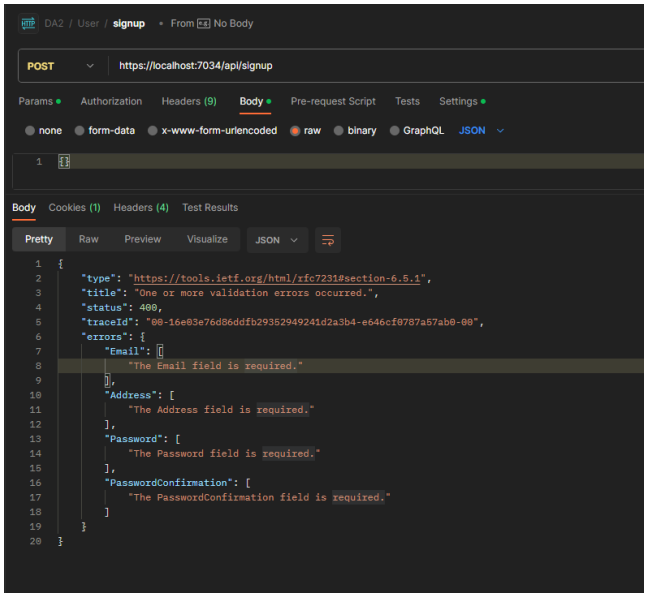
Link al repositorio:
https://github.com/IngSoft-DA2-2023-2/239420-252443-251363

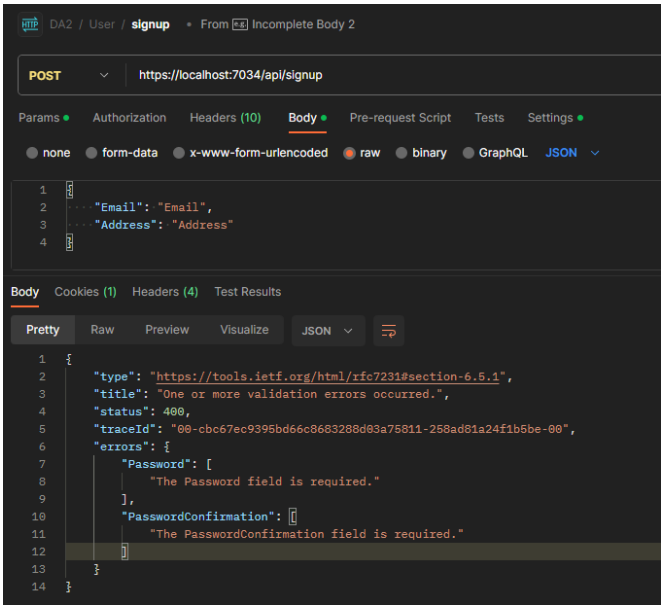
Video:

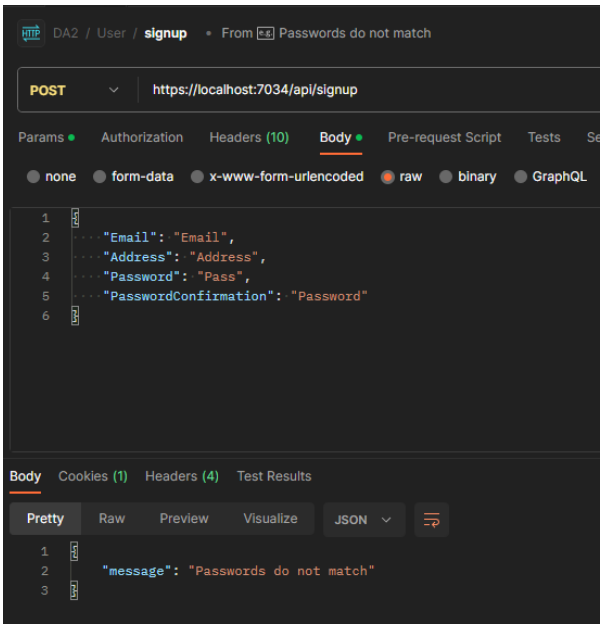
https://www.youtube.com/watch?v=elMXH3i1h9o

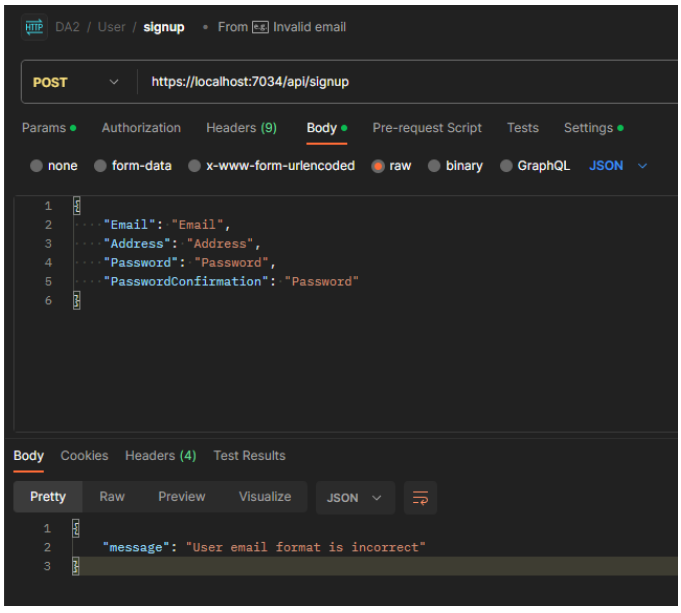
USER:

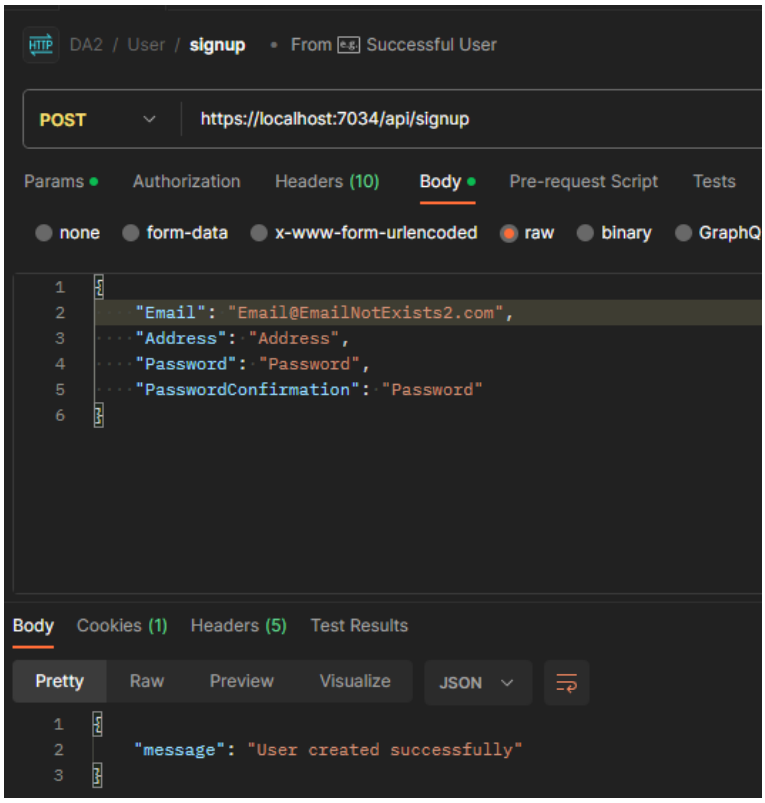
Signup:



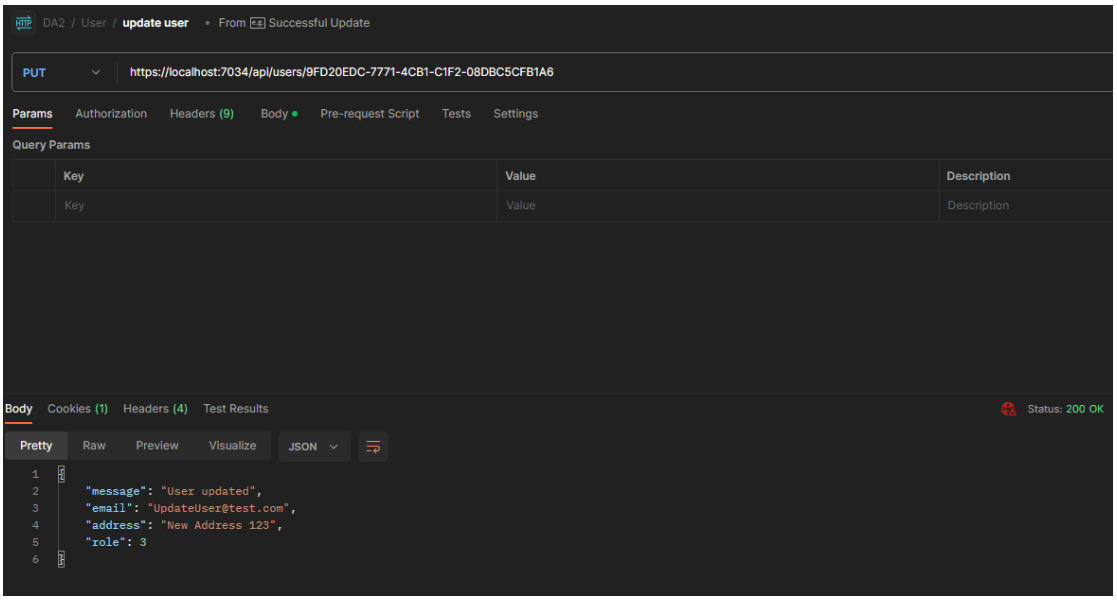




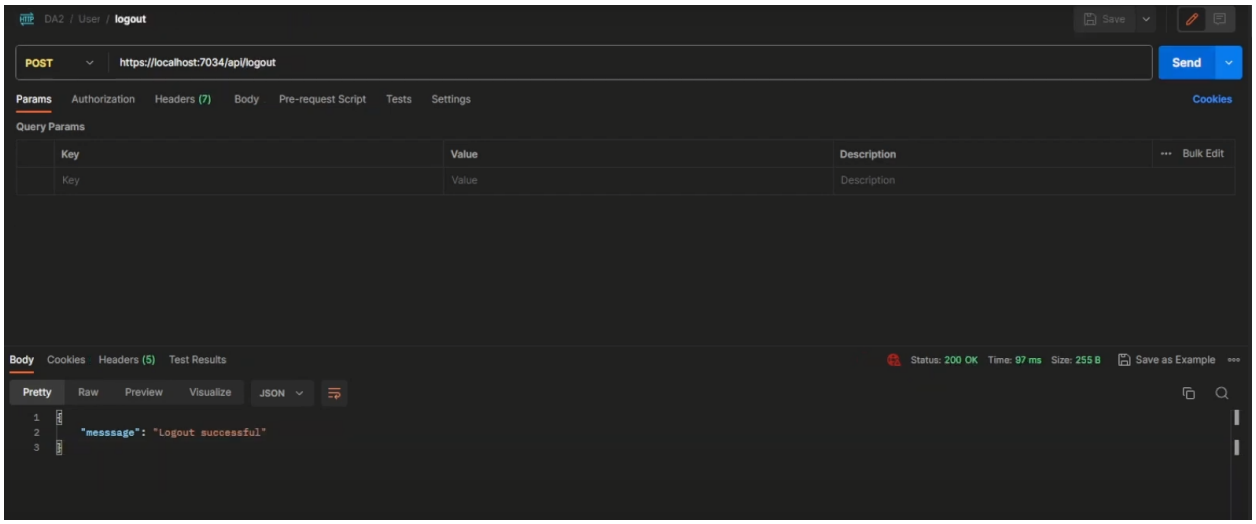



DA2 / User / signup • From Successful User
POST
https://localhost:7034/api/signup
Params • Authorization Headers (10) Body • Pre-request Script Tests
none form-data x-www-form-urlencoded raw binary GraphQ
{
"Email": "Email@EmailNotExists2.com",
"Address": "Address",
"Password": "Password",
"PasswordConfirmation": "Password"
}
Body Cookies (1) Headers (5) Test Results
Pretty Raw Preview Visualize JSON
{
"message": "User created successfully"
}

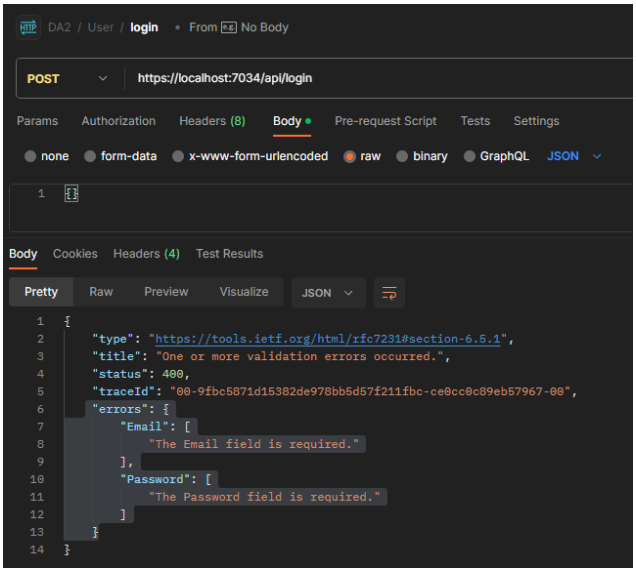
Update:

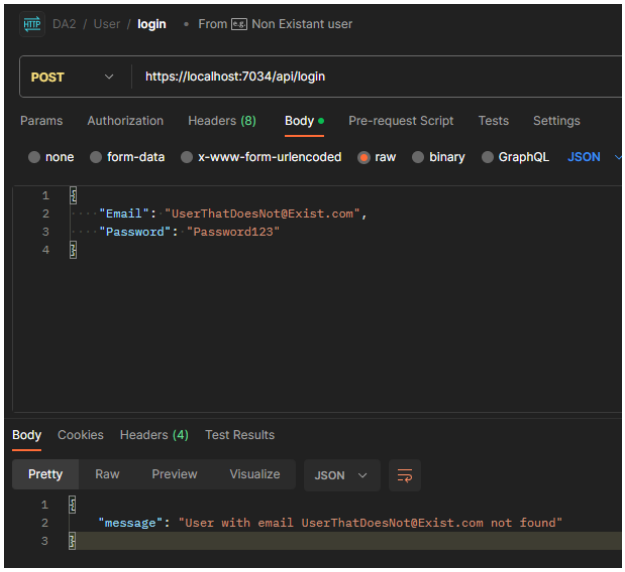

DA2 / User / update user • From Successful Update
PUT
https://localhost:7034/api/users/9FD20EDC-7771-4CB1-C1F2-08DBC5CFB1A6
Params Authorization Headers (9) Body • Pre-request Script Tests Settings
Query Params
Key Value Description
Key Value Description
Body Cookies (1) Headers (4) Test Results
Status: 200 OK
Pretty Raw Preview Visualize JSON
{
"message": "User updated",
"email": "UpdateUser@test.com",
"address": "New Address 123",
"role": 3
}

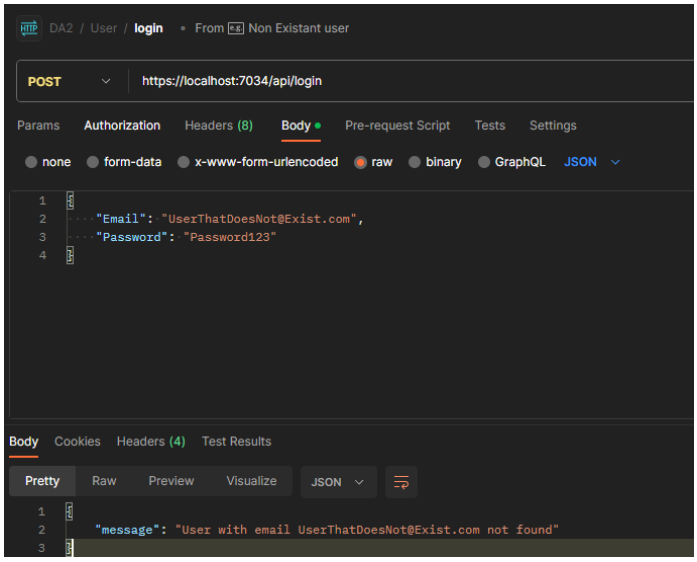
Logout:

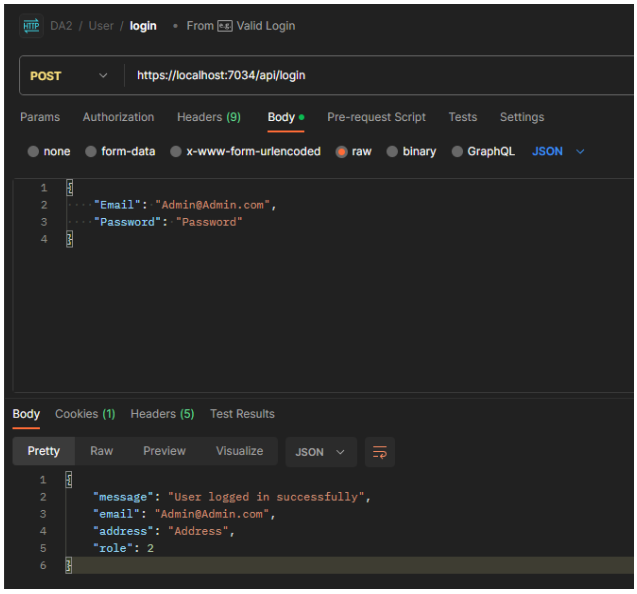

DA2 / User / logout
Save
POST
https://localhost:7034/api/logout
Send
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
Key
Value
Description
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 97 ms
Size: 255 B
Save as Example
Pretty
Raw
Preview
Visualize
JSON
1 {
2 "messsage": "Logout successful"
3 }

Login:


DA2 / User / login • From No Body
POST
https://localhost:7034/api/login
Params Authorization Headers (8) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Body Cookies Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"type": "https://tools.ietf.org/html/rfc7231#section-6.5.1",
"title": "One or more validation errors occurred.",
"status": 400,
"traceId": "00-9fbc5871d15382de978bb5d57f211fbc-ce0cc0c89eb57967-00",
"errors": {
"Email": [
"The Email field is required."
],
"Password": [
"The Password field is required."
]
}
}

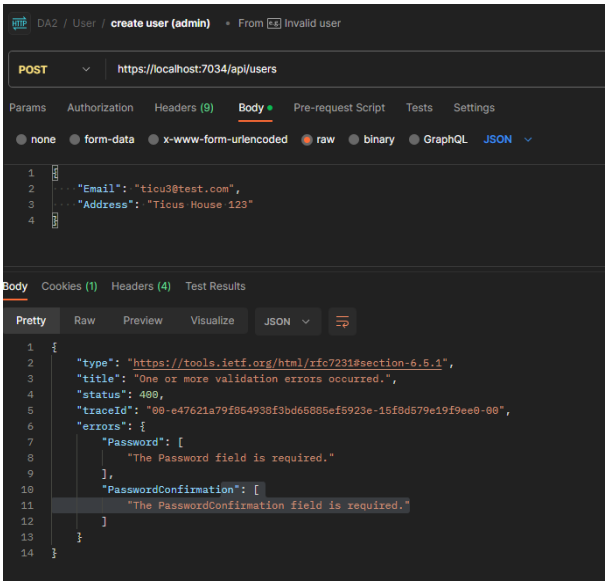

DA2 / User / login • From Non Existant user
POST
https://localhost:7034/api/login
Params Authorization Headers (8) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"Email": "UserThatDoesNot@Exist.com",
"Password": "Password123"
}
Body Cookies Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"message": "User with email UserThatDoesNot@Exist.com not found"
}

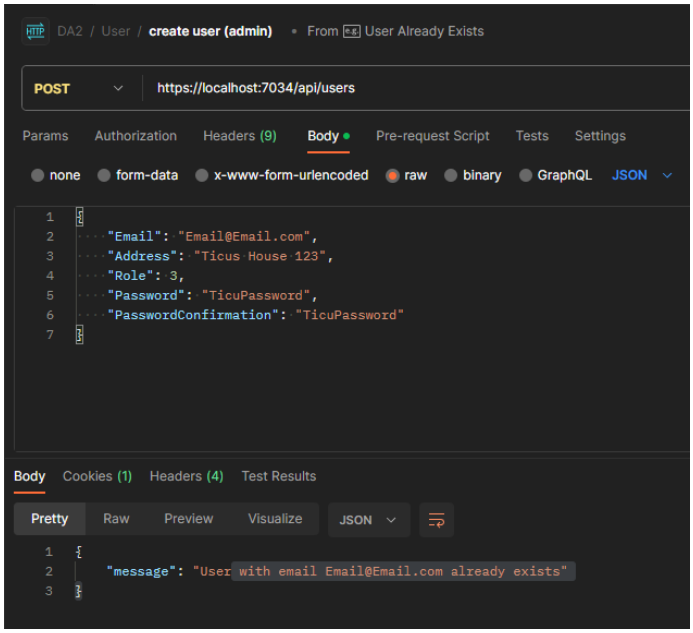

DA2 / User / login • From Non Existant user
POST
https://localhost:7034/api/login
Params Authorization Headers (8) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"Email": "UserThatDoesNot@Exist.com",
"Password": "Password123"
}
Body Cookies Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"message": "User with email UserThatDoesNot@Exist.com not found"
}


DA2 / User / login • From Valid Login
POST
https://localhost:7034/api/login
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"Email": "Admin@Admin.com",
"Password": "Password"
}
Body Cookies (1) Headers (5) Test Results
Pretty Raw Preview Visualize JSON
{
"message": "User logged in successfully",
"email": "Admin@Admin.com",
"address": "Address",
"role": 2
}

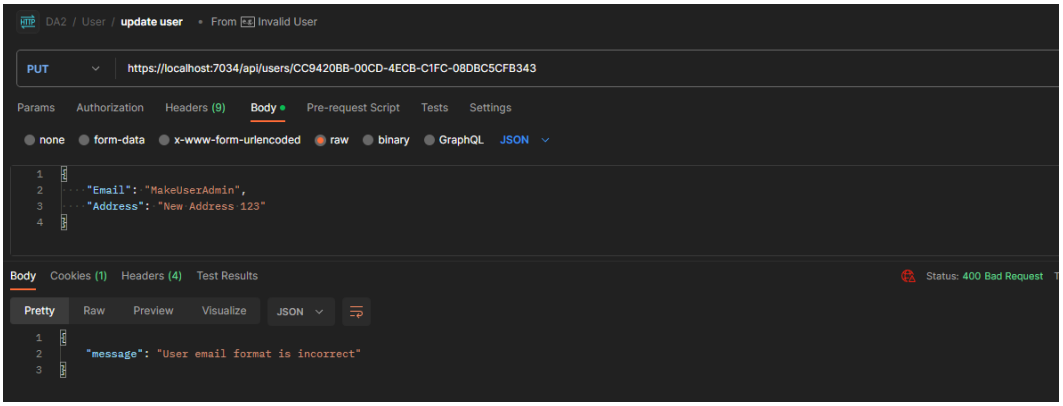
ADMIN:

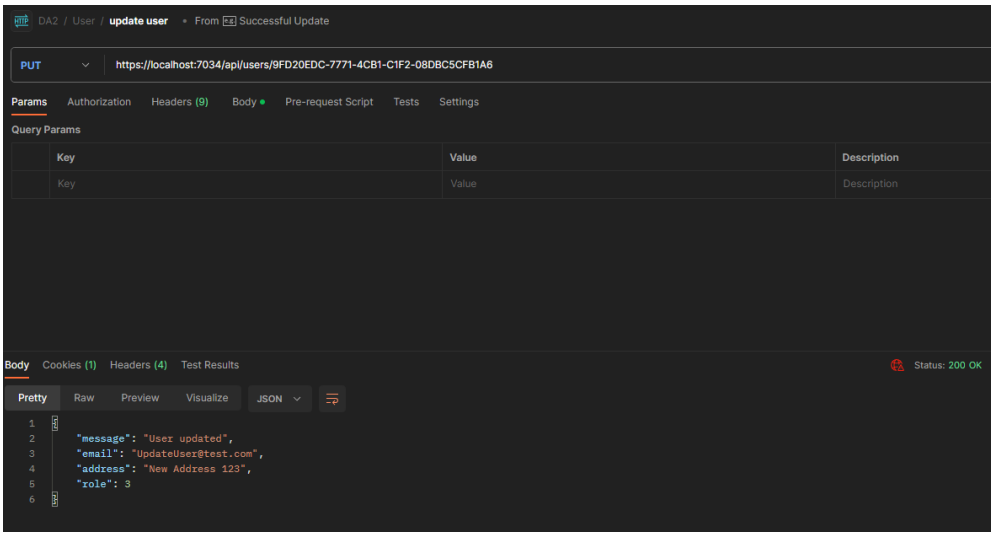
Create user:

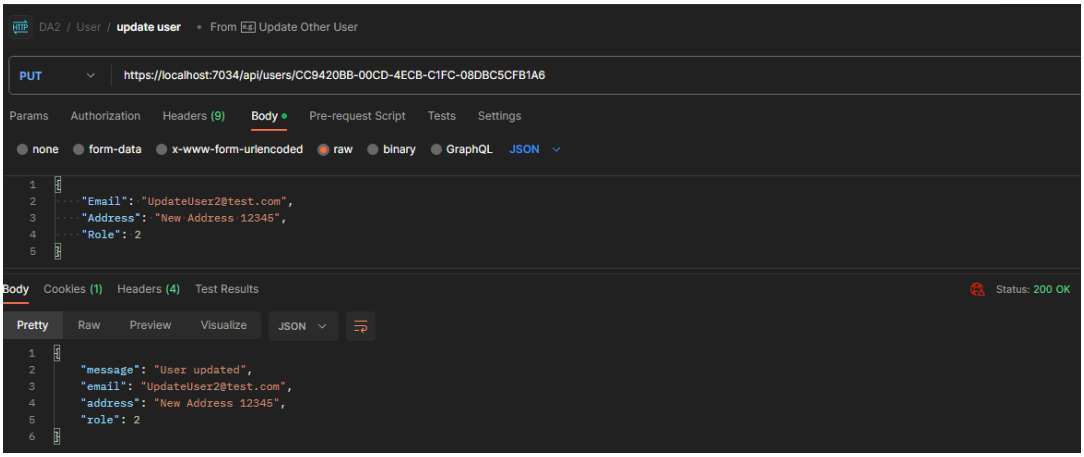




Update user:

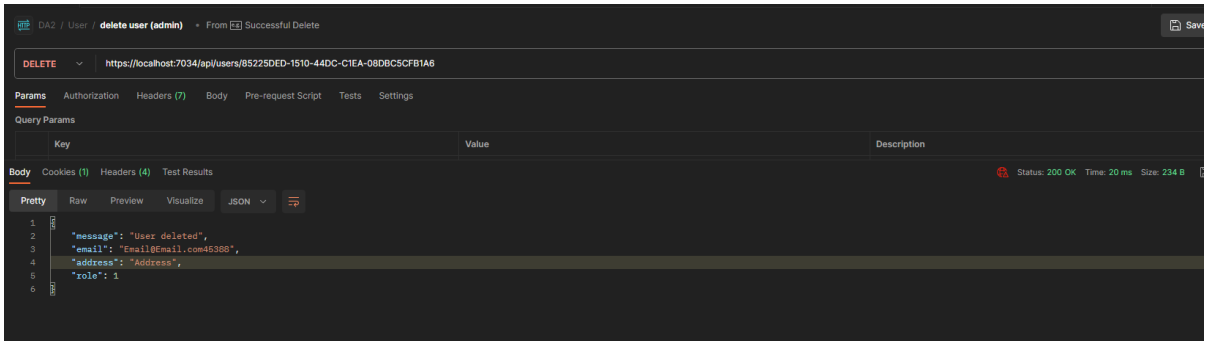

DA2 / User / update user • From Invalid User
PUT
https://localhost:7034/api/users/CC9420BB-00CD-4ECB-C1FC-08DBC5CFB343
Params Authorization Headers (9) Body • Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"Email": "MakeUserAdmin",
"Address": "New Address 123"
}
Body Cookies (1) Headers (4) Test Results
Status: 400 Bad Request
Pretty Raw Preview Visualize JSON
{
"message": "User email format is incorrect"
}


DA2 / User / update user • From Successful Update
PUT
https://localhost:7034/api/users/9FD20EDC-7771-4CB1-C1F2-08DBC5CFB1A6
Params Authorization Headers (9) Body • Pre-request Script Tests Settings
Query Params
Key Value Description
Key Value Description
Body Cookies (1) Headers (4) Test Results
Status: 200 OK
Pretty Raw Preview Visualize JSON
{
"message": "User updated",
"email": "UpdateUser@test.com",
"address": "New Address 123",
"role": 3
}

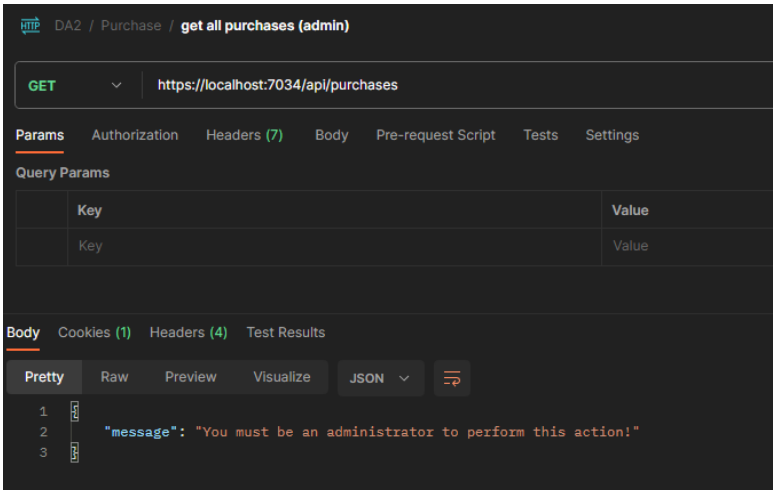

DA2 / User / update user • From Update Other User
PUT
https://localhost:7034/api/users/CC9420BB-00CD-4ECB-C1FC-08DBC5CFB1A6
Params Authorization Headers (9) Body • Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"Email": "UpdateUser2@test.com",
"Address": "New Address 12345",
"Role": 2
}
Body Cookies (1) Headers (4) Test Results
Status: 200 OK
Pretty Raw Preview Visualize JSON
{
"message": "User updated",
"email": "UpdateUser2@test.com",
"address": "New Address 12345",
"role": 2
}

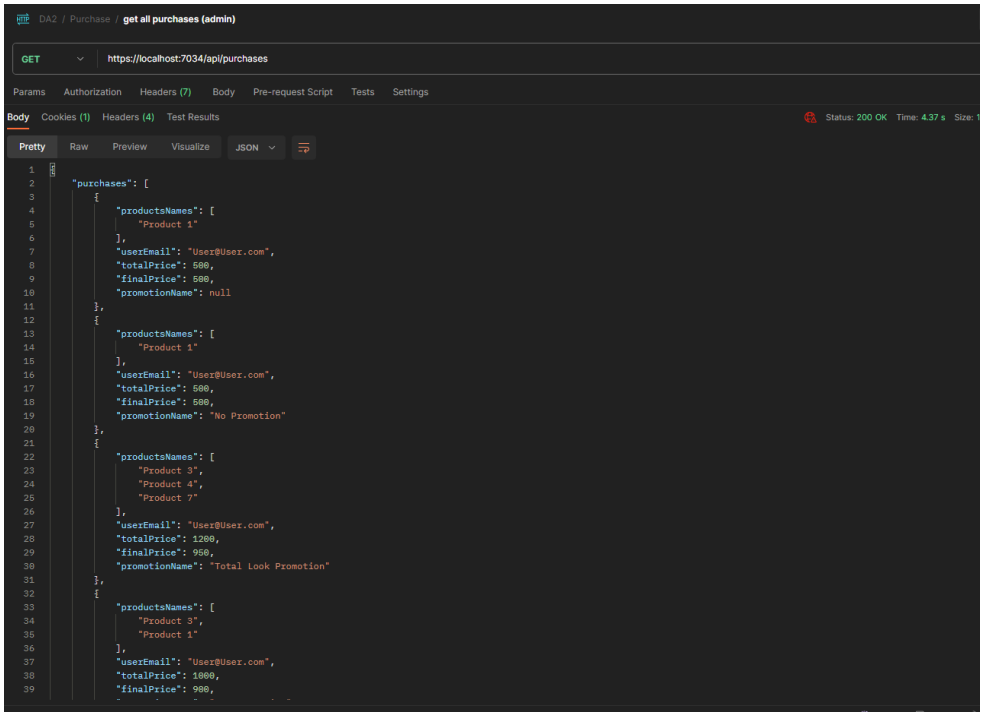
Delete user:

DA2 / User / delete user (admin) • From Invalid User
Save
DELETE
https://localhost:7034/api/users/665d8e77-9d40-4f00-af3e-08dbc52c5616ot
Send
Params Authorization Headers (7) Body Pre-request Script Tests Settings
Cookies
Query Params
Key Value Description Bulk Edit
Body Cookies (1) Headers (4) Test Results
Status: 400 Bad Request Time: 14 ms Size: 441 B Save as Example
Pretty Raw Preview Visualize JSON
{
"type": "https://tools.ietf.org/html/rfc7231#section-6.5.1",
"title": "One or more validation errors occurred.",
"status": 400,
"traceId": "00-8531b467e8fe43739111823877564899-f76a144098658673-00",
"errors": {
"id": [
"The value '665d8e77-9d40-4f00-af3e-08dbc52c5616ot' is not valid."
]
}
}

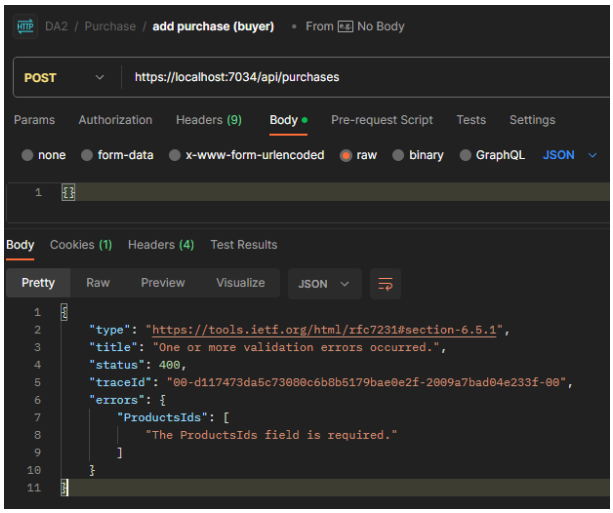

DA2 / User / delete user (admin) • From Successful Delete
DELETE
https://localhost:7034/api/users/85225DED-1510-44DC-C1EA-08DBC5CFB1A6
Params Authorization Headers (7) Body Pre-request Script Tests Settings
Query Params
Key Value Description
Body Cookies (1) Headers (4) Test Results
Status: 200 OK Time: 20 ms Size: 234 B
Pretty Raw Preview Visualize JSON
{
"message": "User deleted",
"email": "Email@Email.com45388",
"address": "Address",
"role": 1
}

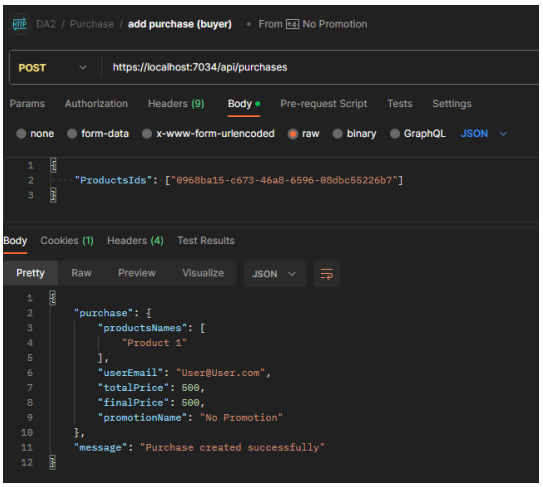
Purchase History:

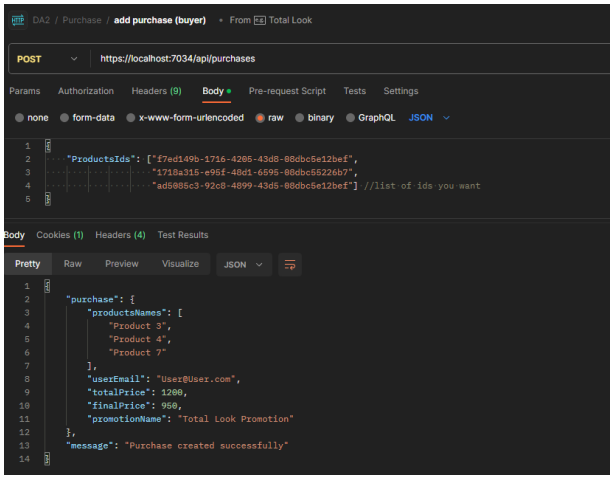


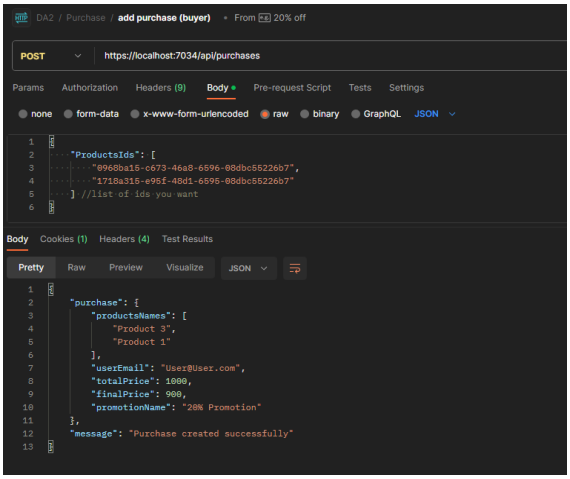


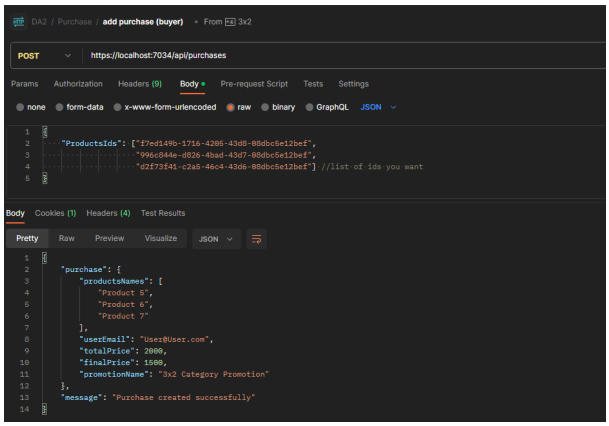
Promotions:

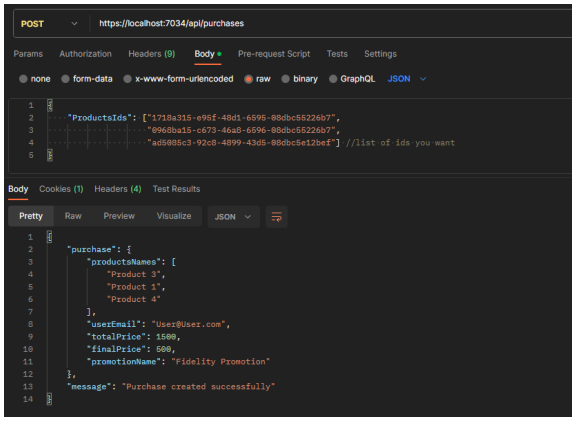

DA2 / Purchase / add purchase (buyer) • From No Body
POST
https://localhost:7034/api/purchases
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Body Cookies (1) Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"type": "https://tools.ietf.org/html/rfc7231#section-6.5.1",
"title": "One or more validation errors occurred.",
"status": 400,
"traceId": "00-d117473da5c73080c6b8b5179bae0e2f-2009a7bad04e233f-00",
"errors": {
"ProductsIds": [
"The ProductsIds field is required."
]
}
}


DA2 / Purchase / add purchase (buyer) • From No Promotion
POST
https://localhost:7034/api/purchases
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"ProductsIds": ["0968ba15-c673-46a8-6596-08dbc55226b7"]
}
Body Cookies (1) Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"purchase": {
"productsNames": [
"Product 1"
],
"userEmail": "User@User.com",
"totalPrice": 500,
"finalPrice": 500,
"promotionName": "No Promotion"
},
"message": "Purchase created successfully"
}


DA2 / Purchase / add purchase (buyer) • From Total Look
POST
https://localhost:7034/api/purchases
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"ProductsIds": ["f7ed149b-1716-4205-43d8-08dbc5e12bef",
"1718a315-e95f-48d1-6595-08dbc55226b7",
"ad5085c3-92c8-4899-43d5-08dbc5e12bef"] //list of ids you want
}
Body Cookies (1) Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"purchase": {
"productsNames": [
"Product 3",
"Product 4",
"Product 7"
],
"userEmail": "User@User.com",
"totalPrice": 1200,
"finalPrice": 950,
"promotionName": "Total Look Promotion"
},
"message": "Purchase created successfully"
}


DA2 / Purchase / add purchase (buyer) • From 20% off
POST
https://localhost:7034/api/purchases
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"ProductsIds": [
"0968ba15-c673-46a8-6596-08dbc55226b7",
"1718a315-e95f-48d1-6595-08dbc55226b7"
] //list of ids you want
}
Body Cookies (1) Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"purchase": {
"productsNames": [
"Product 3",
"Product 1"
],
"userEmail": "User@User.com",
"totalPrice": 1000,
"finalPrice": 900,
"promotionName": "20% Promotion"
},
"message": "Purchase created successfully"
}


DA2 / Purchase / add purchase (buyer) • From 3x2
POST
https://localhost:7034/api/purchases
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"ProductsIds": ["f7ed149b-1716-4205-43d8-08dbc5e12bef",
"996c844e-d826-4bad-43d7-08dbc5e12bef",
"d2f73f41-c2a5-46c4-43d6-08dbc5e12bef"] //list of ids you want
}
Body Cookies (1) Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"purchase": {
"productsNames": [
"Product 5",
"Product 6",
"Product 7"
],
"userEmail": "User@User.com",
"totalPrice": 2000,
"finalPrice": 1500,
"promotionName": "3x2 Category Promotion"
},
"message": "Purchase created successfully"
}


POST
https://localhost:7034/api/purchases
Params Authorization Headers (9) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
{
"ProductsIds": ["1718a315-e95f-48d1-6595-08dbc55226b7",
"0968ba15-c673-46a8-6596-08dbc55226b7",
"ad5085c3-92c8-4899-43d5-08dbc5e12bef"] //list of ids you want
}
Body Cookies (1) Headers (4) Test Results
Pretty Raw Preview Visualize JSON
{
"purchase": {
"productsNames": [
"Product 3",
"Product 1",
"Product 4"
],
"userEmail": "User@User.com",
"totalPrice": 1500,
"finalPrice": 500,
"promotionName": "Fidelity Promotion"
},
"message": "Purchase created successfully"
}