

```
MINGW64/c/Users/ivan-
{"mensaje":"Carrito eliminado"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X POST "$URL_PROD/compra_articulo" \ -H "Content-Type: application/
json" \ -d '{"id_articulo": "$ID_ARTICULO", "cantidad": 2, "id_usu
ario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje":"Compra registrada"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: appli
cation/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": 1,
"id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje":"Se modificó la cantidad del carrito"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: appli
cation/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1,
"id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje":"Se modificó la cantidad del carrito"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: appli
cation/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": 1,
"id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje":"Se modificó la cantidad del carrito"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ |
```

Figura 30. Error “No hay suficientes artículos en stock”

- Error “No hay más artículos en el carrito” (si cantidad ya es 0 y se intenta -1):

```
curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type:
application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1,
"id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
}'
```



```
MINGW64/c:/Users/ivan-
{"id_usuario": "$ID_USUARIO", "token": "$TOKEN"}
}
{"mensaje": "Se modificó la cantidad del carrito"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: appli
cation/json" \ -d '{ "id_articulo": "$ID_ARTICULO", "incremento": -1,
" id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Se modificó la cantidad del carrito"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: appli
cation/json" \ -d '{ "id_articulo": "$ID_ARTICULO", "incremento": -1,
" id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: appli
cation/json" \ -d '{ "id_articulo": "$ID_ARTICULO", "incremento": -1,
" id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan-@LAPTOP-X515JA MINGW64 ~
$ |
```

- **Figura 31. Error “No hay más artículos en el carrito”**

9.4.9 Función de apoyo: consulta_carrito

`curl -s "$URL_PROD/consulta_carrito?id_usuario=$ID_USUARIO&token=$TOKEN"`

- Evidencia:


```
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Se modificó la cantidad del carrito"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s "$URL_PROD/consulta_carrito?id_usuario=$ID_USUARIO&token=$TOKEN"
[]
ivan@LAPTOP-X515JA MINGW64 ~
$
```

Figura 32. Respuesta JSON del carrito con artículos y cantidades

9.4.10 Checkout opcional: finaliza_compra (“Comprar todo”)

```
curl -s -X POST "$URL_PROD/finaliza_compra?id_usuario=$ID_USUARIO&token=$TOKEN" \ -H "Content-Type: application/json" \ -d "{}"
```

- Evidencias:


```
MINGW64/c/Users/ivan-
{"id_usuario": "$ID_USUARIO", "token": "$TOKEN"}
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s "$URL_PROD/consulta_carrito?id_usuario=$ID_USUARIO&token=$TOKEN"
[]
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X POST "$URL_PROD/compra_articulo" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "cantidad": 2, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Compra registrada"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X POST "$URL_PROD/finaliza_compra?id_usuario=$ID_USUARIO&token=$TOKEN" \ -H "Content-Type: application/json" \ -d "{}"
{"id_orden": 2, "total": 91.00}
ivan@LAPTOP-X515JA MINGW64 ~
$
```

Figura 33. Respuesta JSON con id_orden y total de la compra

```
MINGW64/c/Users/ivan-
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X PUT "$URL_PROD/modifica_carrito_compra" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "incremento": -1, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Transaction has already been rolled back or is not pending"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s "$URL_PROD/consulta_carrito?id_usuario=$ID_USUARIO&token=$TOKEN"
[]
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X POST "$URL_PROD/compra_articulo" \ -H "Content-Type: application/json" \ -d '{"id_articulo": "$ID_ARTICULO", "cantidad": 2, "id_usuario": "$ID_USUARIO", "token": "$TOKEN"}'
{"mensaje": "Compra registrada"}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s -X POST "$URL_PROD/finaliza_compra?id_usuario=$ID_USUARIO&token=$TOKEN" \ -H "Content-Type: application/json" \ -d "{}"
{"id_orden": 2, "total": 91.00}
ivan@LAPTOP-X515JA MINGW64 ~
$ curl -s "$URL_PROD/consulta_carrito?id_usuario=$ID_USUARIO&token=$TOKEN"
[]
ivan@LAPTOP-X515JA MINGW64 ~
$
```

Figura 34. Carrito vacío después del checkout (consulta_carrito)

10 Conclusiones

Se realizó el desarrollo y despliegue de un prototipo serverless de comercio electrónico, integrando Azure Functions con MySQL PaaS y un front-end HTML/JavaScript servido desde Azure Files. Se accedió al conjunto de servicios en la región Canada, aplicando la nomenclatura obligatoria y garantizando consistencia en las pruebas locales y móviles. El uso de transacciones en operaciones críticas como la compra, eliminación y modificación del carrito demostró ser fundamental para la integridad de datos, mientras que la verificación de acceso por token aseguró el control de seguridad.

La utilización de inteligencia artificial de GitHub Copilot aceleró la creación de plantillas de código, la estandarización de respuestas y la organización del proyecto, permitiendo concentrar esfuerzos en las reglas de negocio y la correcta orquestación entre front-end y back-end. Se accedió a las evidencias solicitadas y se instalaron configuraciones clave (variables y montajes) que facilitaron el funcionamiento en producción.

Enlace del chat de la IA generativa

Enlace: <https://github.com/copilot/share/421c1186-0264-88a1-a002-2047003800c2>

11 Referencias (Formato IEEE)

- [1] R. T. Fielding, “Architectural Styles and the Design of Network-based Software Architectures,” Ph.D. dissertation, Univ. California, Irvine, 2000.
- [2] Oracle, “Java Platform Standard Edition 17 API Specification – Class SecureRandom,” 2025. [Online]. Available: <https://docs.oracle.com/>
- [3] ISO/IEC 9075-1:2016, “Information technology — Database languages — SQL — Part 1: Framework (SQL/Framework).”
- [4] Oracle, “InnoDB Storage Engine,” MySQL 8.0 Reference Manual, 2025. [Online]. Available: <https://dev.mysql.com/doc/>
- [5] Apache Software Foundation, “Apache Tomcat 10 Documentation,” 2025. [Online]. Available: <https://tomcat.apache.org/>
- [6] Eclipse Foundation, “Jakarta RESTful Web Services (JAX-RS) Specification,” 2025. [Online]. Available: <https://jakarta.ee/specifications/>
- [7] FasterXML, “Jackson Databind and Core Libraries,” 2025. [Online]. Available: <https://github.com/FasterXML/jackson>
- [8] GitHub, “GitHub Copilot: AI Pair Programmer,” 2025. [Online]. Available: <https://github.com/features/copilot>
- [9] NIST, “Secure Hash Standard (SHS),” FIPS PUB 180-4, Aug. 2015.
- [10] W3C, “File API,” W3C Recommendation, 2015. [Online]. Available: <https://www.w3.org/TR/FileAPI/>
- [11] W3C, “Web Cryptography API,” W3C Recommendation, 2017. [Online]. Available: <https://www.w3.org/TR/WebCryptoAPI/>
- [12] OWASP Foundation, “OWASP Top 10: Web Application Security Risks,” 2023. [Online]. Available: <https://owasp.org/>
- [13] D. Thomas et al., “Bcrypt Password Hashing,” OpenBSD Project Documentation, 2025. [Online]. Available: <https://man.openbsd.org/>
- [14] P. Wuille et al., “Argon2: The Memory-Hard Function for Password Hashing,” RFC Draft (argon2), 2025.
- [15] Google, “Fetch API Living Standard,” WHATWG, 2025. [Online]. Available: <https://fetch.spec.whatwg.org/>

- [16] Mozilla, "Content Security Policy (CSP) Guide," MDN Web Docs, 2025. [Online]. Available: <https://developer.mozilla.org/>
- [17] Git, "Distributed Version Control System," Git Project, 2025. [Online]. Available: <https://git-scm.com/>
- [18] Azure, "Azure Virtual Machines Documentation," Microsoft Learn, 2025. [Online]. Available: <https://learn.microsoft.com/azure/virtual-machines/>