**Retrospectiva del Proyecto**

**“CondorShop (E-commerce)”**

**Tabla de contenido**

**Contenido**

[Datos del documento 3](#_heading=h.30j0zll)

[Descripción del proyecto. 4](#_heading=h.1fob9te)

[Descripción de las retrospectivas de los Sprint. 4](#_heading=h.3znysh7)

[Puntos de mejoras. 5](#_heading=)

[Lecciones aprendidas. 5](#_heading=h.tyjcwt)

# Datos del documento

**Histórico de Revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Fecha | Descripción/cambio | autor |
| 01.00 | 29/08/2025 |  | Juan Rodríguez |
| 01.01 | 26/09/2025 |  | Juan Rodríguez |
| 01.02 | 10/10/2025 |  | Juan Rodríguez |
|  |  |  |  |

**Información del Proyecto**

|  |  |
| --- | --- |
| Organización | Duoc UC. Escuela de Informática y Telecomunicaciones |
| Sección | 004V |
| Proyecto (Nombre) | “CondorShop (E-commerce)” |
| Fecha de Inicio | 11/08/2025 |
| Fecha de Término | 10/10/2025 |
| Patrocinador principal | DuocUC |
| Docente | Marcos Valenzuela |

**Integrantes**

| Rut | Nombre | Correo |
| --- | --- | --- |
| 21.343.914-6 | Benjamin Huenupe | Be.huenupe@duocuc.cl |
| 19.880.451-7 | Juan Rodríguez | Juap.rodriguez@duocuc.cl |

# Descripción del proyecto.

|  |
| --- |
| El proyecto CondorShop consiste en un sitio web de comercio electrónico con frontend estático (HTML5, Tailwind CSS y JavaScript) y una API REST en Node.js/Express conectada a una base de datos MySQL. El alcance del MVP incluye: catálogo por categorías/marcas, búsqueda y filtros; ficha de producto (imágenes, descripción, precio, stock); registro/login y perfil; carrito persistente (invitado/usuario); checkout con dirección (región/comuna), envío, IVA 19% y aceptación de términos; integración Webpay (sandbox) con manejo de estados (aprobado, rechazado, abortado, timeout); órdenes (creación, detalle, estados, correo de confirmación) y un admin mínimo para productos/órdenes. Opcional/post-MVP: cupones, promociones y reseñas moderadas. |

# Descripción de las retrospectivas de los Sprint.

|  |
| --- |
| **Sprint 1:**   * **Principales Problemas Detectados:** Cierre de criterios de aceptación medibles; datos semilla para catálogo/PDP. * **Solución Adoptada:** Criterios Gherkin y DoD; carga de datos; checklist de rendimiento por vista.   **Sprint 2:**   * **Principales Problemas Detectados:** Integración Webpay sandbox (callbacks, abort/timeout) y consistencia de IVA/términos. * **Solución Adoptada:** Manejo de estados; pruebas negativas (rechazo/abort/timeout); logging de callbacks; resumen validado.   **Sprint 3:**   * **Principales Problemas Detectados:** Exactitud de inventario y tiempos en panel admin; confirmación de órdenes y notificaciones. * **Solución Adoptada:** Bitácora de stock y conciliación; optimización de vistas; correo de confirmación y exportación de pedidos.   **Sprint 4:**   * **Principales Problemas Detectados:** Moderación de reseñas y control de promociones/cupones (si se implementan). * **Solución Adoptada:** Política de moderación y roles; seguimiento de campañas; métricas de uso de cupones.   **Sprint 5:**   * **Principales Problemas detectados:** flakes en pruebas de regresión, latencias puntuales en checkout bajo carga ligera, alertas con umbrales por ajustar, checklist legal/privacidad fine-tuning. * **Solución adoptada:** endurecer smoke suite (+retries/fixtures), cache estáticos/CDN, ajustar umbrales de alertas, revisión legal final de textos de privacidad/cookies. |

# Puntos de mejoras.

|  |
| --- |
| * **Rendimiento/UX:** mantener checklist de rendimiento (TTI, assets) en frontend HTML/Tailwind/JS. * **Integraciones:** pruebas exhaustivas de Webpay (incluye casos negativos: rechazo/abort/timeout). * **Seguridad y datos:** reforzar control de acceso y consistencia de stock en MySQL. * **Monitoreo (post-MVP):** instrumentar métricas (web-vitals/eventos de negocio) y alertas de pago/envío. |

# Lecciones aprendidas.

|  |
| --- |
| * **Comunicación efectiva:** criterios y DoD definidos antes de iniciar el desarrollo del sprint. * **Flexibilidad enfocada en valor:** priorizar slices verticales (Catálogo→PDP→Carrito→Checkout). * **Pruebas rigurosas:** incluir pruebas negativas y monitoreo desde el día 1 en integraciones de pago. * **Documentación operativa:** guías de Admin y conciliación de inventario para reducir incidentes. |