**Guía3. Informe final Proyecto APT**

**Asignatura Capstone**

|  |
| --- |
| **1. Informe final Proyecto APT** |
| El objetivo de este informe es que describas los aspectos más relevantes de tu Proyecto APT. Es importante que fundamentes las decisiones que tuviste que tomar a lo largo del proceso.  A continuación, encontrarás distintos campos que deberás completar con la información solicitada, los que dan cuenta del resumen de tu proyecto APT y sus principales resultados. |

|  |  |
| --- | --- |
| Nombre del proyecto | SuperLocaliza — Localización de productos en tienda y gestión administrativa (catálogo, promos, reportes). |
| Área (s) de desempeño(s) | Análisis y diseño de sistemas; Desarrollo de software; Bases de datos; Arquitectura y despliegue; Gestión de proyectos; Aseguramiento de calidad. |
| Competencias | Levantamiento y priorización de requisitos y casos de uso; modelado UML (clases, actividades) y **ER**; arquitectura 4+1 y vista física; diseño de procesos TO-BE con **RACI** y control de cambios; decisiones tecnológicas (Next.js, Django/DRF, PostgreSQL); documentación técnica y métricas de calidad (rendimiento, seguridad, disponibilidad). |

|  |  |
| --- | --- |
| **Contenidos del informe final** | |
| 1. Relevancia del proyecto APT | El proyecto **SuperLocaliza** aborda la dificultad de **encontrar rápidamente productos** en tiendas físicas y la necesidad de **gestión centralizada** del catálogo (promociones, reportes). Es relevante para el **retail** porque reduce tiempos de atención, mejora la **experiencia del cliente** y optimiza la operación en sala. Se sitúa en el contexto local (supermercados/comercio minorista) y afecta a **usuarios finales** y **personal de tienda/administración**. El aporte de valor es una solución con **trazabilidad y métricas de calidad**, lista para escalar. |
| 2. Objetivos | * **General:** Diseñar y documentar la arquitectura y procesos de SuperLocaliza para **localización de productos** y **gestión administrativa**. * **Específicos:** (i) modelar dominio (**UML/ER**), (ii) definir procesos clave (Registro, Login, Búsqueda/Ubicación, Detalle, Promos, Reportes), (iii) establecer **despliegue** y **vista física**, (iv) fijar métricas de **calidad** (rendimiento, seguridad, disponibilidad). |
| 3. Metodología | **Ágil por iteraciones**: análisis → diseño (UML/ER) → validación (escenarios de calidad) → despliegue lógico/físico → documentación final. Pertinencia: permite priorizar flujos de mayor valor, controlar cambios y entregar evidencias incrementales. Herramientas: draw.io, suite ofimática, control de versiones. |
| 4. Desarrollo | **Etapas realizadas:** modelado del dominio (UML/ER), definición de procesos clave (Registro/Login, Búsqueda/Ubicación, Detalle, Promos, Reportes), diagramas de actividad (Usuario/Admin), decisiones tecnológicas (Next.js + Django/DRF + PostgreSQL), despliegue (Auth/Overview/Detallado) y vista física, documentación (DAS y TO-BE con RACI y control de cambios). **Facilitadores:** diccionario de datos y mockups claros, uso de draw.io y control de versiones, trabajo por iteraciones con revisiones cortas. **Dificultades:** cambios de alcance a mitad de iteración, incongruencias iniciales entre diccionario y ER, falta de datos reales para validar búsqueda/tiempos, retroalimentación tardía. **Ajustes realizados:** “congelamiento de alcance” por sprint; fuente única de verdad del diccionario conciliada con el ER; creación de datos sintéticos y scripts de carga; división del ER en Operacional/Soporte; simplificación de flujos y normalización de estilos/pies de figura. |
| 5. Evidencias | * **DAS** completo (vistas 4+1). * **Proceso TO-BE** con **RACI** y **política de control de cambios**. * **Diagramas** exportados (casos de uso, actividades, clases, **ER**, despliegue y **vista física**). * **Priorización y calidad:** tabla de casos de uso relevantes y **6 escenarios de calidad** con métricas (p95 de tiempos, 99,5% de disponibilidad, RTO/RPO, seguridad JWT/TLS). |
| 6. Intereses y proyecciones profesionales | El proyecto fortaleció mis intereses en **arquitectura de software**, **modelado de datos** y **diseño de procesos**. Quiero seguir profundizando en **backend/DB con Python**, **observabilidad** (trazabilidad y dashboards) y **automatización de despliegues**. **Proyección laboral:** roles de **Analista/Arquitecto junior** o **Desarrollador backend/DB**, con foco en APIs, datos y calidad; a mediano plazo, especialización en **arquitectura y plataformas**. |