Proyecto Data Alive: Registro de Riesgos

Fecha: 25 de agosto de 2025
Autor: Emmanuel Eduardo Pérez Cabrera, Líder del Proyecto
Versión: 1.2

1. Introducción

Este documento identifica, analiza y establece planes de mitigación para los riesgos potenciales que podrían afectar el cronograma, el alcance o el éxito del **Proyecto Data Alive**. El propósito de este registro es gestionar proactivamente estos riesgos para minimizar su impacto y asegurar el cumplimiento de los objetivos del proyecto. Este es un documento vivo y será actualizado a medida que el proyecto evolucione.

2. Matriz de Riesgos y Mitigaciones

ID	Categoría	Riesgo	Plan de Mitigación y Respuesta
R-001	**Técnico**	**Complejidad de la Integración Multinube:** La interconexión de múltiples tecnologías de vanguardia (AWS, Azure, Databricks, Snowflake, Kafka, RAG) crea puntos de falla frágiles en las "costuras" entre servicios, donde un cambio en una plataforma puede impactar a otra de forma inesperada.	**Estrategia de "Rebanadas Verticales":** En lugar de construir componentes de forma aislada, el enfoque será construir y validar pipelines funcionales de extremo a extremo a pequeña escala (ej. Ingesta → Procesamiento → Consumo) en una sola nube primero. Esto validará la integración de forma temprana y reducirá la complejidad inicial.
R-002	**Técnico**	**Calidad y Disponibilidad de Datos de Origen:**	**Adelantar la Fase 10 (Calidad de Datos):**

		Los datos de fuentes externas (ej. portal de la CNH) pueden presentar inconsistencias, errores de formato o interrupciones no previstas, lo que podría corromper los datasets o detener los pipelines de procesamiento.	Implementar un marco de validación de datos (ej. Great Expectations o scripts de PySpark) en las etapas tempranas del pipeline. Se establecerán reglas de calidad para asegurar que solo los datos que cumplen con un estándar mínimo pasen de la zona raw a la bronze.
R-003	**Gestión y Ejecución**	**Agotamiento y Pérdida de Foco (Burnout):** Al ser un proyecto personal de gran ambición (21 fases), existe un alto riesgo de que la magnitud del trabajo genere una pérdida de motivación o agotamiento en el líder del proyecto, afectando la consistencia y el avance.	**Gestión Ágil con Sprints Cortos:** Utilizar el tablero de Trello para definir sprints semanales con objetivos claros y alcanzables. Priorizar la finalización de tareas para generar "victorias tempranas" visibles que mantengan el impulso y la motivación. La consistencia es más importante que la velocidad.
R-004	**Gestión y Ejecución**	**Desviación del Alcance (Scope Creep):** La tentación de	**Uso Estricto del Backlog de Trello:** El backlog de 21 fases actúa como

		incorporar nuevas tecnologías o funcionalidades no contempladas en el plan maestro antes de que el núcleo del proyecto esté consolidado, arriesgando el cronograma y los objetivos principales.	el guardián del alcance. Cualquier nueva idea o requerimiento se registrará como una nueva tarjeta al final del backlog y no será priorizada hasta que el alcance actual se haya completado y validado.
R-005	**Estratégico**	**Complejidad de la Capa RAG:** La implementación del sistema de Generación Aumentada por Recuperación (Fase 18) puede convertirse en un "pozo sin fondo", consumiendo un tiempo desproporcionado en el ajuste fino de modelos y prompts sin entregar valor de negocio tangible de forma temprana.	**Definir un Producto Mínimo Viable (MVP) para el RAG:** El objetivo inicial del RAG se acotará a un caso de uso específico y medible (ej: *explicar la variación de producción de un día específico usando un reporte como única fuente*). Se expandirán las capacidades del RAG de forma iterativa solo después de haber resuelto exitosamente el MVP.
R-006	**Técnico / Datos**	**Dependencia de Datos de Streaming Inexistentes:** La funcionalidad de ingesta en tiempo	**Desarrollo de un Generador de Datos Sintéticos:** Crear un script (ej. en Python) que simule los datos de

		real con Kafka (Fase 7) depende de flujos de datos de producción diaria que no están disponibles actualmente, impidiendo el desarrollo y la validación de los pipelines de streaming.	producción (ej. ID de pozo, volumen, presión, temperatura) y los emita a un tópico de Kafka. Esto permitirá construir, probar y validar todo el pipeline de streaming de extremo a extremo con datos realistas antes de la integración con la fuente real.
R-007	**Gestión / Financiero**	**Gasto no controlado en la Nube:** El costo de los servicios en la nube (AWS, Azure, Snowflake) podría exceder el presupuesto mensual asignado (aprox. $5,000 MXN), debido a recursos de cómputo dejados activos innecesariamente o a costos inesperados (ej. transferencia de datos).	**Implementación Proactiva de la Fase 4 (Control de Costos):** 1) Configurar **AWS Budgets y Azure Cost Management** con alertas al 50%, 75% y 90% del presupuesto. 2) Aplicar **etiquetado de costos** riguroso a todos los recursos. 3) Establecer una **rutina de apagado** para recursos de cómputo no esenciales en DEV/QA. 4) Priorizar servicios **serverless o con auto-pausa**.