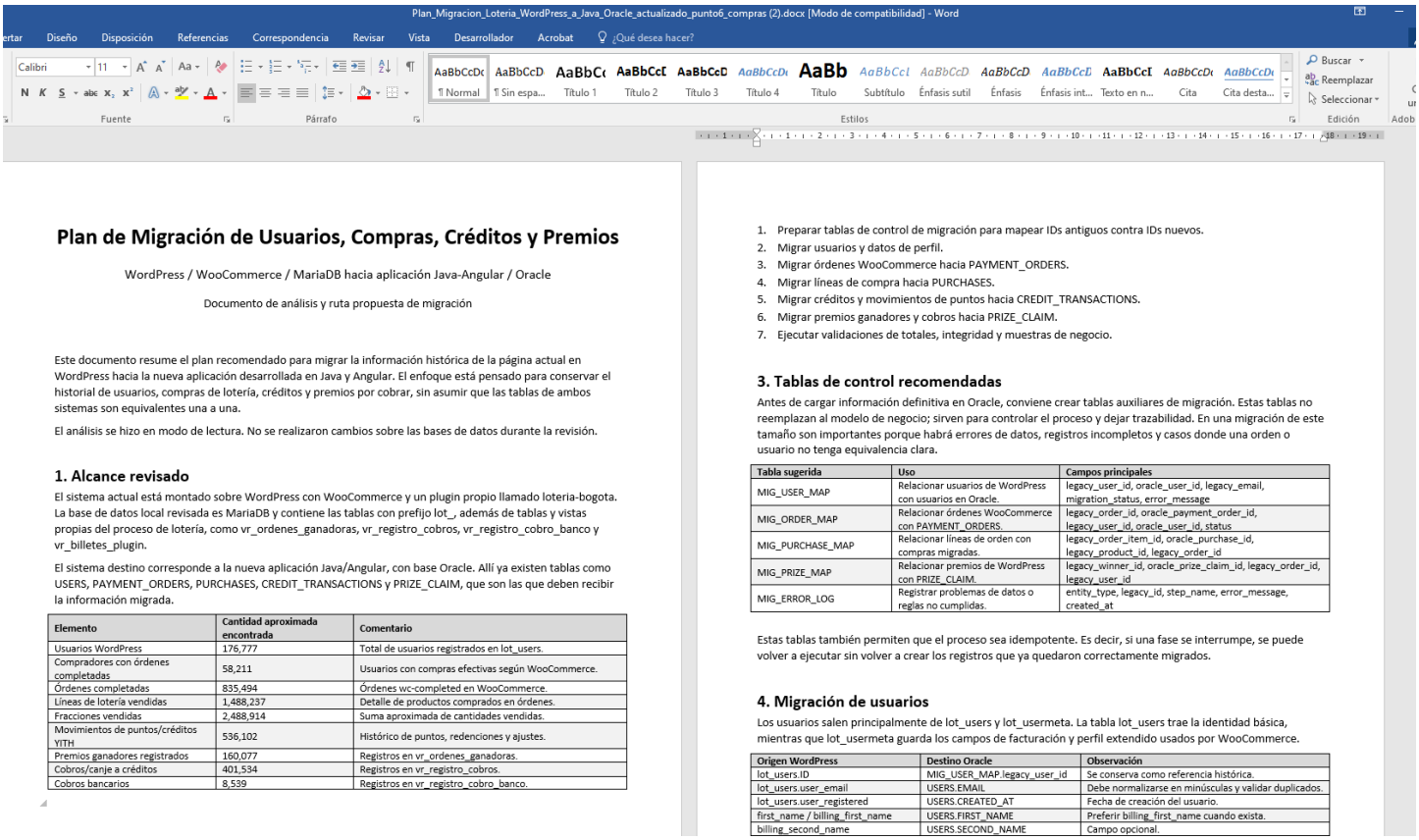
**Obligación 5. El contratista debe desarrollar las actividades de análisis, diseño, construcción, de los requerimientos en acompañamiento a los usuarios de la entidad asignados para realizar las pruebas funcionales, parametrizar y poner en funcionamiento la solución**

Se ejecutan 2 pruebas de migración con la entrega de dos dumps de pagina web. Se ejecutan plan de migración perfeccionándolo con las pruebas realizadas para entregar al final al cliente la misma información que podría encontrar de datos de perfil, compras, créditos y premios.


Plan_Migracion_Loteria_WordPress_a_Java_Oracle_actualizado_punto6_compras (2).docx [Modo de compatibilidad] - Word

## Plan de Migración de Usuarios, Compras, Créditos y Premios

WordPress / WooCommerce / MariaDB hacia aplicación Java-Angular / Oracle

Documento de análisis y ruta propuesta de migración

Este documento resume el plan recomendado para migrar la información histórica de la página actual en WordPress hacia la nueva aplicación desarrollada en Java y Angular. El enfoque está pensado para conservar el historial de usuarios, compras de lotería, créditos y premios por cobrar, sin asumir que las tablas de ambos sistemas son equivalentes una a una.

El análisis se hizo en modo de lectura. No se realizaron cambios sobre las bases de datos durante la revisión.

### 1. Alcance revisado

El sistema actual está montado sobre WordPress con WooCommerce y un plugin propio llamado loteria-bogota. La base de datos local revisada es MariaDB y contiene las tablas con prefijo lot_, además de tablas y vistas propias del proceso de lotería, como vr_ordenes_ganadoras, vr_registro_cobros, vr_registro_cobro_banco y vr_billetes_plugin.

El sistema destino corresponde a la nueva aplicación Java/Angular, con base Oracle. Allí ya existen tablas como USERS, PAYMENT_ORDERS, PURCHASES, CREDIT_TRANSACTIONS y PRIZE_CLAIM, que son las que deben recibir la información migrada.

| Elemento | Cantidad aproximada encontrada | Comentario |
|---|---|---|
| Usuarios WordPress | 176,777 | Total de usuarios registrados en lot_users. |
| Compradores con órdenes completadas | 58,211 | Usuarios con compras efectivas según WooCommerce. |
| Órdenes completadas | 835,494 | Órdenes wc-completed en WooCommerce. |
| Líneas de lotería vendidas | 1,488,237 | Detalle de productos comprados en órdenes. |
| Fracciones vendidas | 2,488,914 | Suma aproximada de cantidades vendidas. |
| Movimientos de puntos/créditos YITH | 536,102 | Histórico de puntos, redenciones y ajustes. |
| Premios ganadores registrados | 160,077 | Registros en vr_ordenes_ganadoras. |
| Cobros/canje a créditos | 401,534 | Registros en vr_registro_cobros. |
| Cobros bancarios | 8,539 | Registros en vr_registro_cobro_banco. |

1. Preparar tablas de control de migración para mapear IDs antiguos contra IDs nuevos.
2. Migrar usuarios y datos de perfil.
3. Migrar órdenes WooCommerce hacia PAYMENT_ORDERS.
4. Migrar líneas de compra hacia PURCHASES.
5. Migrar créditos y movimientos de puntos hacia CREDIT_TRANSACTIONS.
6. Migrar premios ganadores y cobros hacia PRIZE_CLAIM.
7. Ejecutar validaciones de totales, integridad y muestras de negocio.

### 3. Tablas de control recomendadas

Antes de cargar información definitiva en Oracle, conviene crear tablas auxiliares de migración. Estas tablas no reemplazan al modelo de negocio; sirven para controlar el proceso y dejar trazabilidad. En una migración de este tamaño son importantes porque habrá errores de datos, registros incompletos y casos donde una orden o usuario no tenga equivalencia clara.

| Tabla sugerida | Uso | Campos principales |
|---|---|---|
| MIG_USER_MAP | Relacionar usuarios de WordPress con usuarios en Oracle. | legacy_user_id, oracle_user_id, legacy_email, migration_status, error_message |
| MIG_ORDER_MAP | Relacionar órdenes WooCommerce con PAYMENT_ORDERS. | legacy_order_id, oracle_payment_order_id, legacy_user_id, oracle_user_id, status |
| MIG_PURCHASE_MAP | Relacionar líneas de orden con compras migradas. | legacy_order_item_id, oracle_purchase_id, legacy_product_id, legacy_order_id |
| MIG_PRIZE_MAP | Relacionar premios de WordPress con PRIZE_CLAIM. | legacy_winner_id, oracle_prize_claim_id, legacy_order_id, legacy_user_id |
| MIG_ERROR_LOG | Registrar problemas de datos o reglas no cumplidas. | entity_type, legacy_id, step_name, error_message, created_at |

Estas tablas también permiten que el proceso sea idempotente. Es decir, si una fase se interrumpe, se puede volver a ejecutar sin volver a crear los registros que ya quedaron correctamente migrados.

### 4. Migración de usuarios

Los usuarios salen principalmente de lot_users y lot_usermeta. La tabla lot_users trae la identidad básica, mientras que lot_usermeta guarda los campos de facturación y perfil extendido usados por WooCommerce.

| Origen WordPress | Destino Oracle | Observación |
|---|---|---|
| lot_users.ID | MIG_USER_MAP.legacy_user_id | Se conserva como referencia histórica. |
| lot_users.user_email | USERS.EMAIL | Debe normalizarse en minúsculas y validar duplicados. |
| lot_users.user_registered | USERS.CREATED_AT | Fecha de creación del usuario. |
| first_name / billing_first_name | USERS.FIRST_NAME | Preferir billing_first_name cuando exista. |
| billing_second_name | USERS.SECOND_NAME | Campo opcional. |

