**CREA Y PRESENTA PROYECTOS DE BD**

**Nombre y Apellidos de los integrantes del grupo:**

**Estudiante1(E1):** Jesus Sebastian Silva Mendoza

**Estudiante2(E2):** Juan Jair Z’carlos Salvador Chumbes

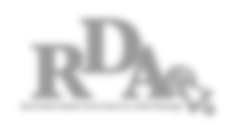
**Estudiante3(E3):** Sebastian Demian Quichca Soto

## Crear proyectos de base de datos

* + Ciclo de vida de desarrollo de sistemas

Se refiere a la metodología o enfoque que se utiliza para planificar, diseñar, desarrollar, implementar y mantener sistemas de software utilizando las tecnologías y herramientas proporcionadas por Oracle Corporation. Oracle ofrece una variedad de productos y soluciones para el desarrollo de aplicaciones empresariales, bases de datos y sistemas, y su ciclo de vida de desarrollo de sistemas se adapta a estas tecnologías.

* + 1. Requisitos: En esta fase, se recopilan y documentan los requisitos del sistema. Esto implica la comprensión de las necesidades del negocio, la identificación de los objetivos y la definición de los requisitos funcionales y no funcionales.
    2. Diseño: En esta etapa, se crea un diseño detallado del sistema. Esto puede incluir la arquitectura de la base de datos, el diseño de la interfaz de usuario, la definición de flujos de trabajo y la planificación de la seguridad.
    3. Desarrollo: Durante esta fase, se desarrollan las aplicaciones y componentes del sistema utilizando las herramientas y tecnologías de Oracle, como Oracle Database, Oracle Application Express (APEX), Java, PL/SQL, entre otros. Los desarrolladores escriben código, realizan pruebas unitarias y aseguran que el sistema cumpla con los requisitos.
    4. Pruebas: En esta etapa, se realizan pruebas exhaustivas para garantizar que el sistema funcione correctamente y cumpla con los requisitos. Esto puede incluir pruebas de unidad, pruebas de integración, pruebas de rendimiento y pruebas de aceptación del usuario.
    5. Implementación: Una vez que el sistema ha pasado las pruebas satisfactoriamente, se implementa en el entorno de producción. Esto puede incluir la migración de datos, la configuración de servidores y la capacitación de usuarios.
    6. Mantenimiento y Soporte\*\*: Después de la implementación, se proporciona soporte continuo y se realizan actualizaciones y mejoras según sea necesario. Esto asegura que el sistema siga funcionando de manera eficiente y se adapte a las cambiantes necesidades del negocio.



Oracle ofrece herramientas y productos específicos para cada una de estas fases, como Oracle Developer Suite, Oracle APEX, Oracle Database, Oracle Fusion Middleware, entre otros. Además, Oracle también promueve enfoques ágiles y DevOps para el desarrollo de sistemas, lo que implica ciclos de desarrollo más cortos y una colaboración más estrecha entre los equipos de desarrollo y operaciones.

Es importante destacar que el ciclo de vida de desarrollo de sistemas en Oracle puede variar según el proyecto y las necesidades específicas de la organización, pero estas son las fases generales que se siguen en la mayoría de los proyectos de desarrollo de software.

* + Descripción general del proyecto y primeros pasos

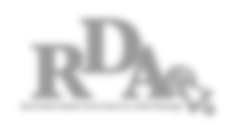
La "Descripción general del proyecto y primeros pasos" en Oracle se refiere al proceso inicial de planificación y establecimiento de los fundamentos para un proyecto que involucra tecnologías y soluciones de Oracle. Antes de comenzar a desarrollar o implementar una solución específica, es esencial comprender claramente el alcance, los objetivos, los recursos necesarios y otros aspectos clave del proyecto. A continuación, te proporciono una descripción general de los pasos comunes que se siguen en esta etapa:

* + 1. Definición de Objetivos y Alcance del Proyecto:
       - Identificar y documentar los objetivos del proyecto: ¿Qué se espera lograr con la implementación de la solución Oracle?
       - Determinar el alcance del proyecto: ¿Qué funcionalidades y componentes estarán incluidos en la solución? ¿Cuál será su extensión y límites?
    2. Identificación de Stakeholders:
       - Identificar todas las partes interesadas (stakeholders) en el proyecto, tanto internas como externas.
       - Establecer comunicación y colaboración efectiva con los stakeholders para asegurar una comprensión común de los objetivos y expectativas.

Logotipo

El contenido generado por IA puede ser incorrecto.17

* + 1. Formación de un Equipo de Proyecto:

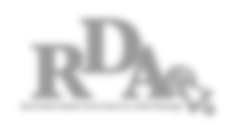


* + - * Reclutar y asignar a los miembros del equipo de proyecto, incluyendo roles como el líder del proyecto, desarrolladores, administradores de bases de datos, analistas de negocio y otros expertos necesarios.
    1. Planificación del Proyecto:
       - Crear un plan de proyecto que incluya un cronograma detallado, asignación de recursos, estimación de costos y riesgos potenciales.
       - Definir las fases del proyecto y los entregables esperados en cada etapa.
    2. Evaluación de Tecnologías Oracle:
       - Determinar qué tecnologías y productos de Oracle son adecuados para cumplir con los objetivos del proyecto.
       - Evaluar la necesidad de adquirir licencias de software y hardware de Oracle.
    3. Análisis de Requisitos Iniciales:
       - Iniciar el proceso de recopilación y análisis de requisitos para comprender las necesidades del negocio y del usuario final.
       - Documentar los requisitos iniciales que servirán como base para el diseño y desarrollo subsiguientes.
    4. Selección de Arquitectura y Diseño Inicial:
       - Definir una arquitectura inicial de alto nivel para la solución Oracle, considerando factores como la escalabilidad, la disponibilidad y el rendimiento.
       - Realizar un diseño preliminar de la solución.
    5. Establecimiento de Procedimientos y Políticas:
       - Definir los procedimientos y políticas que regirán el desarrollo, implementación y gestión de la solución Oracle, incluyendo prácticas de seguridad, copias de seguridad y políticas de acceso.
    6. Presupuesto y Financiamiento:
       - Establecer un presupuesto para el proyecto, incluyendo los costos de licencias, hardware, recursos humanos y otros gastos asociados.
       - Identificar fuentes de financiamiento y asegurar el respaldo financiero necesario.
    7. Logotipo

       El contenido generado por IA puede ser incorrecto.Aprobación del Plan:

18

* + - * Presentar el plan de proyecto, la descripción general y los presupuestos a los patrocinadores y partes interesadas clave para su aprobación.



* + 1. Inicio del Proyecto:
       - Una vez que se ha obtenido la aprobación, dar inicio al proyecto formalmente y comenzar con las fases de diseño y desarrollo de la solución Oracle.

Esta fase inicial del proyecto es fundamental para establecer una base sólida y garantizar que todos los participantes tengan una comprensión clara de lo que se espera lograr. Proporciona la dirección y estructura necesarias para el éxito continuo del proyecto basado en tecnologías de Oracle.

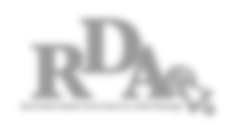
A continuación, brindaremos un ejemplo de un proyecto que implica la implementación de una base de datos Oracle en una empresa:

1. Proyecto: Implementación de una Base de Datos Oracle para la Gestión de Inventarios.
2. Objetivos:
   * Mejorar el seguimiento y control de los niveles de inventario de la empresa.
   * Optimizar la eficiencia en la gestión de existencias y reducir pérdidas debido a productos obsoletos o faltantes.
   * Facilitar la generación de informes y análisis de inventario en tiempo real.
3. Alcance:
   * La solución incluirá la creación de una base de datos Oracle que almacene información detallada sobre productos, proveedores, transacciones de inventario y ubicaciones de almacenamiento.
   * La implementación abarcará todos los departamentos de la empresa involucrados en la gestión de inventarios.
   * Se proporcionarán interfaces de usuario para la entrada y consulta de datos de inventario.
4. Stakeholders:
   * Director de Operaciones.
   * Gerente de Logística.
   * Gerente de TI.
   * Personal de almacén y logística.
   * Logotipo

     El contenido generado por IA puede ser incorrecto.Proveedores de hardware y software de Oracle.

19

1. Equipo de Proyecto:

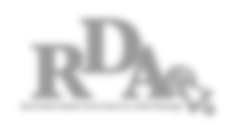


* + Líder del Proyecto: John Doe.
  + Desarrolladores de Bases de Datos: María Pérez y David Smith.
  + Analista de Negocios: Laura Johnson.
  + Administrador de Bases de Datos Oracle: Carlos González.

1. Planificación del Proyecto:
   * Duración estimada del proyecto: 6 meses.
   * Presupuesto estimado: $150,000.
   * Cronograma con fases de análisis, diseño, desarrollo, pruebas y despliegue.
   * Evaluación de riesgos, incluyendo posibles interrupciones operativas durante la implementación.
2. Tecnologías Oracle:
   * Oracle Database 19c.
   * Oracle SQL Developer para desarrollo y administración de la base de datos.
3. Análisis de Requisitos Iniciales:
   * Reuniones con departamentos de logística y almacén para comprender las necesidades específicas de seguimiento de inventario.
   * Documentación de los tipos de datos requeridos, flujos de trabajo actuales y futuros y requisitos de informes.
4. Selección de Arquitectura y Diseño Inicial:
   * Arquitectura de tres capas con una base de datos Oracle centralizada y aplicaciones de interfaz de usuario basadas en la web.
   * Diseño preliminar de la estructura de la base de datos, incluyendo tablas para productos, transacciones y usuarios.
5. Procedimientos y Políticas:
   * Políticas de seguridad para restringir el acceso a datos sensibles.
   * Procedimientos de copias de seguridad regulares para garantizar la integridad de los datos.
6. Presupuesto y Financiamiento:
   * Logotipo

     El contenido generado por IA puede ser incorrecto.El presupuesto de $150,000 cubrirá licencias de Oracle, hardware de servidor, costos de desarrollo y capacitación del personal.

20



1. Aprobación del Plan:
   * Presentación del plan aprobado ante el Comité Ejecutivo de la empresa.
2. Inicio del Proyecto:
   * Inicio de la fase de análisis y diseño de la base de datos Oracle.

Este es solo un ejemplo simplificado, pero ilustra cómo se pueden aplicar los pasos mencionados en la descripción general de un proyecto que involucra tecnologías Oracle. Cada proyecto puede ser único en cuanto a sus objetivos, alcance y requisitos, pero estos pasos generales ayudan a establecer una base sólida para su ejecución exitosa.

* Gestión de proyectos de presentación

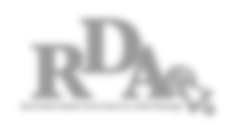
A continuación, proporcionaremos una descripción general de cómo las herramientas de Oracle, como Oracle Business Intelligence (OBI) o Oracle Analytics, pueden utilizarse en la gestión de proyectos de presentación:

* 1. Recopilación de Datos y Análisis: Utiliza Oracle Business Intelligence o Oracle Analytics para recopilar datos relevantes y realizar análisis que respalden la creación de la presentación. Estas herramientas pueden conectarse a diversas fuentes de datos para obtener información clave.
  2. Diseño de la Presentación: Utiliza herramientas de creación de informes y visualización de datos en Oracle para diseñar las diapositivas de la presentación. Puedes crear gráficos, tablas y visualizaciones interactivas para comunicar de manera efectiva la información.
  3. Colaboración y Revisión: Oracle proporciona capacidades de colaboración que permiten a los equipos trabajar juntos en la creación de presentaciones. Puedes compartir los informes y las visualizaciones con los miembros del equipo para obtener comentarios y revisiones.
  4. Seguimiento y Control: Utiliza las capacidades de seguimiento y control de proyectos en Oracle para mantener un registro de las versiones de la presentación, realizar un seguimiento de los cambios y asegurarte de que el proyecto de presentación avance de acuerdo con el cronograma.
  5. Logotipo

     El contenido generado por IA puede ser incorrecto.Distribución y Presentación: Una vez que la presentación esté lista, puedes utilizar las herramientas de Oracle para distribuir el contenido a los interesados y, si es necesario, presentar la información en reuniones o sesiones informativas.

21

Oracle ofrece una amplia gama de herramientas y soluciones que pueden ser útiles en la gestión de proyectos de presentación.



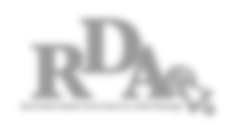
* Componentes de la presentación final

Se refieren a los elementos que se utilizan para crear y mostrar la interfaz de usuario de una aplicación en una base de datos Oracle. Estos componentes son esenciales para diseñar y construir aplicaciones basadas en bases de datos que permitan a los usuarios interactuar con los datos de manera efectiva.

* 1. Formularios: Los formularios son interfaces de usuario que permiten a los usuarios ingresar y modificar datos en una base de datos Oracle. Oracle Forms es una herramienta popular para desarrollar formularios que se ejecutan en aplicaciones cliente-servidor o web.
  2. Informes: Los informes son componentes que se utilizan para presentar datos en formato tabular o gráfico. Oracle Reports es una herramienta que permite diseñar y generar informes basados en datos de una base de datos Oracle.
  3. Páginas web: Oracle permite crear aplicaciones web utilizando tecnologías como Oracle Application Express (APEX) o Java Server Pages (JSP). Estas aplicaciones web pueden contener formularios, informes y otros elementos de interfaz de usuario.
  4. Componentes de interfaz de usuario en APEX: En Oracle APEX, los componentes de interfaz de usuario incluyen elementos como cuadros de texto, botones, listas desplegables y gráficos que se pueden agregar y personalizar en las páginas web de la aplicación.
  5. Gráficos: Oracle ofrece capacidades para generar gráficos interactivos basados en datos de la base de datos. Estos gráficos pueden ayudar a los usuarios a visualizar la información de manera más efectiva.
  6. Paneles de control: Los paneles de control (dashboards) son páginas o áreas de una aplicación que muestran una vista consolidada de datos importantes. Pueden incluir gráficos, tablas y otros elementos para proporcionar una visión general rápida de la información clave.
  7. Logotipo

     El contenido generado por IA puede ser incorrecto.Componentes personalizados: En algunas aplicaciones, es posible que necesites desarrollar componentes de presentación personalizados utilizando lenguajes de programación como Java o JavaScript para satisfacer requisitos específicos de la interfaz de usuario.

22



Estos componentes de presentación final son fundamentales para desarrollar aplicaciones que permitan a los usuarios interactuar con los datos almacenados en una base de datos Oracle de manera eficiente y efectiva.

## Presentación de proyectos de bases de datos

* + Creación de tablas para la presentación final

La creación de tablas en Oracle es un paso fundamental en el diseño de una base de datos relacional y es crucial para la presentación final de datos en aplicaciones. Las tablas son estructuras que almacenan datos de manera organizada en filas y columnas, y representan entidades o relaciones específicas en la base de datos.

* + 1. Definición de la estructura de la tabla:
       - Nombre de la tabla: Debes asignar un nombre descriptivo a la tabla.
       - Columnas: Define las columnas que compondrán la tabla y especifica sus nombres, tipos de datos y restricciones. Los tipos de datos pueden ser numéricos, de texto, de fecha, etc.
       - Restricciones: Puedes aplicar restricciones como claves primarias, claves foráneas, restricciones únicas o restricciones de verificación para garantizar la integridad de los datos.
    2. Clave primaria:
       - Define una o más columnas como clave primaria. Esto garantiza que cada fila en la tabla tenga un valor único en esa columna o conjunto de columnas. La clave primaria se utiliza para identificar de manera única cada registro.
    3. Clave foránea (Foreign Key):
       - Si tienes relaciones entre tablas, puedes definir claves foráneas en una tabla para establecer vínculos con registros en otra tabla. Esto permite mantener la integridad referencial de los datos.
    4. Índices:
       - Puedes crear índices en las columnas que se utilizan con frecuencia en consultas para mejorar el rendimiento de las búsquedas. Los índices aceleran la recuperación de datos.

Logotipo

El contenido generado por IA puede ser incorrecto.23

* + 1. Particionamiento:
       - En bases de datos grandes, puedes utilizar el particionamiento para dividir una tabla en particiones más pequeñas basadas en ciertos criterios (por ejemplo, por fecha o rango de valores). Esto puede mejorar la administración y el rendimiento de consultas.
    2. Almacenamiento y espacio en disco:
       - Define la cantidad de espacio de almacenamiento asignado a la tabla y la forma en que se gestionarán los datos en el disco, como tablespaces y esquemas de almacenamiento.
    3. Seguridad y permisos:
       - Establece permisos adecuados para la tabla para controlar quién puede ver, insertar, actualizar o eliminar datos en ella. Oracle permite administrar la seguridad a nivel de usuario y rol.
    4. Creación de la tabla:
       - Utiliza la sentencia SQL `CREATE TABLE` para crear la tabla en la base de datos. Asegúrate de que la sintaxis sea correcta y de que se apliquen las restricciones necesarias.

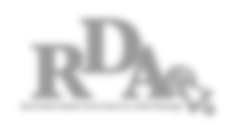
Ejemplo de creación de una tabla en Oracle:

Texto

El contenido generado por IA puede ser incorrecto.

**Figura N° 3** Creación de una tabla en Oracle

Una vez que la tabla se ha creado, puedes utilizarla para almacenar y presentar datos en aplicaciones Oracle mediante consultas SQL y herramientas de desarrollo como Oracle Forms, Oracle APEX o cualquier otra aplicación personalizada que interactúe con la base de datos.

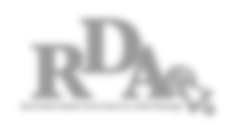


24

Logotipo

El contenido generado por IA puede ser incorrecto.

* + Preparación de documentación escrita



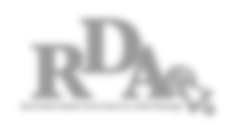
La preparación de documentación escrita en Oracle se refiere a la creación de documentación detallada y completa relacionada con una base de datos Oracle, sus objetos, procesos y procedimientos. Esta documentación es esencial para facilitar la comprensión, el mantenimiento y la administración eficiente de la base de datos.

* + 1. Propósito de la documentación:
       - La documentación escrita tiene varios propósitos clave en Oracle:
         * Facilitar la comprensión: Ayuda a los desarrolladores, administradores y usuarios a comprender la estructura de la base de datos y cómo interactuar con ella.
         * Apoyar el mantenimiento: Permite realizar cambios y mejoras de manera controlada y planificada.
         * Facilitar la solución de problemas: Facilita la identificación y corrección de problemas y errores.
         * Cumplimiento y auditoría: Puede ser requerida para cumplir con regulaciones y estándares de seguridad.
    2. Contenido de la documentación:
       - La documentación escrita en Oracle puede incluir los siguientes elementos:
         * Diagramas de diseño de la base de datos: Representación visual de la estructura de la base de datos, incluyendo tablas, relaciones y objetos.
         * Diccionario de datos: Lista detallada de todas las tablas, columnas, tipos de datos y restricciones en la base de datos.
         * Descripciones de objetos: Información sobre cada objeto de la base de datos, incluyendo su propósito, estructura y uso.
         * Procedimientos y scripts: Instrucciones detalladas sobre cómo realizar tareas específicas, como respaldos, restauraciones y mantenimiento.
         * Documentación de seguridad: Detalles sobre las políticas y medidas de seguridad implementadas en la base de datos.
         * Documentación de rendimiento: Información sobre el rendimiento de la base de datos y cómo optimizar consultas y procesos.
    3. Herramientas de documentación:
       - Logotipo

         El contenido generado por IA puede ser incorrecto.Oracle ofrece herramientas como Oracle SQL Developer, que permite generar automáticamente documentación de base de datos, incluyendo diagramas ER (Entity-Relationship) y descripciones de objetos.

25

* + - * También puedes utilizar herramientas de generación de documentación de terceros o crear documentación personalizada utilizando aplicaciones de procesamiento de texto.



* + 1. Mantenimiento y actualización:
       - La documentación debe mantenerse actualizada a medida que se realizan cambios en la base de datos. Esto asegura que la documentación sea precisa y útil en todo momento.
       - Es importante asignar responsabilidades para la actualización de la documentación y establecer un proceso de revisión periódica.
    2. Acceso y distribución:
       - La documentación debe estar disponible para todas las partes interesadas, incluyendo desarrolladores, administradores y usuarios. Puede distribuirse en formato impreso o electrónico, o estar disponible en una ubicación centralizada en línea.

La preparación de documentación escrita en Oracle es una práctica esencial para garantizar la integridad, la seguridad y la eficiencia de una base de datos. Proporciona una referencia valiosa para todos los involucrados en el desarrollo, mantenimiento y uso de la base de datos, lo que facilita la colaboración y la gestión de proyectos de base de datos exitosos.

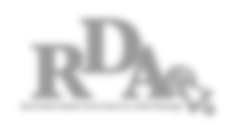
* + Preparación de materiales visuales

La preparación de materiales visuales en Oracle se refiere a la creación de representaciones visuales de datos, diagramas, gráficos y otros elementos que ayudan a comprender y presentar información relacionada con sistemas, bases de datos, arquitecturas y procesos que involucran tecnología Oracle. Estos materiales visuales son útiles para comunicar conceptos técnicos y facilitar la toma de decisiones.

* + 1. Diagramas de arquitectura:
       - Los diagramas de arquitectura representan la estructura y los componentes de una solución de Oracle, incluyendo servidores, bases de datos, aplicaciones y las relaciones entre ellos. Estos diagramas ayudan a los equipos técnicos y a los stakeholders a comprender la disposición de los recursos y cómo interactúan entre sí.
    2. Diagramas de flujo de procesos:
       - Logotipo

         El contenido generado por IA puede ser incorrecto.Los diagramas de flujo de procesos muestran las secuencias de actividades y tareas en un proceso de negocio o técnico que involucra tecnología Oracle.

26



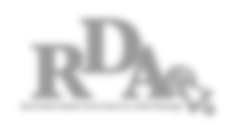
* + 1. Modelos entidad-relación (ER):
       - Los modelos ER representan la estructura de una base de datos Oracle, incluyendo tablas, columnas y las relaciones entre ellas. Estos modelos son esenciales para diseñar y comprender la arquitectura de la base de datos.
    2. Gráficos de rendimiento:
       - Los gráficos de rendimiento muestran métricas clave de rendimiento de una base de datos Oracle, como la utilización de CPU, el tiempo de respuesta de las consultas y el uso de recursos. Estos gráficos ayudan a identificar problemas de rendimiento y tomar decisiones informadas para su optimización.
    3. Gráficos de tendencias y análisis:
       - Los gráficos de tendencias muestran la evolución de datos a lo largo del tiempo, lo que es útil para identificar patrones y tendencias. Pueden utilizarse en informes y presentaciones para respaldar análisis y decisiones estratégicas.
    4. Tableros de control (dashboards):
       - Los tableros de control son interfaces visuales que muestran información clave de manera consolidada y en tiempo real. Oracle ofrece herramientas como Oracle Business Intelligence (OBIEE) y Oracle APEX para crear tableros de control personalizados.
    5. Visualización de datos:
       - La visualización de datos implica el uso de gráficos, mapas y otros elementos visuales para representar datos de manera efectiva. Puedes utilizar herramientas como Oracle Data Visualization para crear visualizaciones interactivas.
    6. Documentación técnica visual:
       - Los materiales visuales también pueden formar parte de la documentación técnica, incluyendo manuales, guías de usuario y documentación de diseño de sistemas Oracle.

La preparación de materiales visuales en Oracle es esencial para comunicar de manera efectiva información técnica y facilitar la toma de decisiones informadas. Estos materiales son útiles para equipos técnicos, ejecutivos y usuarios finales, y pueden contribuir significativamente al éxito de proyectos relacionados con tecnología Oracle.

Logotipo

El contenido generado por IA puede ser incorrecto.27

* + Presentaciones finales



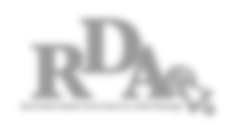
Se refieren a la etapa de un proyecto de tecnología que implica la comunicación de los resultados y hallazgos relacionados con un proyecto, aplicación o solución basada en la tecnología Oracle. Estas presentaciones son esenciales para compartir información clave con los stakeholders, equipos de proyecto y otros interesados, y pueden tomar varias formas, como informes, demostraciones, sesiones de capacitación o reuniones formales. A continuación, mostraremos información al respecto:

* + 1. Objetivos de las presentaciones finales:
       - Comunicación de resultados: Las presentaciones finales permiten mostrar los resultados obtenidos a lo largo del proyecto, como la funcionalidad de la aplicación, el rendimiento del sistema y los beneficios logrados.
       - Validación de requerimientos: Se utilizan para verificar que la solución implementada cumple con los requerimientos iniciales del proyecto.
       - Capacitación y transferencia de conocimientos: Pueden incluir sesiones de capacitación para usuarios finales y equipos de soporte técnico.
       - Aprobación y toma de decisiones: En algunas presentaciones, se busca la aprobación o la toma de decisiones críticas por parte de los stakeholders.
    2. Contenido de las presentaciones finales:
       - Descripción del proyecto: Incluye una visión general del proyecto, los objetivos, el alcance y el cronograma.
       - Demostración de la aplicación: Se muestra la funcionalidad de la aplicación en acción, lo que puede incluir casos de uso, flujos de trabajo y ejemplos prácticos.
       - Resultados y logros: Se presentan los resultados obtenidos, como mejoras en el rendimiento, eficiencia o ahorro de costos.
       - Lecciones aprendidas: Es importante compartir las lecciones aprendidas durante el proyecto, tanto las experiencias positivas como las áreas de mejora.
       - Plan de soporte y mantenimiento: Se discuten los planes futuros para el soporte técnico, mantenimiento y actualización de la solución Oracle.
    3. Formatos de presentaciones finales:
       - Informes escritos: Pueden incluir documentación detallada sobre el proyecto, sus resultados y recomendaciones.
       - Logotipo

         El contenido generado por IA puede ser incorrecto.Sesiones de demostración: Se pueden llevar a cabo demostraciones en vivo de la aplicación o sistemas Oracle.

28

* + - * Reuniones formales: Las reuniones cara a cara o en línea con stakeholders son comunes para presentaciones finales.



29

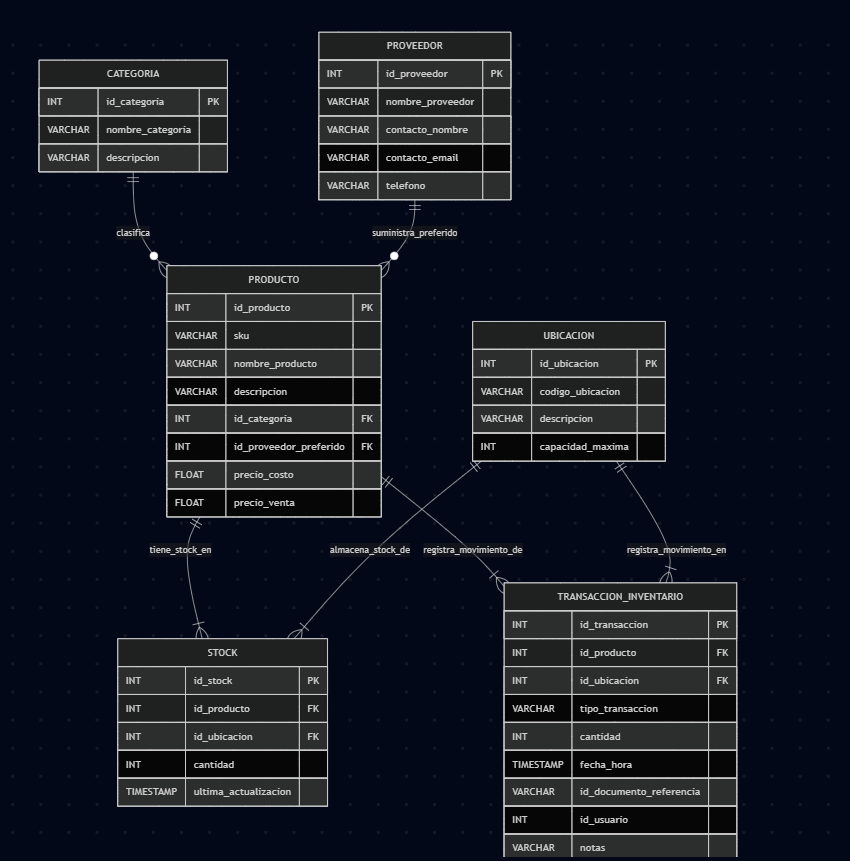
* + - * Materiales visuales: Se pueden utilizar diapositivas de presentación, gráficos y diagramas para ilustrar puntos clave.
    1. Audiencia y stakeholders:
       - Las presentaciones finales deben adaptarse a la audiencia específica. Esto puede incluir a ejecutivos, usuarios finales, equipos técnicos y otros stakeholders clave.
    2. Seguimiento y retroalimentación:
       - Después de la presentación final, es importante recopilar retroalimentación de los stakeholders y seguir el proceso de cierre del proyecto, que puede incluir la aceptación formal.

Las presentaciones finales en Oracle son un paso crítico en la finalización exitosa de un proyecto. Ayudan a asegurar que los resultados se comuniquen de manera efectiva, se validen con los interesados y se preparen para la siguiente fase, que puede incluir la implementación, la capacitación y el soporte continuo. La calidad de las presentaciones finales puede influir en la percepción del éxito del proyecto y en la satisfacción de los stakeholders.

**1. Introducción**

Este documento describe la estructura de la base de datos relacional InventarioDB. Su propósito es gestionar de manera eficiente el inventario de productos de una empresa, incluyendo información sobre productos, categorías, proveedores, ubicaciones de almacenamiento, niveles de stock y el historial de transacciones. El diseño aplica conceptos de **Mapeo de Relaciones** (1:N, M:N) y **Mapeo de Subtipos** (Single Table Inheritance) discutidos en la Tarea 4.

**2. Diagrama Entidad-Relación (ERD)**

**3. Descripción Detallada de las Tablas**

**3.1. Tabla Categoria**

* **Propósito:** Almacenar las categorías para clasificar los productos.
* **Columnas:**
  + id\_categoria: INT, PK, Autoincremental.
  + nombre\_categoria: VARCHAR(100), NOT NULL, UNIQUE.
  + descripcion: TEXT, NULL.

**3.2. Tabla Proveedor**

* **Propósito:** Registrar información de los proveedores de productos.
* **Columnas:**
  + id\_proveedor: INT, PK, Autoincremental.
  + nombre\_proveedor: VARCHAR(150), NOT NULL.
  + contacto\_nombre: VARCHAR(100), NULL.
  + contacto\_email: VARCHAR(100), NULL.
  + telefono: VARCHAR(50), NULL.

**3.3. Tabla Ubicacion**

* **Propósito:** Definir las ubicaciones físicas dentro del almacén donde se guardan los productos.
* **Columnas:**
  + id\_ubicacion: INT, PK, Autoincremental.
  + codigo\_ubicacion: VARCHAR(50), NOT NULL, UNIQUE (Ej: "A1-S3-N2").
  + descripcion: VARCHAR(255), NULL.
  + capacidad\_maxima: INT, NULL (Ej: en unidades o volumen).

**3.4. Tabla Producto**

* **Propósito:** Almacenar los detalles maestros de cada producto en el inventario.
* **Columnas:**
  + id\_producto: INT, PK, Autoincremental.
  + sku: VARCHAR(50), NOT NULL, UNIQUE (Stock Keeping Unit).
  + nombre\_producto: VARCHAR(200), NOT NULL.
  + descripcion: TEXT, NULL.
  + id\_categoria: INT, NULL, FK -> Categoria(id\_categoria) ON DELETE SET NULL.
  + id\_proveedor\_preferido: INT, NULL, FK -> Proveedor(id\_proveedor) ON DELETE SET NULL.
  + precio\_costo: DECIMAL(10, 2), DEFAULT 0.00.
  + precio\_venta: DECIMAL(10, 2), DEFAULT 0.00.
* **Mapeo de Relaciones (1:N):** La tabla Producto tiene relaciones 1:N con Categoria y Proveedor a través de las claves foráneas id\_categoria y id\_proveedor\_preferido. La opción ON DELETE SET NULL indica que si se borra una categoría o proveedor, el producto no se borra, pero pierde esa asociación.

**3.5. Tabla Stock**

* **Propósito:** Tabla de unión que mapea la relación **Muchos-a-Muchos (M:N)** entre Producto y Ubicacion. Almacena la cantidad actual de cada producto en cada ubicación específica.
* **Columnas:**
  + id\_stock: INT, PK, Autoincremental.
  + id\_producto: INT, NOT NULL, FK -> Producto(id\_producto) ON DELETE CASCADE.
  + id\_ubicacion: INT, NOT NULL, FK -> Ubicacion(id\_ubicacion) ON DELETE CASCADE.
  + cantidad: INT, NOT NULL, DEFAULT 0, CHECK (cantidad >= 0). Stock actual.
  + ultima\_actualizacion: TIMESTAMP, DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP.
  + UNIQUE (id\_producto, id\_ubicacion): Asegura una sola entrada de stock por producto/ubicación.
* **Mapeo de Relaciones (M:N):** Implementa la relación M:N "Producto está almacenado en Ubicacion". ON DELETE CASCADE significa que si se borra un producto o una ubicación, las entradas de stock asociadas se eliminan.

**3.6. Tabla TransaccionInventario**

* **Propósito:** Registrar todos los movimientos de inventario (entradas, salidas, ajustes, traslados) para auditoría y trazabilidad. Implementa **Mapeo de Subtipos** usando la estrategia **Single Table Inheritance (STI)**.
* **Columnas:**
  + id\_transaccion: INT, PK, Autoincremental.
  + id\_producto: INT, NOT NULL, FK -> Producto(id\_producto).
  + id\_ubicacion: INT, NOT NULL, FK -> Ubicacion(id\_ubicacion).
  + tipo\_transaccion: ENUM(...) o VARCHAR, NOT NULL. **Discriminador** que identifica el tipo de transacción (subtipo).
  + cantidad: INT, NOT NULL. Cambio en el stock (positivo o negativo).
  + fecha\_hora: DATETIME, DEFAULT CURRENT\_TIMESTAMP.
  + id\_documento\_referencia: VARCHAR(100), NULL (Ej: N° Orden Compra, Guía, Ajuste).
  + id\_usuario: INT, NULL (FK a una tabla Usuario si existiera).
  + notas: TEXT, NULL.
* **Mapeo de Subtipos (STI):** La columna tipo\_transaccion actúa como discriminador. Todas las transacciones (entradas, salidas, ajustes) se guardan en esta única tabla. Los atributos específicos de cada subtipo (como un id\_orden\_compra para 'ENTRADA' o id\_orden\_venta para 'SALIDA') podrían mapearse a columnas genéricas como id\_documento\_referencia o a columnas adicionales que permitan NULL.

**4. Resumen de Mapeo Aplicado**

* **Mapeo de Relaciones:**
  + 1:N implementado entre Categoria/Proveedor y Producto.
  + M:N implementado entre Producto y Ubicacion a través de la tabla de unión Stock.
  + 1:N implementado entre Producto/Ubicacion y TransaccionInventario.
* **Mapeo de Subtipos:**
  + Implementado en TransaccionInventario usando la estrategia Single Table Inheritance (STI) con la columna tipo\_transaccion como discriminador.

**5. Consideraciones Adicionales**

* **Integridad:** Se debe asegurar la consistencia entre la tabla Stock (cantidad actual) y las TransaccionInventario (historial de movimientos), idealmente mediante triggers o lógica de aplicación.
* **Tipos de Dato:** Se usan tipos específicos como DECIMAL para precios y ENUM (o VARCHAR con CHECK) para tipo\_transaccion.
* **Rendimiento:** Considerar índices en claves foráneas y columnas usadas frecuentemente para búsquedas (ej: sku en Producto, codigo\_ubicacion en Ubicacion, fecha\_hora en TransaccionInventario).

6. Codigo sql completo: