

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

UNIQUE-YANBAL

Estándares de Desarrollo - Base de Datos Relacional

Estado : Borrador
Confidencialidad : Público

YANBAL  UNIQUE

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

Contenido

1.	Historial de versiones	4
2.	Autores	4
3.	Lista de distribución	4
4.	Área interesada/proyecto	4
5.	Resumen ejecutivo	4
6.	Introducción y objetivos.....	5
7.	Definiciones	5
8.	Documentos relacionados.....	5
9.	Excepciones	5
10.	Principios de diseño	6
11.	Estándares y prácticas de base de datos.....	7
11.1.	Comentarios y documentación	7
11.2.	Nomenclatura.....	8
11.3.	Formato y estilo de código.....	10
11.4.	Scripts	11
1.1.	Consultas	11
1.2.	Índices	12
1.3.	Procedimientos almacenados, Jobs y Funciones	12
1.4.	Procesos batch	13
1.5.	Data histórica (archiving)	13
1.6.	Prácticas y consideraciones específicas para DB2	14
1.7.	Prácticas y Consideraciones específicas para SQL Server	14
1.8.	Prácticas y Consideraciones específicas para SQLite (en móviles).....	15
2.	Usuarios y Seguridad.....	15
3.	Herramientas.....	15
4.	Fuentes.....	16

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

1. Historial de versiones

Responsable	Fecha	Versión/Comentarios
Jorge Cabrera	14/01/2015	Versión inicial - borrador
Jorge Cabrera	09/2/2015	Versión Candidata

2. Autores

Nombre	Rol
Martín Valdivia	Revisor/Colaborador - seguridad
Luis Alcántara	Revisor/Colaborador – CID
Joanna Mezaldi	Revisor/Colaborador – PAN
Luis Castro	Revisor/Colaborador – Área de Producción
Jorge Cabrera	Responsable/Elaborador

3. Lista de distribución

Nombre	Cargo /Rol
Centro de Integración y Desarrollo	
Área de Arquitectura y Nueva Tecnología	
Área de Infraestructura Soporte Técnico y Seguridad	

4. Área interesada/proyecto

El presente es elaborado por el área de Arquitectura y Nuevas Tecnologías (PAN), teniendo como áreas interesadas la organización IT corporativa así como todas las áreas corporativas (y de las empresas del grupo) interesadas en desarrollar componentes de base de datos (u aplicaciones y proyectos que los incluyan).

5. Resumen ejecutivo

El cumplimiento de los presentes estándares y prácticas está relacionado directamente a las siguientes características en las aplicaciones:

1. Tiempo de desarrollo
2. Costo y tiempo de mantenimiento
3. Flexibilidad
4. Portabilidad
5. Re-uso
6. Reducción de las incidencias
7. Rendimiento
8. Costo
9. Soporte

Para su cumplimiento es imprescindible el compromiso de los líderes y responsables de la implementación, debiendo brindar los recursos y mecanismos necesarios para esto.

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

6. Introducción y objetivos

El objetivo del presente documento es definir los estándares y las prácticas **a ser cumplidas (de forma obligatoria)** en el desarrollo de componentes de base de datos pertenecientes a las aplicaciones desarrolladas a medida.

Se considera al momento de elaboración las siguientes bases de datos y versiones:

- ✓ DB2 9.x (en adelante)
- ✓ SQL Server 2000 (en adelante.)
- ✓ SQLite (en dispositivos móviles) 3.x

7. Definiciones

A continuación se enumeran las definiciones a ser tomadas en cuenta en el presente documento y anexos:

- ✓ Documento de Diseño Técnico: Documento que detalla el diseño técnico de un aplicativo o solución en particular.
- ✓ Documento de Análisis Técnico: Documento que detalla el análisis técnico de un aplicativo o solución en particular.
- ✓ Documento de Arquitectura: Documento que detalla la arquitectura de un aplicativo ó solución en particular.
- ✓ Practica: Ejercicio realizado bajo ciertas reglas; para los fines de este manual se considera obligatoria.
- ✓ Estándar: Modelo, regla o patrón
- ✓ CID : Centro de integración y desarrollo corporativo Yanbal
- ✓ PAN: Área de proyectos , arquitectura y nuevas tecnologías corporativa de Yanbal
- ✓ ISS : Área de infraestructura, soporte y seguridad de sistemas corporativo
- ✓ Portal Interno Corporativo: Portal interno de colaboradores de la corporación Yanbalnet o YN

8. Documentos relacionados

El diseño de datos, así como de aplicaciones debe de enmarcarse dentro de lo especificado en los siguientes documentos, lo indicado en el presente no implica bajo ningún criterio el no cumplimiento de lo allí definido.

- ✓ GUIA DE SEGURIDAD DE SISTEMAS DE INFORMACION
- ✓ POLITICAS DE SISTEMAS DE INFORMACION
- ✓ PRINCIPIOS DE ARQUITECTURA
- ✓ POLITICAS DE ARQUITECTURA

Dichos documentos pueden ser encontrados en el portal interno corporativo.

9. Excepciones

Sin perjuicio de lo expuesto y reconociendo que una práctica y/o estándar no es aplicable a todos los proyectos y situaciones, **cualquier excepción al presente deberá de ser presentada, justificada y documentada mediante alguno de los siguientes mecanismos de acuerdo a su condición:**

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

1. Excepción aprobada por el proyecto y director del área de seguridad (ISS): Cuando la excepción vaya en perjuicio de los principios y políticas definidas por el área de seguridad.
2. Excepción aprobada por el proyecto y director del área PAN: Cuando la excepción vaya en perjuicio de los principios y políticas definidas por el área.
3. Detallada y justificada en el documento de arquitectura (o equivalente): Cuando se trate de una desviación a los presentes estándares y donde este documento lo indique.
4. Detallada y justificada en el diseño técnico (o equivalente): En los puntos que especifica así el presente documento.

10. Principios de diseño

Los siguientes principios deben ser cumplidos durante el diseño de componentes de componentes de bases de datos:

1. Toda base de datos creada deberá de registrarse en el catálogo de aplicaciones corporativo, esta labor será realizada por el arquitecto líder asignado al proyecto.
2. Las aplicaciones utilizarán procedimientos almacenados para acceder a los datos.
3. Se preferirá el uso de campos tipo 'Identity' (surrogate keys) y deberán denominarse 'ID', el uso de los campos de identidad ofrecerán ventajas de uniformidad y performance (al generar un índice automático con poca fragmentación) y flexibilidad al modelo.
4. El código seguirá el estándar SQL-92, restringiendo el uso de funciones propietarias a las siguientes situaciones :
 - a. Cuando sea necesario para reducir los costos de acuerdo al tipo de elemento (ver punto 5)
 - b. Sea necesario para cumplir un tiempo de respuesta y rendimiento determinado
 - c. Se trate de un trabajo en Job (trabajo en lotes), en donde se preferirá el uso de sentencias propietarias con mejor performance.
 - d. Cuando el estándar no sea correctamente implementado por la Base de Datos
5. Los costos de acuerdo al tipo de elemento deberán mantenerse bajo los siguientes valores, de ser superados deberán de ser identificados en el documento de diseño el cual deberá de ser aprobado por Unique-Yanbal :
 - a. Donde el motor de base de datos no reconozca o implemente el estándar
 - b. Sentencias SELECT máximo **250 timerons** (incluyendo las embebidas dentro de los Procedimientos Almacenados)
 - c. PROCEDIMIENTOS ALMACENADOS máximo **400 timerons**
6. No deberá existir duplicidad de datos (o replicación). De ser necesaria deberá de formalizarse en el documento de arquitectura indicando entre otros aspectos cual será la fuente 'única de verdad' para los datos replicados, así como los mecanismos de sincronización y validación de los mismos.
7. De ser necesario acceder a datos de negocio de otros sistemas se realizará mediante servicios, salvo excepción al presente formalizada en el documento de arquitectura.
8. No utilizar vistas de datos, el uso de vistas requerirá una excepción al presente (indicada en el documento de arquitectura) y será decidido por el arquitecto y/o líder técnico del proyecto (en coordinación con el personal encargado de la administración de bases de datos) en los casos que sea conveniente hacerlo como por ejemplo :
 - a. No afectar un sistema ya desarrollado
 - b. Disminuir la excesiva complejidad de un modelo de datos en particular.

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

- En caso de crearse llevarán el prefijo “VW” (luego del módulo al que pertenecen)
9. El diseño de datos deberá ser descompuesto hasta su quinta forma normal.
 10. Los algoritmos de encriptación están indicados en la **GUIA DE SEGURIDAD DE SISTEMAS DE INFORMACION¹** y deben de utilizarse para los datos indicados allí.
 11. Los requerimientos necesarios para la aplicación, (ej: bases de datos, “schemas” y tablespaces) serán definidos por arquitecto responsable del proyecto, en coordinación con ISS.
 12. No deberán existir valores ‘hardcodeados’.
 13. Deberán utilizarse variables para los valores constantes, declarándose al inicio su valor.
 14. Para los procesos que requieran una transformación y transporte masivo de información se deberá de utilizar un ETS o DTS, según lo definido en la **GUIA DE PRACTICA Y ESTANDARES DE HERRAMIENTAS ETL**, la cual podrá ser ubicada en el portal interno corporativo.
 15. Se debe evitar el uso de nombres específicos a un país para los campos y tablas tales como IGV/IVA debiéndose utilizar el concepto general como IMPUESTO.
 16. No se utilizaran alias globales para las tablas (ej : table alias) u otros objetos de la base de datos, debiendo referenciarse a los mismos por el nombre (definido en su creación) su uso deberá de detallarse en el documento de Arquitectura.
 17. Se deberán evitar las búsquedas sobre columnas y campos encriptados salvo excepción especificada en el documento de arquitectura. Debiéndose evaluar métodos alternativos como índices parciales hexadecimales u árboles para búsquedas por criterios.

11. Estándares y prácticas de base de datos

Los estándares y prácticas a ser cumplidas e detallan a continuación:

11.1. Comentarios y documentación

1. Los objetos correspondientes a tablas y sus datos estarán documentados de manera obligatoria en el documento de diseño (o equivalente) de acuerdo al formato y datos que este indique.
2. Los procedimientos almacenados, funciones y jobs tendrán el siguiente encabezado:

- * Nombre SP: <nombre del SP>
- * Propósito: <explicar en forma detallada >
- * Input: <Parámetro> - descripción de los parámetros
- * Output: <descripción de la salida>
- * Creado por: <responsable>
- * Modificado por: <responsable> - <fecha> <descripción o Requerimiento> (*1 por cada modificación)

¹ Estas pueden ser encontradas en el portal interno corporativo

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

* Fecha. Creación: <fecha creación>

3. Los cambios deberán ser indicados en el código fuente de la siguiente manera (debiendo dejar el código cambiado en comentarios):

* Responsable Cambio: <nombre del responsable>

* Propósito: <explicar en forma detallada el requerimiento o razón del cambio>

* Fecha. Creación: <fecha creación>

11.2. Nomenclatura

Los estándares de nomenclatura para todos los objetos de base de datos a seguir son (no considerar los scripts de creación o carga):

- a. Los nombres para los objetos de BD tendrán un tamaño máximo de 40 caracteres.
- b. Se preferirán los nombres no utilizados actualmente por otra entidad de datos, en caso contrario se deberá evaluar utilizar un código de sistema como prefijo.
- c. Los objetos (de la base de datos) correspondientes a un aplicativo deben utilizar solo mayúsculas
- d. Las tablas correspondientes a un aplicativo se encontrarán emplazados en un esquema 'SCHEMA' de datos propio.
- b. Los nombres de todos los objetos de datos deben ser descriptivos (sin dejar de considerar los prefijos y sub fijos del presente), pudiendo utilizar “_” como separador, salvo el caso del ID o llave primaria y el campo de llave externa o foránea deberá denominarse: ID_<nombre tabla padre>, como por ejemplo :
 - i. FACTURA.ID
 - ii. DETALLE_FACTURA.ID_FACTURA
- c. No utilizar abreviaturas salvo en los casos donde se supere el tamaño máximo para los nombres definido en el ítem a.
- d. Los nombres de las bases de datos corresponderán al sistema al cual pertenecen o información que manejan.
- e. No se deberán utilizar los prefijo o sufijos “db”, “BD”, “DB”, “bd”, “test”, “pru” para los nombres de base de datos.
- f. No utilizar la letra Ñ, ñ , caracteres especiales o tildados.
- g. La letra ñ será reemplazada por ‘NH’ (EJ: COMPANHIA)
- h. Los nombres de campos deben estar en mayúsculas.
- i. El nombre de las funciones y procedimientos almacenados deben indicar su fin.
- j. Los procedimientos almacenados utilizarán el prefijo (USP_XXX)
- k. Las vistas llevarán el prefijo “VW” luego del prefijo del módulo, por tanto este prefijo no deberá de utilizarse en otros objetos (VW_XXX)

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

- l. Las funciones tendrán el prefijo 'FN' luego del prefijo del módulo, no debiendo utilizarse dichas siglas en ningún objeto (FN_XXX).
- m. Los JOBS, llevarán el prefijo JOB
- n. Los objetos temporales llevarán el prefijo "TMP", luego de las siglas del módulo, por tanto solo podrá ser utilizado para este fin (EXT_TMP_XXX).
- o. No están permitidos los sufijos o posfijos "usr", "USR," "tmp","TMP", "bkp","BKP", "bak","BAK", "back".
- p. El nombre de la clave primaria o primary key debe seguir el siguiente formato
 - i. PK_NOMBRETABLA
 1. PK: representa que el objeto es un primary key
 2. NombreTabla: representa el nombre de la tabla
- q. El nombre de la clave foránea o foreign key debe seguir el siguiente formato
 - i. FK_NOMBRETABLA_##
 1. FK: representa que el objeto es un foreign key
 2. NombreTabla: representa el nombre de la tabla
 3. ##:representa el número correlativo del índice
- r. El nombre del índice debe seguir el siguiente formato
 - i. IDX_NOMBRETABLA_##
 1. IDX: representa que el objeto es un índice
 2. NombreTabla: representa el nombre de la tabla
 3. ##:Representa el número correlativo del índice
- s. Con respecto a los procedimientos almacenados se deberá seguir el siguiente patrón :
 - i. USP_<TABLA>_SEL_PK: Consultar un registro (todos sus campos)
 - ii. USP_<TABLA>_SEL_POR_<COLUMNA DE FILTRO> : Consultas utilizando un criterio o filtro
 - iii. USP_<TABLA>_SEL_TODOS: Selección completa sobre una tabla
 - iv. USP_<TABLA>_UPD: Actualización de un registro mediante la llave primaria
 - v. USP_<TABLA>_UPD_POR_<COLUMNA DE FILTRO> : Actualización de uno o varios registros mediante filtros
 - vi. USP_<TABLA>_DEL – Eliminación de un registro mediante la llave primaria
 - vii. USP_<TABLA>_DEL_POR_<COLUMNA DE FILTRO>- Eliminación de uno o varios registros mediante un criterio
 - viii. USP_<TABLA>_INS : Inserción de un registro
 - ix. USP_<NOMBRE DESCRIPTIVO PROCESO> : Para procesos
 - x. USP_OBTENER_SECUENCIA_<NOMBRE DESCRIPTIVO>: Obtener secuencias y correlativos manuales
 - xi. USP_<NOMBRE DESCRIPTIVO QUE INDIQUE EL FILTRO>_CONSULTA : Consultas que retornen un valor

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

- xii. USP_<NOMBRE DESCRIPTIVO QUE INDIQUE EL FILTRO>_LISTA: Consultas que retornen una lista de valores

t. Para los usuarios :

- i. Los nombres para los usuarios de base de datos (LOGINS) deben estar precedidos por el prefijo “USR”.

Ejemplo:

USRMET4

- ii. Para los casos donde sea necesaria la duplicación y sincronización de datos se utilizarán prefijos indicando donde se ejecutarán y/o su direccionalidad, por ejemplo :

- <SIST>_STR_H para los SPs que se ejecutan en el hosting (EXT_STR_H_PEDIDO_INSERTAR).
- <SIST>_STR_L para los SPs que se ejecutan únicamente en el servidor local y no en hosting. (EXT_STR_L_CARGA_PEDIDO)

La replicación de datos deberá de evitarse, sin embargo de ser necesaria será formalizada en el documento de arquitectura, indicando entre otras razones la fuente única de verdad y las razones bajo las cuales se justifica.

***En el caso de objetos no mencionados explícitamente se deberá consultar con el arquitecto sobre la nomenclatura a ser utilizada.**

11.3. Formato y estilo de código

- El código deberá de estar formateado utilizando las funciones provistas por las herramientas de desarrollo (de acuerdo la base de datos, versión y proyecto) con sus valores por default.
- El alias para las tablas (en la sentencia SQL) debe ser el nemónico al nombre de la tabla o campo al cual se referencie.
- Las palabras correspondientes a las sentencias SQL deben ir en mayúsculas.
- Las palabras correspondientes a los comandos propios del SQL deben ir en mayúsculas.

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

11.4. Scripts

Con respecto a los programas tipo script, se deberá considerar:

- a. Se utilizarán para :
 - a. Creación de bases de datos y objetos en ella
 - b. Carga de datos inicial
 - c. Modificación masiva (en el caso que esta solo sea necesaria por única vez)
 - d. Asignación de permisos
- b. Estarán diseñados para ejecutarse en una sola instancia de la consola de comandos.
- c. Utilizarán el carácter “,” como delimitador.
- d. Los scripts de base de datos deberán indicar su correcta ejecución mediante el mensaje “Ejecución Exitosa” o mostrar los errores durante su ejecución.
- e. El nombre de la base de datos deberá de ser indicado mediante parámetro para aquellos que requieran hacer referencia a una.
- f. De la misma manera se parámetrizará el usuario y contraseña del usuario de conexión.
- g. Los scripts indicarán en su cabecera los cambios que realicen de acuerdo a los estándares de comentarios y documentación del presente.
- h. Los scripts deberán :
- i. Tener nombres descriptivos de acuerdo a la tarea de que realicen
- j. Indicar que objetos de datos generen (de ser el caso)
- k. Empezarán con las siglas del sistema al cual corresponden :
 - a. EXT_CREACION_CREATE_TABLAS.sql
 - b. EXT_CREACION_PERFIL_SEL.sql,
 - c. EXT_CREACION_PERFIL_INS.sql,
- l. Se deberá de considerar un script único para la asignación de permisos a los usuarios de base de datos en los casos que así se requiera, el permiso brindado deberá de ser el mínimo indispensable para ejecutar las tareas que dicho usuario requiera.
- m. Las creaciones de base de datos se realizarán con el usuario administrador o dueño de la instancia
- n. En el caso particular de la base de datos DB2 se deberá utilizar el comando RESTRICTIVE en la creación de Bases de Datos.

1.1. Consultas

Las consultas a bases de datos:

- a. No utilizaran la sentencia “SELECT * ” , las columnas seleccionadas deben ser nombradas específicamente.
- b. Utilizar las sentencias JOIN (LEFT, RIGHT, INNER, OUTER) para relacionar tablas, no hacerlo mediante el WHERE
- c. Los JOINS deben darse mediante las llaves foráneas
- d. Ordenar los resultados por alguno de los campos índices utilizando el nombre de la columna (no su posición relativa)

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

- e. Las búsquedas y ordenamientos deberán ser sobre las llaves primarias o campos índices, no debiéndose ejecutar **'full table scan'**, esto se deberá validar mediante el plan de ejecución
- f. No se deberán **traer** (hacer fetch) a un resultado de más de 100 registros por vez, salvo pertenezcan a un proceso en batch
- g. No deberán de traer campos cuyos datos no sean necesarios
- h. No se deberán utilizar comparaciones en el WHERE o JOIN para relacionar entidades.
- i. Utilizar sub-queries donde su uso reduzca el costo de la sentencia, al ejecutarse primero el uso de sub-queries puede reducir el conjunto de datos sobre el cual las siguientes operaciones deben actuar.
- j. No deberán utilizarse concatenaciones ni transformaciones en las comparaciones (luego del WHERE), salvo cuando se utilice para concatenar listas de valores como por ejemplo
 - a. WHERE A.ID NOT IN ('+...
- k. Las variables deberán de ser enviadas por parámetros (parametrizados) en el caso de sentencias SQL dentro de aplicaciones.
- l. Se deberá utilizar UNION ALL (y no SELECT DISTINCT) en los casos donde sepamos que los resultados son únicos.

1.2. Índices

- a. El máximo de índices por tabla será de 10, de ser necesario sobrepasar este número se deberá de detallar en el documento de arquitectura y diseño técnico como una excepción al presente.
- b. Evitar el uso de índices con ordenamiento fijo (Clustered)

1.3. Procedimientos Almacenados, Jobs y Funciones

El uso de procedimientos almacenados, Jobs y Funciones deberá de seguir las siguientes normas:

- a. Las consultas para estos procesos deben seguir los lineamientos de las consultas (ver sección anterior).
- b. El uso de Jobs y Funciones deberá ser considerado una excepción en el documento de diseño y detallado en el mismo.
- c. Errores y Excepciones: El código de los procedimientos almacenados deberá asegurar capturar todos los errores que pudieran ocurrir devolviendo el mensaje adecuado, para esto se debe considerar :
 - a. No ocultar las excepciones raíz, debiendo mostrar la excepción u error específico.
 - b. Utilizar las tablas de auditoria del aplicativo para registrar la excepción u error; en caso no poder escribir en las tablas de auditoria del aplicativo podrán utilizar un archivo para este fin o los mecanismos de registro de eventos de la base de datos.
 - c. Deberán especificar en qué operación ocurrió el error
 - d. Controlar los puntos principales donde pudiera ocurrir un error, como :

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

- i. Ejecución de las sentencias SELECT/DELETE/INSERT/UPDATE
 - ii. Lectura de archivos externos
 - iii. Llamadas a otros procedimientos almacenados, Jobs o funciones.
- e. Controlar los errores esperados en la línea(s) de ejecución de los mismos (Ej : datos incorrectos ó inconsistentes)
- f. Controlar los errores inesperados (no considerados), esto se puede realizar capturando las excepciones del bloque total o de los principales bloques del código.
- d. Deberán definir el nivel de aislamiento (isolation level) de manera explícita de acuerdo a la tarea y las condiciones del mismo.
- e. Deberán asegurar la consistencia de los datos implementando la transaccionalidad adecuada mediante el uso de las operaciones de COMMIT y ROLLBACK parciales o totales de acuerdo al caso.
- f. No repetir funciones o llamadas equivalentes con el mismo resultado.
- g. Utilizarán las sentencias de inserción masiva (bulk insert), para la subida masiva de registros.

1.4. Procesos batch

Los procesos en batch deberán de estar implementadas dentro de procedimientos almacenados (stored procedures) o Jobs, siguiendo los lineamientos de los mismos (ver anterior), además se deberá:

- a. Detallar el uso y frecuencia de ejecución del mismo, así como el proceso en caso de fallas en su ejecución en el documento de diseño (o equivalente)
- b. Deberán asegurar la consistencia de datos implementando la transaccionalidad adecuada
- c. Considerar mecanismos de recuperación y re-inicio en caso de fallas, estos deben estar considerados en el documento de diseño
- d. Los procesos en batch que ingresen más de 250,000 registros considerar re-construir los índices y de acuerdo al caso re-iniciar las estadísticas de las tablas que acceden
- e. Para los procesos en batch considere obtener todos los datos necesarios para el proceso y hacer uso de las sentencias de ingreso y actualización masiva (bulk insert)
- f. Se deberán controlar los errores y excepciones que puedan ocurrir durante la ejecución del mismo devolviendo el mensaje de error adecuado
- g. No repetir funciones o llamadas equivalentes con el mismo resultado

1.5. Data histórica (archiving)

- a. El diseño de la aplicación debe considerar la estrategia y procesos para la migración de datos históricos para aquellas tablas que estimen crecer sobre los

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

500,000 registros por año, esto debe estar contemplado en el documento de diseño, para esto se deberá considerar :

- La base de datos debe ser distinta a la transaccional
- Los datos históricos se accederán (de ser necesario) mediante otra conexión a base de datos.

1.6. Prácticas y consideraciones específicas para DB2

- Preferir el uso de la función COALESCE a VALUE.

1.7. Prácticas y Consideraciones específicas para SQL Server

- Todos los procedimientos almacenados deberán considerar la etiqueta @Debug
- Utilizar la sentencia SET NOCOUNT al inicio de los Store Procedures
- En el caso de los procedimientos almacenados que utilicen parámetros con valores que consideren distintas cardinalidades y una alta variación en la cantidad de resultados que retornan dependiendo de dicho parámetro, deberán utilizar una variable intermedia para dicho fin, como se muestra a continuación :

```
CREATE PROCEDURE [dbo].[USP_TIPO_DOCUMENTO_PAGO_LISTA_]
    @CODIGO_INSTANCIA TINYINT
AS
BEGIN
    SET NOCOUNT ON

    DECLARE @CODIGO_INSTANCIA_AUX TINYINT
    SET @CODIGO_INSTANCIA_AUX = @CODIGO_INSTANCIA

    SELECT
        t.intCodigoInstancia,
        t.intTipoDocumento,
        t.vchNombreDocumento,
        t.vchDescripcion,
        t.chrOpcional,
        t.chrTransferible,
        t.chrAplicaVtaPublica,
        t.dtmFechaActualizacion
    FROM
        usrTipoDocumentoPago t WITH (NOLOCK)
    WHERE
        t.intCodigoInstancia = @CODIGO_INSTANCIA_AUX
```

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

1.8. Prácticas y Consideraciones específicas para SQLite (en móviles)

Las siguientes prácticas deben ser tomadas específicamente en el caso de esta base de datos sobre dispositivos móviles:

- a. Considerar des-normalizar de acuerdo al caso.
- b. Traer solo los campos necesarios (salvo sentencias que sean re-utilizables y similares en los datos que se necesitan).
- c. Realizar búsquedas solo por las PK o índices, así como las sentencias “order by”.
- d. Utilizar joins solo con los campos FK.
- e. No hacer operaciones en los joins.
- f. Uso de transacciones explícitas, es decir no deben de utilizar el AUTO COMMIT o COMMIT IMPLICITO, debe de estar indicado el inicio y fin de la transacción.
- g. Traer solo los registros necesarios (o a mostrar).
- h. No utilizar blobs para archivos e imágenes.
- i. Uso de parametrización de consultas.
- j. No utilizar sub-queries.
- k. Respetar el orden de las columnas de cada tabla para las consultas, además se deben colocar primero las columnas que se soliciten con mayor frecuencia.

2. Usuarios y Seguridad

Se deberán cumplir las siguientes condiciones:

1. Las contraseñas para los ambientes de desarrollo externos que no contengan datos pertenecientes a la corporación podrán ser libres y de conocimiento del equipo de desarrollo.
2. Los usuarios y contraseñas no deben encontrarse en el código fuente de la aplicación, debiendo ser parametrizables (Ej: mediante archivos de texto).
3. En el caso de ser necesario el almacenamiento de las contraseñas en algún archivo, estas deberán de estar encriptados de acuerdo a la GUIA DE SEGURIDAD DE SISTEMAS DE INFORMACION.
4. En la etapa de pruebas y productiva serán gestionadas por Administrador de Base de Datos, quien indicará las pautas de las mismas así como de su política de cambios.

3. Herramientas

Las siguientes herramientas pueden ser utilizadas para verificar el cumplimiento de lo presentado (la lista presentada no es restrictiva ni limitativa, sin embargo las herramientas y versiones a utilizar deberán de ser especificadas por el proyecto):

1. Db2 Explain Tools
2. IBM DB2 Control Center (Centro de Control)
3. IBM Data Studio
4. SQL Server Management Studio
5. Mimer Sql (http://developer.mimer.com/downloads/downloads_thank.tml?id=621)

	Estándares de Desarrollo - Base de Datos Relacional	Código ARQ-EST-DB-001	
		Fecha de emisión 07.01.2015	Versión 01

4. Fuentes

1. <http://www.ibm.com/developerworks/data/library/techarticle/dm-0407tessarek/#UDFs>
2. <http://www.contrib.andrew.cmu.edu/~shadow/sql/sql1992.txt>