



UNIVERSIDAD MAYOR DE SAN SIMON
FACULTAD DE CIENCIAS Y TECNOLOGIA
CARRERA DE INGENIERIA DE SISTEMAS



ESTANDARES A MENEJAR EN EL PROCESO DE DESARROLLO



INTEGRANTES DE GRUPO	:	Abasto Argote Gustavo
	:	Alvarado Llanos José Milton
	:	Ayma Marza Savina
	:	Pool Chavez Jonathan
	:	Ramos Maldonado Abat
	:	Sarmiento Cadima Sergio Daniel
DOCENTE	:	Flores Soliz Juan Marcelo
CARRERA	:	Ingeniería de Sistemas

Cochabamba - Bolivia



ESTANDARES DE NOTACIÓN

ESTANDARES DE DISEÑO DE BASE DE DATOS

Reglas generales:

- Los nombres de tablas y campos deben especificarse bajo el estándar camelCase, este estándar especifica escribir palabras compuestas eliminando los espacios poniendo mayúscula en la primera letra de cada palabra.
- Únicamente se utilizaran caracteres alfabéticos, salvo que por naturaleza del nombre se necesite dígitos numéricos, se prohíbe el uso de caracteres de puntuación y símbolos:

Ej. nombreConferencista

- Las letras acentuadas se reemplazaran con las equivalentes no aceptadas, en lugar de la letra ñ se utilizara ni:

Ej. anioConferencia

- El nombre debe ser lo más descriptivo posible, evitando términos ambiguos o que presenten distintos interpretaciones:

Ej. categorisConferencia => tipoConferencia

- El nombre no debe abreviarse de manera entendible y específica con al menos las primeras tres letras de la palabra.

Tablas:

- Los nombres de las tablas deben especificarse de forma plural y de acuerdo a las reglas generales:

Ej. Estudiantes, conferencistas, cursos, etc.

- Todas las tablas de las BD deben relacionarse con al menos una tabla.



Campos clave (Atributos):

- Toda tabla debe poseer una o más llaves referenciales:

Ej. Tabla Estudiantes (codEst).

- Toda relación entre tablas debe implementarse mediante constraints (Llaves foraneas) con integridad referencial.
- La integridad referencial deberá actualizar en cascada en todos los casos y restringir el borrado, salvo para las entidades débiles.

Ej. No se podrá eliminar un registro de la tabla estudiantes que tenga ocurrencias en otras tablas, si sucediera el caso deberá implementarse el borrado lógico, por el contrario si podrá habilitarse el borrado en cascada, si la relación fuera (tablas estudiantes conferencista).

- Las llaves referenciales deben ubicarse al inicio de la definición de la tabla (Los primeros).
- El nombre de la llave referencial debe estar compuesto por: “id + nombre de la tabla en singular”:

Ej. Tabla conferencista => idConf

Otros campos:

- Todo campo o tributo que presente un nombre o descripción se colocará inmediatamente después de las llaves referenciales:

Ej. Tabla Conferencistas => idConf, nomConf

- En caso de los campos que presenten datos de acuerdo a su representación conceptual en el ámbito del negocio deberá prefijarse de la siguiente manera:

*Numeros: **num** (Ejemplo: numeroEstudiantes => **numEst**)*

*Fechas: **fecha** (Ejemplo: fecha de inscripcion => **fechalnsc**)*

*Codigos: **Codigo** (Ejemplo: Código de estudiante => **codEst**)*

- Los campos de relación (Llaves foráneas) deben nombrarse de la misma manera que las llaves primarias (Usando el nombre de la tabla que hace referencia).



ESTANDARES DE CODIFICACIÓN

Estructura de las clases: Atributos - Constructor - Funciones o métodos.

Declaración de variables:

- Las variables generales deben ser descriptivos, y los locales no necesariamente descriptivos pero si cortos.
- Los nombres de las constantes de las clases deberían escribirse todo en mayúscula con las palabras separadas por un guion bajo ("_").
- En el caso de las variables generales, deben ser encapsulados.

Nombre de los métodos:

- Los métodos deben ser encapsulados.
- Los métodos deberán ser verbos (en infinitivo), en mayúsculas y minúsculas, con las primeras letras del nombre en minúscula, y con las primeras letras de cada palabra interna en mayúscula (lowerCamelCase).
- No se permiten caracteres especiales.
- El nombre debe ser lo suficiente descriptivo, no importando a priori la longitud del mismo.

Comentarios:

- Los comentarios deben hacerse al inicio de cada clase, antes de cada método de la siguiente manera:

```
/*Nombre del método (Descripción)
 * Parámetros que recibe (Descripción)
 * Dato de retorno (Descripción)
 */
```

Los comentarios más cortos se las debe realizar en una sola línea ("//").



ESTANDARES DEL VERSIONADOR

GIT HUB

- Todo el proyecto se trabajará solamente en una rama.
- Las sugerencias dentro el desarrollo deberán se podrán hacer en una rama interna.

La descripción de los commit's deben ser especificas con detalle de los cambios realizados.

REGLAS DE DISEÑO DE INTERFAZ GRÁFICA DE USUARIO

Colores:

- Fondo de las ventanas (R 118, G 204, B 245)
- Letras de las ventanas: (Negro)
- Sub Panels (Items): (R 86, G 192, B 243)
- Botones: (R 193, G 202, B 210)
- Color de la letra de Botón: (R 2, G 10, B 81)

Tamaño de letras:

- Títulos tamaño 25
- Subtítulos tamaño 18
- Párrafos tamaño 14
- Botones 15

Tipo de letra:

- Todos son de tipo "Andale mono"

Botones:

- Todos los botones tienen un mensaje de ayuda al aproximar el puntero del mouse.

Logo de la entidad: Todas las ventanas que puedan ocupar toda la pantalla deben tener el logo de la entidad.

Ventanas de error: En caso de fallos del programa, estos deber mostrarse en una ventana de notificación.

Ventanas de notificación: Todas las ventanas de notificación se las harán mediante la "librería JOptionPane".

Tamaños (ventanas, botones, etc): Manejados por la librería toolkit para que sea multiplataforma.



ESTÁNDAR DE EJECUCIÓN

- OS Windows 7, 8 o 10.

DISEÑO

- Unified Modeling Language (UML)
- Entidad Relación (ER)

ENTORNOS DE DESARROLLO

- JAVA 8
- POSTGRES 9.4

OTROS

- **Patrón de arquitectura de software:** Modelo Vista Controlador (MVC).
- **Editor de código:** “Eclipse luna” o “Eclipse Mars”.

CRITERIOS DE HECHO

- El trabajo de todos los miembros del equipo de desarrollo tiene que estar totalmente integrado en cada iteración.
- El trabajo de cada miembro del equipo ha sido revisado por al menos otro miembro del equipo.
- Todo el equipo considera que para cada objetivo/requisito se cumplen sus Criterios de Validación.
- El trabajo tiene que estar documentado D, E, F.
- El equipo ha validado y aceptado el objetivo/requisito.



REGLAS DE EQUIPO

1. Existen 15 minutos de tolerancia en las reuniones.
2. Los retrasos con más de 15 minutos injustificados serán sancionados con la compra de agua de 50 centavos para cada persona del equipo.
3. Al tercer retraso consecutivo, la persona debe pagar su multa con una coca cola (3 litros).
4. Cada falta injustificada tiene la multa de una coca cola (3 litros).
5. A la tercera falta, la persona también recibirá un memorándum de llamada de atención y una descripción de la sanción en caso de persistir con la misma actitud.
6. La falta de interés (No hace sus deberes y no lo hace conocer al equipo repetidas veces) de algún integrante de grupo le costará la expulsión del equipo.
7. Si una persona del equipo, no puede solucionar alguna tarea, debe hacerlo conocer al resto del equipo con anticipación y no así a “último momento”.
8. Cualquier acto que se interprete como falta de respeto por el equipo significará un memorándum de llamada de atención en el cual también se detallaran las sanciones que serán aplicados en caso de persistir con la misma actitud.