



Vista de Implementación:

1. Aspectos Generales:

a) Clases:

- i. Nombres:** Empiezan por mayúscula todas las primeras letras de las palabras que conforman el nombre de la clase.

EJEMPLO: ControlPanelCompartidos.

- ii. Atributos de la clase:** Empiezan por '_'. La primera letra de la primera palabra en minúscula y la primera letra de las siguientes palabras en mayúsculas.

EJEMPLO: _ejemploCorrectoCorrectisimo, _ejemplo.

- iii. Nombres de Métodos:** Empiezan por minúscula y la primera letra de las siguientes palabras en mayúsculas. No se utilizará '_' entre palabras.

EJEMPLO: getContador(), actualizarEstadoDeAutomata().

- iv. Swing:** Siguen **empezando por** _ seguidos de una información acerca del tipo de componente que es.

EJEMPLO: _btnConectar, _txtEntrada, _lblHola, _panelEstado.

- v. Constantes:** En mayúsculas todas las letras. Si tuviera dos o más palabras estarán separadas por '_'.

EJEMPLO: RUTA_FICHERO_FICHEROSO, RUTA2.

b) Variables:

- i. NOMBRES SIGNIFICATIVOS EN ESPAÑOL** y no importa que sean nombres largos. Empiezan por minúscula y la primera letra de las siguientes palabras en mayúsculas.

EJEMPLO: conexionServidor, contadorIteraciones, archivoTemporal.





c) **Constantes y Enumerados:**

- i.** Usar **public enum** y **NO public static final**. Los nombres de los enumerados que sean **significativos**.

EJEMPLO CORRECTO: public enum EventoPanelCompartidos {INICIO, PULSACION}

EJEMPLO INCORRECTO: public final static int ACCION1.

d) **Comentarios:**

- i.** Todas las clases llevarán comentarios en Javadoc **EN ESPAÑOL**, es decir, cada método de la clase llevará su javadoc separado por la **línea divisoria correspondiente**.
- ii.** **Comentarios Normales:** Cuando haya **algo pendiente añadir la etiqueta TODO:** con una pequeña explicación acerca de la tarea.
- iii.** Para comentarios de **una línea** usar **//** y para comentarios de **más de una línea** usar el **/* */**.

2. **Herramientas de Desarrollo:**

- a) NetBeans 6.1 sobre jdk 1.6.** La razón en particular de ésta versión y no de la anterior es porque **ofrece un mayor rendimiento para hacer los UMLs** ya que **la versión 6.5 no funciona del todo bien en este aspecto**.

NetBeans ofrece también la posibilidad de editar de una forma más fácil y sencilla el manejo de interfaces gráficas, hecho que junto con el anterior acabó convenciendo de su uso frente a la otra herramienta de desarrollo propuesta **Eclipse**.

- b) La codificación** seguida será **UTF-8** para evitar problemas de compatibilidad entre caracteres usados en diferentes plataformas.

