# 

# 

# 

# 

DOCUMENTACIÓN DEL SISTEMA INFORMÁTICO DE LA IMPRENTA

# 

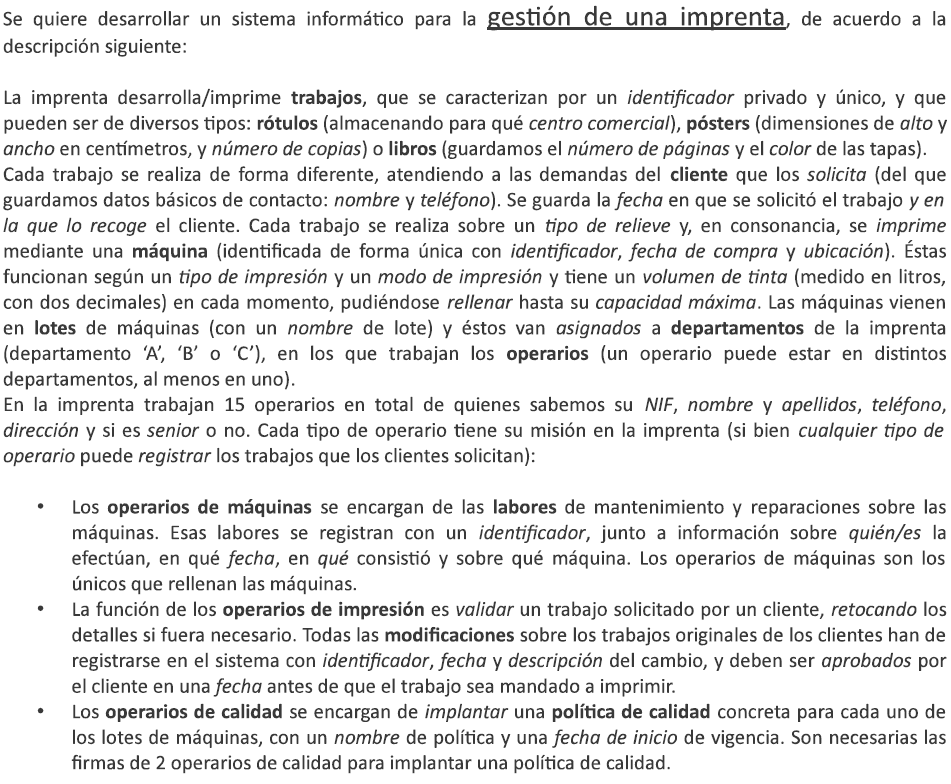
Autores:

Alberto Liñan

Ander Rodríguez

Vindio Pérez

# Sistema a modelar



# Diagrama de clases

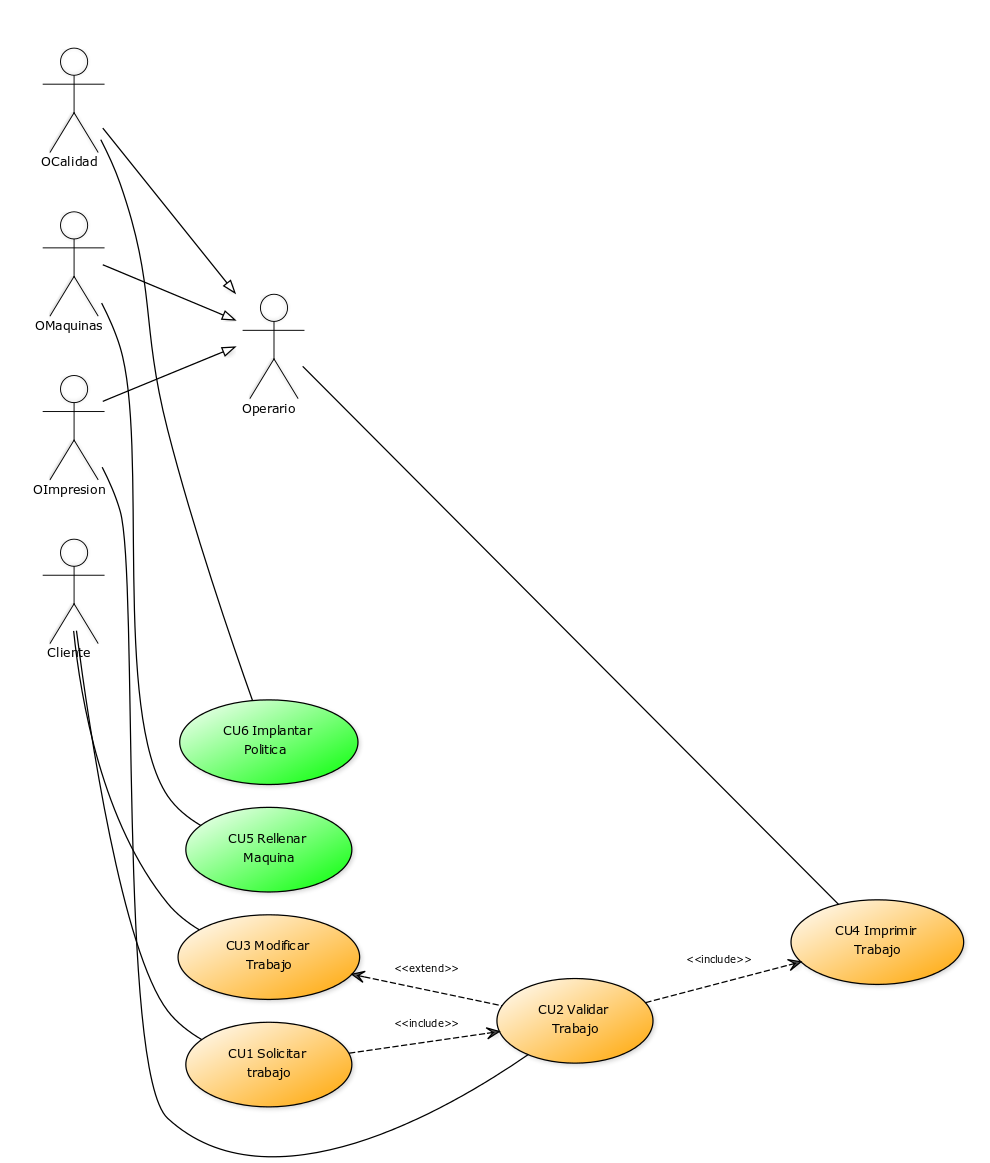
# 

# Explicación del diagrama de clases

Todas las clases tienen un atributo id de tipo long con visibilidad protected, que se generará automáticamente. Además, todas poseen métodos getters y setters, los tres constructores básicos (por defecto, por atributos y de copia), sobreescriben el método toString y tienen un método data que devuelve todos los atributos de la clase separados por “|”. Además, tienen también un método nuevoXXX, que no recibe atributos, y que va pidiendo por la entrada de datos estándar que el usuario introduzca un valor para cada atributo de la clase. Al final del método, se muestran al usuario los datos introducidos y se pide su confirmación, y devuelve un objeto de esa clase. Por último, incluyen un método getAllXXX, que no recibe atributos y devuelve un ArrayList de todos los objetos de la clase; y un método getXXXById, que recibe un id de tipo long y devuelve el objeto de esa clase con ese id, en caso de que exista, o null si no existe.

* Trabajo: recoge las fechas de solicitud y recogida del trabajo en formato Date y el relieve en formato String. Incluye además un atributo de tipo Cliente que es el cliente que encarga el trabajo, y dos de tipo long que reflejan los id del objeto Maquina que imprime el trabajo y del Operario que lo registra.
* Poster: hereda de trabajo, y añade los atributos alto y ancho, de tipo double, y numCopias (el número de copias a imprimir) de tipo int.
* Rotulo: hereda de trabajo, y añade el atributo centroComercial (el centro comercial en el que se instalará el rótulo), de tipo String.
* Libro: hereda de trabajo, y añade los atributos numPag (el número de páginas), de tipo int, y color, de tipo String.
* Cliente: incluye los atributos nombre y telefono, de tipo String.
* Maquina: incluye los atributos ubicacion, imprTipo (el tipo de impresión) e imprModo (el modo de impresión), de tipo String; volTinta (el volumen actual de tinta de la máquina) y capMax (la capacidad máxima), de tipo float; y fechaCompra (la fecha de compra), de tipo Date.
* Lote: incluye el atributo nom (el nombre del lote), de tipo String, así como un atributo de tipo Departamento, que refleja el departamento al que pertenece el lote, y un ArrayList de objetos de tipo Maquina, que son las máquinas que componen el lote.
* Departamento: incluye el atributo letraDep (la letra de departamento) de tipo char, así como un ArrayList de objetos de tipo Operario, la lista de operarios que trabajan en el departamento.
* Operario: incluye los atributos nombre, apellidos, nif, telefono y direccion, de tipo String, y senior, de tipo boolean, que recoge si el operario es o no senior. También incluye un ArrayList de objetos de tipo Departamento, la lista de departamentos en los que trabaja el operario.
* OCalidad: hereda de Operario. No añade atributos propios.
* OMaquinas: hereda de Operario. No añade atributos propios.
* OImpresion: hereda de Operario. No añade atributos propios.
* Politica: incluye los atributos nom (el nombre de la política), de tipo String, y fechaIni (la fecha de implantación de la política), de tipo Date. También incluye los atributos idOCalidad1 e idOCalidad2 (los id de los dos operarios de calidad que aprueban la política), de tipo long, y otro de tipo Lote, el lote sobre el que se implanta la política.
* Labor: incluye los atributos fecha, de tipo Date, y tarea, de tipo String. También incluye un atributo de tipo Maquina (la máquina sobre la que se realiza la labor) y un ArrayList de objetos de tipo OMaquinas, los operarios de máquinas que han realizado la labor.
* Modificación: incluye los atributos fecha (la fecha de realización de la modificación) y fechaAprob (la fecha en la que se aprueba la modificación), de tipo Date; desc (la descripción), de tipo String; aprob, de tipo boolean, que recoge si el cliente ha aprobado o no la modificación. También incluye los atributos idCliente e idTrabajo, de tipo long, que reflejan el cliente y el trabajo con los que se relaciona la modificación, y un ArrayList de objetos de tipo OImpresion, la lista de operarios que han realizado la modificación.

# Diagrama de Casos de Uso



# Subsistemas del Diagrama de Casos de Uso

* Subsistema Naranja:El **cliente** *solicita* untrabajo, el trabajo se *valida,* si el trabajo se *valida*este es *registrado* por un **operario de impresión**, el **cliente** puede *solicitar una modificación*, si la modificación *es validada* por el **operario de impresión***,* a continuación *será registrada.*
* Subsistema Verde: el **operario de máquinas** se encarga de *rellenar* la máquina que realizará la impresión del trabajo. El **operario de calidad** se encarga de *implantar* las políticas de calidad de las máquinas que realizarán el trabajo.

# Explicación Diagrama de Casos de Uso

## CASOS DE USO

#### CU1 Solicitar Trabajo

En este Caso de Uso, el Cliente Solicita un Trabajo y éste es registrado por un Operario

#### CU2 Validar trabajo

En este Caso de Uso, el Oimpresion revisa el trabajo y lo valida o no, en caso positivo, se pasa al CU4 y en caso negativo se pasa al CU3.

#### CU3 Modificar Trabajo

En este Caso de Uso, el OImpresion realizará los cambios pertinentes al Trabajo y lo mandará al Cliente para que de su aprobado, en caso afirmativo se pasa al CU4 , en caso negativo se repite el CU3.

#### CU4 Imprimir Trabajo

En este Caso de Uso el OImpresion, con el trabajo ya validado, lo mandará a imprimir

#### CU5 Rellenar Máquina

En este Caso de Uso el OMaquinas rellenara el volumen de tinta de la Máquina, no pasando nunca de su volumen máximo.

#### CU6 Implantar Política

En este Caso de Uso el OCalidad implantará una Política en un Lote

## ACTORES

#### Operario:

De este actor heredan el resto de operarios, puede registrar trabajos

#### OMaquinas:

Se encarga de rellenar la Maquina

#### OCalidad:

Se encarga de implantar la politica de calidad

#### OImpresion:

Se encarga de validar los trabajos, proponer modificaciones y mandar los trabajos a imprimir

#### Cliente:

Se encarga de solicitar trabajos así como de aprobar las modificaciones planteadas por el OImpresion

### Reparto de tareas

**V 1.0**

Diagrama de clases: Ander y Vindio

Implementación:

* Cliente:Ander
* Departamento:Vindio
* Imprenta:Ander
* Labor:Alberto
* Libro:Ander
* Lote:Alberto
* Máquina:Alberto
* Modificación:Alberto
* Operario de Calidad:Vindio
* Operario de Impresión:Vindio
* Operario de maquinas:Vindio
* Operario:Vindio
* Política:Alberto
* Poster:Ander
* Rótulo:Ander
* Trabajo:Ander

**V 1.1**

Diagrama de casos de uso:Alberto

Diagrama de clases: Ander y Vindio

Implementación:

* Cliente:Ander
* Departamento:Alberto
* Imprenta:Alberto
* Labor:Alberto
* Libro:Vindio
* Lote:Alberto
* Máquina:Alberto
* Modificación:Ander
* Operario de Calidad:Ander
* Operario de Impresión:Ander
* Operario de maquinas:Ander
* Operario:Vindio
* Política:Alberto
* Poster:Vindio
* Rótulo:Vindio
* Trabajo:Ander

**Documentación:** Alberto, Ander y Vindio