# Diagrama de estado Almacenar GuíasDiagramaAlmacenarGuia.jpeg

# Escenarios

* Un profesor solicita almacenar una guía, introduciendo archivo y asignatura válidos.
* Un profesor solicita almacenar una guía, introduciendo archivo inválido y asignatura válida.
* Un profesor solicita almacenar una guía, introduciendo archivo válido y asignatura inválida

# Estados:

1. **Creación almacenar guías**: Este estado permite la recolección de todos los datos que van a caracterizar a este tipo de solicitud. Es estado inicial.

Eventos:

* **solicitar\_guía**: Recibe como parámetro el objeto de almacenar guía que es un tipo de solicitud, por lo que, contiene almacenados únicamente los campos correspondientes a la clase Solicitud de Servicio. Permite al profesor iniciar el proceso de almacén de la guía.

Transiciones:

* **infoAlmacenarGuías**: Cambia hacia estado **Validando información almacenar guías.** Recibe como parámetro el objeto Almacenar\_Guías, y los datos relevantes para crear una guía: archivo, nombre, asignatura, tema y categoría. Este evento ocurre cuando ya se ha obtenido toda la información sobre la guía

Actividades:

* **Solicitar información guía**: Esta actividad consiste en recibir los datos de una guía e irlo guardando en variables temporales para próximamente ser validados, en otro estado.

1. **Validando información almacenar guías:** Este estado realiza la validación de los datos recolectados en el estado **Creación almacenar guías** para así poder verificar que se creará una guía válida.

Eventos: Evento de entrada explicado como transición en el estado **Creación almacenar guías.**

Transiciones: Los eventos son implícitos y responden a una condición basada en alguno de los siguientes atributos booleanos de almacenar guía: “asignaturaVálida” y “archivoVálido”.

* + En el caso en que asignaturaVálida evalúa falso cambia hacia estado **Creación almacenar guías.**
  + En el caso en que archivoVálido evalúa falso cambia hacia estado **Creación almacenar guías**
* **Validado**: Este evento recibe como parámetro todos los datos que siguen almacenados en variables temporales que en el caso en que asignaturaVálida y archivoVálido ambos evalúan verdadero entonces se procederá al estado **Procesando información almacenar guías,** en donde se almacenarán definitivamente dentro del objeto almacenar guías (referenciado como self).

Actividades:

* Validar asignatura: Esta actividad consiste en corroborar que la asignatura que se pasó por parámetro es una asignatura existente en la base de datos.
* Validar archivo: Esta actividad consiste en verificar que el archivo recibido en el evento de transición de estado no está corrupto y posee un formato legible.

1. **Procesando información almacenar guías:** En este estado, ya habiendo validado la información de la solicitud de almacenar guía, entonces se procede a guardar estos datos y realizar cálculos sobre ellos.

Eventos:

* **Entry**: El evento de entrada que consiste en el evento de la transición **validado** explicado en el estado anterior.

Acción:

* + - **asignarPropiedadesAlmacenarGuías**: Consiste en asignar los valores de los atributos de la clase Almacenar guías: nombre\_guía, tema\_guía, categoría\_guía y archivo con los datos pasados en las variables temporales.
* Exit: El evento al salir del estado.

Acción:

* + - actualizarPrecio: Consiste en asignar al atributo: precio de la clase AlmacenarGuía (Solicitud de Servicio) con el valor calculado del costo de reproducción de la guía (resultado de la actividad **Calcular costo reproducción guía**).

Transiciones:

* cambioPrecio: Dado que el precio de la solicitud de servicio fue actualizado y éste tiene un valor mayor que 0 (monto válido) entonces se cambia a estado **Almacenar Guías guardada.**
  + Esta transición desencadena un evento que es enviado a las instancias existentes de la clase Guía, llamado **crear\_guía** pasando por parámetro el objeto Almacenar\_Guía (self) y la asignatura con la cual la guía se va a asociar. Este evento está documentado en su completitud en la documentación del diagrama de estado de la clase Guía.

Actividades:

* **Calcular costo reproducción guía**: Consiste en que a partir del número de páginas que contenga el archivo seleccionado y el precio por página (dado por la clase Servicio) entonces se calcula el costo de la reproducción de la guía

1. **Almacenar guías guardada:** Este estado representa que el Almacenar\_Guías contiene toda la información necesaria y ya fue almacenada correctamente. Es estado final.

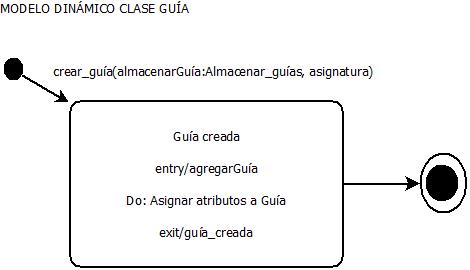
Eventos:

* Guía\_creada: Este evento se dispara en el momento en que la Guía fue creada satisfactoriamente y esta última envía una acción llamada así hacia este diagrama de Estado.

Acción:

* + - Modificar\_dato\_entrega: Dado que la guía fue creada, entonces se puede dar por terminada la solicitud del servicio de tipo Almacenar\_Guías. Para señalar esto, entonces se modifica el atributo llamado dato entrega que representa la fecha y hora en que se concreta el servicio llamado.

# Diagrama de estado Guía



1. **Guía creada:** Este estado es inicial y final, se inicia creándose una instancia de la clase Guía y esta no puede cambiar de estado a partir de ahí

Eventos:

* **Crear\_guía**: Recibe como parámetro un objeto de almacenar guía y una asignatura con la cual la guía se almacena. Evento de entrada.

Acción:

* + - agregarGuía: Al recibir los datos, entonces se crea una instancia de la clase Guía
* **exit**: Evento de salida

Acción:

* + - guía\_creada: Asocia a la Guía creada con la asignatura pasada por parámetro.

Actividades:

* **Asignar atributos a Guía**: A la guía recién creada se le asigna a sus atributos los correspondientes en la clase Almacenar\_Guías, por ejemplo: nombre\_guía se le asigna a nombre. Al costo de reproducción se le asigna el valor que se tiene de precio.