



CU08 – Crear horarios de cursada

TEMPUS



Un Caso de Uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema.

Estos ilustran los requerimientos del sistema al mostrar cómo reacciona una respuesta a eventos que se producen en el mismo

Las Realizaciones de los Casos de Uso se llevan a cabo como resultado de un caso de uso específico. La realización del caso de uso debe cumplir con los requerimientos establecidos y debe reflejar el comportamiento de su caso de uso correspondiente. Este artefacto se halla dentro del Modelo de Diseño reflejando los productos de trabajo relacionados con el caso de uso pero que pertenecen a dicho modelo. Estos productos de trabajos relacionados consisten en los diagramas de comunicación y secuencia que expresan el comportamiento del caso del uso en términos de objetos de colaboración, y dichos diagramas deben elaborarse haciendo uso de (UML).



Tabla de contenido

[Descripción 4](#_Toc257615429)

[Actores del CU 4](#_Toc257615430)

[Precondiciones 4](#_Toc257615431)

[Flujo de Eventos Normal 4](#_Toc257615432)

[Poscondiciones 4](#_Toc257615433)

[Flujo de Eventos Alternativo 4](#_Toc257615434)

[Diagramas Asociados 4](#_Toc257615435)

[Diagrama de Casos de Uso 5](#_Toc257615436)

[Diagrama de Secuencia 5](#_Toc257615437)

[Diagrama de Colaboración 5](#_Toc257615438)

[Diagrama de Estados 5](#_Toc257615439)

CU08 – Crear horarios de cursada

Descripción

El objetivo del caso de uso es crear horarios de cursada. Este caso de uso, puede ser invocado por el actor administrador y secretaría académica.

Una característica de este caso de uso es que se puede invocar desde el sitio web del sistema Tempus.

Actores del CU

1. Administrador.
2. Secretaría Académica.

Precondiciones

Se debe haber iniciado sesión en el sistema.

Flujo de Eventos Normal

1. El actor presiona “Crear” en la Interface Usuario.
2. La Interface Usuario despliega la Pantalla Crear Cursada.
3. La Pantalla Crear Cursada se despliega. Esta pantalla contiene los siguientes campos:

* Carrera: Campo de ingreso de texto obligatorio.
* Asignatura: Campo de ingreso de texto obligatorio.
* Año: Campo de texto obligatorio.
* Día: Desplegable obligatorio.
* Hora Inicio: Desplegable obligatorio.
* Sector: Campo de texto obligatorio.
* Hora Fin: Desplegable obligatorio.
* Nombre: Campo de texto obligatorio.

1. El actor completa el formulario y presiona “Crear”.
2. La Pantalla Crear Cursada envía el evento “Crear” al Manejador Cursada.
3. El Manejador Cursada solicita crear (cursada) a Cursada.
4. Cursada solicita crear (cursada) a la Interface BD.
5. La Interface Base de Datos crea el nuevo registro en la Base de Datos.
6. La Base de Datos devuelve ok a la Interface Base de Datos.
7. La Interface Base de Datos devuelve ok Cursada.
8. Cursada devuelve el resultado de la operación al Manejador Cursada
9. El Manejador Cursada solicita desplegar Pantalla Resultado Crear Cursada.
10. La Pantalla Resultado Crear Cursada se despliega. Esta pantalla muestra el resultado de la operación de creación.
11. El caso de uso finaliza.

Poscondiciones

Se ha creado el nuevo horario de cursada y se encuentra almacenada en la base de datos.

Flujo de Eventos Alternativo

Paso 6 – El horarios de cursada existe

1. El Manejador Cursada solicita desplegar Pantalla Resultado Crear Cursada.
2. La Pantalla Resultado Crear Cursada muestra el siguiente mensaje “El horario de cursada ya existe.”
3. El caso de uso finaliza.

El actor puede cancelar la operación en cualquier paso del caso de uso.

Diagramas Asociados

Diagrama de Casos de Uso

Se observa a continuación una extracción del diagrama de casos de uso del sistema. Se puede ver como los actores interactúan con el presente caso de uso y la relación que este tiene con el caso de uso ingresar al sistema.

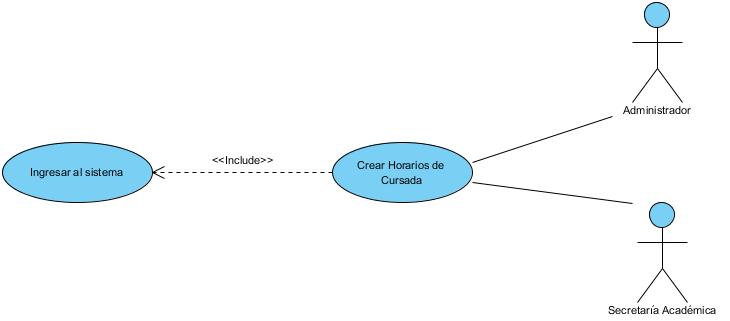


Diagrama de Secuencia

[En el diagrama de secuencia se muestra la interacción de los objetos que componen un sistema de forma temporal.

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada método de la clase]



Diagrama de Colaboración

[Un diagrama de colaboración, se puede decir que es una forma alternativa al diagrama de secuencias a la hora de mostrar un escenario.

Este tipo de diagrama muestra las interacciones que ocurren entre los objetos que participan en una situación determinada. A diferencia del diagrama de secuencia, el diagrama de colaboración se enfoca en la relación entre los objetos y su topología de comunicación. En estos diagramas los mensajes enviados de un objeto a otro se representa mediante flechas, acompañado del nombre del mensaje, los parámetros y la secuencia del mensaje.

Estos diagramas están indicados para mostrar una situación o flujo de programa específico y son considerados uno de los mejores diagramas para mostrar o explicar rápidamente un proceso dentro de la lógica del programa. ]



Diagrama de Estados

[Un estado es una condición durante la vida de un objeto, de forma que cuando dicha condición se satisface se lleva a cabo alguna acción o se espera por un evento. El estado de un objeto se puede caracterizar por el valor de uno o varios de los atributos de su clase, además, el estado de un objeto también se puede caracterizar por la existencia de un enlace con otro objeto.

El diagrama de estados engloba todos los mensajes que un objeto puede enviar o recibir, en otras palabras es un escenario que representa un camino dentro de un diagrama.

Como característica de estos diagramas siempre cuentan con dos estados especiales, el inicial y el final, con la particularidad que este diagrama puede tener solo un estado inicial pero varios estados finales. Una transición entre estados representa un cambio de un estado origen a un estado sucesor destino que podría ser el mismo que el estado origen, dicho cambio de estado puede estar aparejado con alguna acción. Además las acciones se asocian a las transiciones y se consideran que ocurre de forma rápida e ininterrumpible.]

