Requerimientos e introducción a Casos de uso

# Requerimientos

## Propósitos importantes

El sistema debe:

1. Registrar los pagos de los boletos.
2. Mantener un registro de los horarios de las funciones de cine
3. Estar conectado solamente a los dispositivos usados en las taquillas disponibles.
4. Guardar un registro de la disponibilidad de asientos en las salas, y de inventario en la dulcería

## Reglas de negocio

* BR01: La caja sólo se puede abrir con la emisión de boletos
* BR17: Las dulcerías
* BR25: La venta de boletos para funciones de mayores de 18 años a menores de edad no está permitida
* BR47: En selectos días de la semana, hay descuentos para estudiantes que presenten identificación escolar
* BR93: En funciones de medianoche, el precio del boleto aumenta.

## Requerimientos no funcionales

**Sin conocimiento** del producto de software: *Todos los algoritmos de búsqueda (disponibilidad, horarios, inventario, o similares) no deben tener una complejidad mayor a O(n2):* Sin saber el producto de software a realizar, es importante considerar que la velocidad de búsquedas debe ser lo más alta posible. Si bien pueden no hacer la diferencia en el día a día, menor tiempo de ejecución significa menor estrés para la computadora.

**Con conocimiento** del producto de software: *Las claves de acceso a las máquinas, y conexiones con servidores deben estar cifradas en RSA:* Aquí estamos asumiendo que el sistema tendrá un sistema de acceso tanto para usar la terminal como para conectarse con otras computadoras en el sistema. Independientemente del cifrado a utilizar (en este caso es RSA), es necesario reducir la posibilidad de que alguien aproveche vulnerabilidades en el sistema para ingresar sin autorización.

# Casos de uso

## Ambigüedades y omisiones

1. La primera oración repite la palabra boletos.
2. Las primeras oraciones hacen referencia a un uso del sistema y las subsecuentes a otro, esto puede causar conflictos al lector, ya que se puede pensar que es el mismo. Sin embargo, estas oraciones infringen con las restricciones del sistema: no se menciona que se limitan los destinos, la validación de la tarjeta de crédito infringe con la normativa de movimientos bancarios vía electrónica.
3. ¿Qué identificaciones personales y qué tarjetas de crédito son válidas?
4. ¿Puede un menor de edad comprar un boleto dado que no tiene identificación personal?
5. ¿Cómo es que se ingresa la tarjeta y la identificación?
6. ¿Qué pasa cuando algo no es válido?

## Resolución de ambigüedades y omisiones

1. La oración debe ser cambiada por: En un sistema automatizado de emisión de boletos de ferrocaril. Y dejarla como oración yuxtapuesta con la siguiente.
2. Estas oraciones son suprimidas después de hablar con el cliente: nos da su visto buenos
3. Después de aclarar esto con el cliente, se sabe que las tarjetas de crédito deben de tener el logo de visa o el de mastercard en el plástico. Mientras que las identificaciones están limitas a la identificación de ciudadanía, pasaporte y licencia de conducir.
4. Al hablar sobre punto, el cliente solicitó más funciones para el sistema, por lo que el día 4 de marzo se acordó tener una nuevo reunión para esclarecer lo relacionado a que un menor pueda obtener boletos.
5. El cliente delimitó que hay un escaner cuadrado de quince centímetros de lado, en donde se leen las identificaciones. También hay un lector de tarjetas de crédito con la normativa actual del Estado.
6. Despliega un mensaje con el motivo de la invalidación y regresa a la primer pantalla

*Requerimientos Funcionales*

* *Mediante el pulso del botón de inicio, se desplieagan los destinos posibles con respecto a esa estación, junto con un mensaje para solicitar que se seleccione un destino.*
* *Al seleccionar un destino, se despliega un mensaje para solicitar que introduzca una tarjeta de crédito, se muestra un diagrama de cómo debe realizarlo.*
* *Al insertar una tarjeta, se valida la información, cuando es válida, se despliega un mensaje para solicitar que ponga una identificación personal, se enlista los posibles a usar y tres esquemas de cómo debe de mostrar cada una de las identificaciones en un escaner.*
* *Se valida la transacción crediticia, cuando es válida, se emite un boleto.*
* *En todas las validaciones, si resultan inválidas, se despliega un mensaje explicando los motivos y se regresa a la primera pantalla.*

## Caso de Uso: qué es

En esencia, un caso de uso narra una historia estilizada sobre cómo interactúa un usuario final (que tiene cierto número de roles posibles) con el sistema en circunstancias específicas. La historia puede ser un texto narrativo, un lineamiento de tareas o interacciones, una descripción basada en un formato o una representación diagramática. Sin importar su forma, un caso de uso ilustra el software o sistema desde el punto de vista del usuario final. Los casos de uso se definen desde el punto de vista de un actor. Un actor es un papel que desempeñan las personas (usuarios) o los dispositivos cuando interactúan con el software [1].

## Caso de Uso y requisitos funcionales: relación

Cuando una persona se encarga de los requerimientos y de los casos de uso, se da cuenta que lo que se hace es una traducción, del lenguaje que comprende el cliente al lenguaje que comprende a uno que comprenden los desarrolladores (en la especificación de caso de uso) o para personas entre el lenguaje del cliente y el lenguaje del desarrollador (en los diagramas de caso de uso). En general, uno es la representación del otro, pero con un nivel de lenguaje diferente.

## usuario y actor: Diferencias

Un actor es una persona o sistema que tiene un papel específico, un actor es un tipo de usuario, y un usuario es cualquier cosa que intercambia datos con el sistema. Por ejemplo, un mismo usuario puede tener más de un tipo de usuario, como por ejemplo, un administrador de clientes y un administrador de transacciones. Aunque son dos tipos de usuarios (por ende, de actores), resulta ser el mismo usuario. Un usuario es la *instancia* de un actor, si lo vemos con un paradigma orientado a objetos. [2]

## Caso de Uso y diagrama de caso de uso: Diferencias

Uno no requiere de tantas palabras, es mucho más comprensible para personas que no tienen un lenguaje a nivel de la programación, resulta muy útil para personas que no hicieron la codificación, pero que tienen relación con asegurar la calidad del sistema, por ejemplo, cuando se contratan servicios de pruebas, las personas que brindan estos servicios solicitan la documentación del sistema, en particular los casos de uso, y generalmente los diagramas. Por otro lado, el caso de uso suele ser escrito con una precisión alta y con un lenguaje orientado a los programadores. Entre ambos (el caso de uso y el diagrama de caso de uso), resulta existir un *tradeoff*, en específico, comprensión – ambigüedad, mientras más comprensible sea más ambiguo resulta: el diagrama es más comprensible, pero resulta ambiguo en términos de programación, por otro lado, el caso de uso resulta muy puntual en qué sucede pero resulta menos comprensible para una persona que no habla la jerga de la programación.

Referencias

[1] Pressman, R. (2010). Ingeniería del software. 7th ed. México: McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V., p.113.

[2] Richardson, M. (2020). Guideline: Actor. [En línea] Michael-richardson.com. Disponible en: http://www.michael-richardson.com/processes/rup\_classic/core.base\_rup/guidances/guidelines/actor\_6894F48D.html [Acceso 28 feb. 2020].