



Universidad de Guanajuato

División de Ingenierías Campus Irapuato Salamanca

Licenciatura en Ingeniería en Sistemas Computacionales

Practica Individual

Materia: Sistemas de información

Impartida por: Dr. Juan Carlos Gómez Carranza

Alumno: Cesar Manuel Padron Garcia

NUA: 775206

Fecha de entrega: 29 de mayo del 2020

Introducción

Un sistema de información es un conjunto organizado de elementos, que pueden ser personas, datos, actividades o recursos materiales en general. Estos elementos interactúan entre sí para procesar información y distribuirla de manera adecuada en función de los objetivos de una organización. Los sistemas tienen 4 funciones básicas:

Entrada de información; Es el proceso mediante el cual el sistema de información toma los datos que requiere para procesar la misma. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos.

Procesamiento de la información; Es la capacidad del Sistema de Información para efectuar operaciones de acuerdo con una secuencia de pasos preestablecida. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada por la organización.

Almacenamiento de información; Es una de las capacidades más importantes que tiene un sistema, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada mediante bases de datos con estructuras bien definidas.

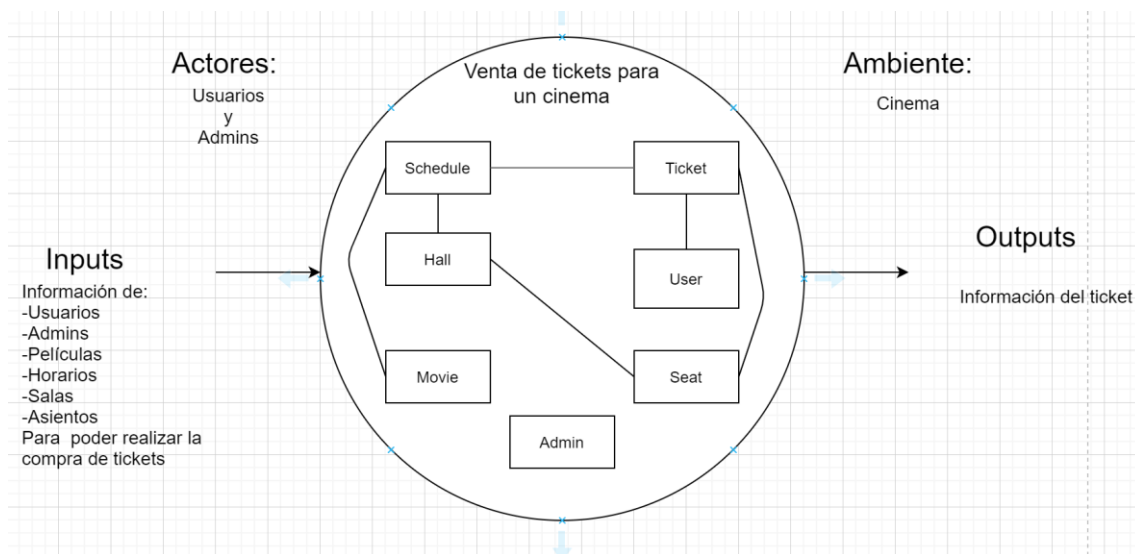
Salida de información; Es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. De hecho, la salida de información de un sistema puede ser la entrada de otro sistema.

En esta practica se realizará el sistema de información para un cine, donde se tendrán 2 tipos de usuarios, usuario común (cliente) y administradores, donde el administrador tendrá habilitada todas las operaciones en el sistema mientras que el usuario solo tendrá acceso a algunas. El fin del sistema es poder permitir a un administrador tener control sobre el cine (películas, horarios, salas). Para el cliente el sistema tiene como fin permitirle ver horarios disponibles y adquirir tickets para alguno de estos.

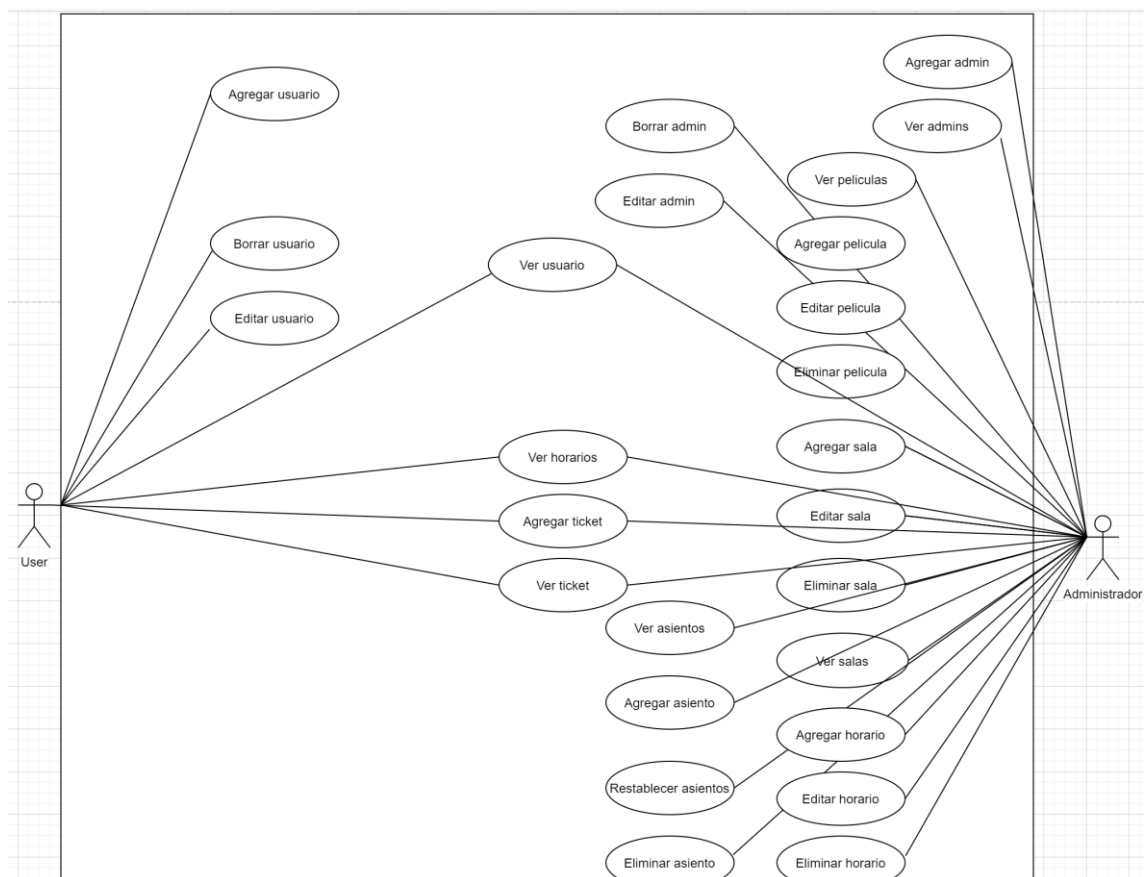
Desarrollo

Para desarrollar un sistema de información, es importante definir bien la estructura de esto antes de comenzar a programar, para esto se realizan tres tipos de diagramas; modular, casos de uso y relacional.

El primero es el diagrama modular, en este se especifican las entradas, salidas, actores, y ambiente del sistema. Los inputs se introducen en las distintas tablas, y la relación entre estas mismas debe estar especificada. No todos los actores tendrán acceso a todas las tablas, sin embargo, todos podrán generar salidas con las operaciones que se les permita.



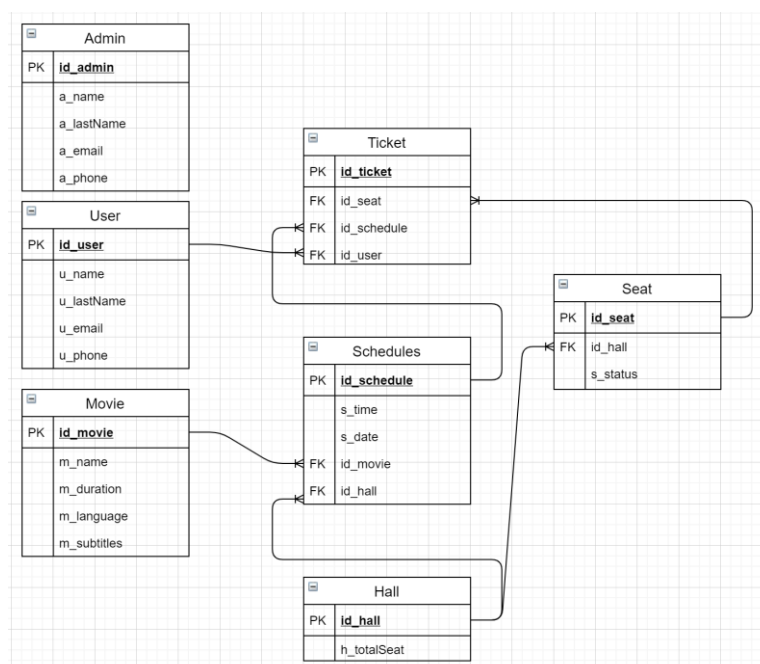
Para el diagrama de casos de uso cada actor (usuarios/admins) tiene definidas todas las acciones que puede realizar en el sistema. Es importante saber la finalidad del sistema para poder asignar correctamente las operaciones que cada actor tiene permitido realizar.



Para explicar un caso de uso tomaremos la operación de Agregar horario, evidentemente esta operación puede ser realizada solo por un admin, la operación agregara un horario a la lista de horarios existentes, en este se deberá incluir la película y sala donde se proyectara, al igual que le fecha y hora de la proyección.

Nombre:	<ul style="list-style-type: none"> • Agregar horario.
Suposiciones:	<ul style="list-style-type: none"> • Sea administrador.
Precondiciones:	<ul style="list-style-type: none"> • Tener al menos una película • Tener al menos una sala
Inicio:	<ul style="list-style-type: none"> • Click sobre la opción "Add schedule"
Flujo de eventos:	<ul style="list-style-type: none"> • S. Muestra un menú sobre agregar horario. • A. Ingresa la fecha y hora del horario, id de la película y sala. • S. Verifica que todos los datos sean validos. • S. Muestra un mensaje de confirmación al haber registrado el horario. • S. Se redirige al menu de horarios
Postcondición:	<ul style="list-style-type: none"> • Se actualiza la base de datos.

Por último, el diagrama relacional, lo primero es establecer todas las tablas necesarias para el manejo de toda la información necesaria del sistema. Para comprobar que las tablas y los elementos de cada una son correctos, se debe de aplicar las 3 formas de normalización con el fin de eliminar la redundancia de datos junto con cualquier anomalía indeseable que pueda surgir al insertar, actualizar o borrar datos. Depende el sistema puede no ser necesario no aplicar las 3 formas, sin embargo, este no fue el caso, por lo que se aplicaron todas.



Comenzando por el actor usuario, como se puede observar solo esta relacionado con la tabla ticket, donde el tipo de relación es uno a varios, ya que un usuario puede tener varios tickets pero un ticket no puede tener varios usuarios. Aparte de la tabla User, esta es la única otra tabla donde puede hacer operaciones que sería el crear un ticket, aunque puede ver horarios y en este películas o asientos, no tiene la posibilidad de modificar estas tablas.

Por otro lado, tenemos el actor admin, aunque este puede realizar todas las operaciones para todas las tablas, estas no dependen de quien las modifique, por esa razón, el admin no está relacionado directamente con ninguna tabla.

La tabla de películas tiene relación uno a varios con la tabla de horarios, debido a que una película puede aparecer en distintos horarios pero un horario no puede tener distintas películas.

Por otro lado la tabla de asientos tiene dos relaciones, la primera es con la tabla de salas, la relación es uno a muchos, ya que una sala tiene muchos asientos, pero un asiento no pertenece a muchas salas. La otra relación de la tabla asientos es con ticket, y esta es de uno a muchos, ya que un asiento puede aparecer en muchos tickets pero un ticket no puede tener muchos asientos. Esta relación es necesaria para poder asignar el asiento al ticket.

La tabla de sala tiene igualmente 2 relaciones, una de uno a muchos con asientos explicada anteriormente y la otra es con la tabla horarios, de tipo uno a muchos, ya que una sala puede aparecer en muchos horarios pero un horario no puede tener más de una sala.

La tabla horarios tiene 3 relaciones, con tickets, salas y películas, estas últimas 2 ya explicadas, la relación con tickets es de uno a muchos, ya que un horario puede aparecer en muchos tickets pero un ticket no puede tener más de un horario.

Por último, esta la tabla tickets, en esta se encuentran 3 relaciones, las cuales ya han sido explicadas, sin embargo es de las mas importantes ya que es donde se relación prácticamente todo el sistema de forma directa e indirecta.

Resultados

El funcionamiento del sistema será explicado en el video, con el fin de demostrar las operaciones más importantes que se pueden realizar para la administración y venta de tickets de un cine.

Discusión

Esta práctica me permitió poner a prueba no solo los conocimientos de programación de un sistema de información, sino también el proceso para poder organizar y plantear bien toda la organización del sistema de manera en que la estructura este correctamente definida y esto a su vez permita un correcto funcionamiento de todas las operaciones que se realicen dentro del sistema. La parte de la normalización y diagramas relacionales es la que mas trabajo me cuesta, sin embargo considero haber mejorado con el desarrollo de esta práctica.