

Actividad: Extracción de las clases de análisis a partir de un documento de información con un enfoque orientado a objetos.

Kalio Fernando O'Farril Villalpando
Metodologías, Desarrollo y Calidad en la Ingeniería de Software
Grupo 1001

Requisitos

Desarrollo de una aplicación aplicación móvil cuya finalidad sea apoyar la comunicación y centralización de diferentes tipos de eventos que se ofrecen en diferentes lugares con el propósito de dar a conocer al público en general y a su vez permitir la reservación a estos eventos de forma más efectiva.

Los eventos pueden ser de diferentes tipos: tours, obras de teatro, conciertos, festivales, actividades ya sea deportivas, recreativas, entre otros. Así los clientes podrán buscar y reservar eventos de su preferencia, en el lugar y fecha deseados; así como llevar un control sobre los eventos a los cuales ya reservaron. Por otra parte, los coordinadores serán los usuarios que podrán dar de alta sus eventos y administrarlos.

Ejemplos de los requerimientos funcionales serían los siguientes:

R001: El sistema deberá permitir registrarse con alguno de los 2 roles posibles, cliente o coordinador.

R002: El sistema deberá solicitar usuario y contraseña para poder registrar un evento o para reservar un evento.

R003: El administrador deberá permitir administrar los distintos módulos (usuarios, catálogos, entre otros).

R004: El Coordinador podrá crear nuevos eventos (actividades, tours o festivales).

R005: El Cliente podrá reservar uno o más eventos.

R006: Cualquier usuario podrá ver el calendario de actividades registradas (existentes) o reservadas.

R007: El cliente podrá cancelar una reservación.

R008: El Coordinador podrá cancelar o cambiar un registro.

R009: El Cliente podrá buscar eventos por tipo de evento (actividades, tours o festivales), por fecha, por ubicación, por categoría (familiar, adultos, niños) o por género (rock, bohemio, cómico).

Clases Extraídas y su Descripción

Clase	Descripción
Rol (enumeración)	Esta clase es una enumeración cuyos valores representan los distintos tipos de roles disponibles. Los roles existentes pueden ser agregados, modificados o eliminados con tan solo editar esta clase, lo que permite la rápida adaptación a cambios de manera segura.
Usuario	Contiene la información y operaciones básicas que todos los distintos tipos de usuario tendrán. Los atributos son privados al contener información sensible como nombre y contraseña, por lo que todos estos atributos tienen métodos getter y setter para asegurar un uso seguro.
Cliente	Extiende (hereda) la clase Usuario. Hace uso de la clase de ServiciosEventos para actualizar su atributo de la lista de eventos a los que esta registrado el usuario. Cuenta con las operaciones para reservar y cancelar reservaciones a eventos.
Coordinador	Extiende (hereda) la clase Usuario. Hace uso de la clase de ServiciosEventos para crear y editar objetos de la clase Evento. Cuenta con un atributo en forma de lista donde se almacenan los eventos que coordina.
Admin	Extiende (hereda) la clase Usuario. Hace uso de las clases de ServiciosEventos y ServiciosUsuario, lo que le permite acceder a todas las funcionalidades relacionadas con usuarios y eventos.
TipoEvento (enumeración)	Esta clase es una enumeración cuyos valores representan los distintos tipos de eventos disponibles (Actividad, Tour, Festival). Los tipos de eventos disponibles pueden ser agregados, modificados o eliminados con tan solo editar esta clase, lo que permite la rápida adaptación a cambios de manera segura.
CategoríaEvento (enumeración)	Esta clase es una enumeración cuyos valores representan los distintos tipos de categorías de eventos (Familiar, Niños, Adultos). Las distintas categorías de los eventos pueden ser agregadas, modificadas o eliminadas con tan solo editar esta clase, lo que permite la rápida adaptación a cambios de manera segura.
GéneroEvento (enumeración)	Esta clase es una enumeración cuyos valores representan los distintos tipos de género de eventos (Rock, Bohemio, Cómic). Los distintos géneros de

	los eventos pueden ser agregados, modificados o eliminados con tan solo editar esta clase, lo que permite la rápida adaptación a cambios de manera segura.
Evento	Cuenta con todas las propiedades necesarias de un evento: id que lo identifica, id del coordinador del evento, nombre del evento, fecha de inicio, cupo del evento (capacidad), lugares restantes (disponibilidad), ubicación del evento y si este evento esta activo o no (ya sea por cancelación del coordinador o porque ya fue efectuado el evento). También hace uso de las tres clases de tipo enumeración para el género, tipo y categoría del evento, al ser estas del tipo enumeración, si una es modificada la clase Evento puede reaccionar automáticamente al cambio sin verse afectada.
ServiciosEventos	Esta clase no cuenta con atributos. Tiene las funcionalidades necesarias para crear y editar eventos, así como las reservaciones a ellos. También incluye las operaciones para extraer todos los eventos con o sin filtros, ya sean de algún tipo de género, categoría y tipo o bien, extraer los eventos que algún usuario tenga asociado a su Id de usuario.
ServiciosUsuarios	Esta clase no cuenta con atributos. Cuenta con todas las funcionalidades relacionadas a los usuario, como son la creación, eliminiación, búsqueda o edición. También incluye la operación de login.

Atributos y Operaciones de Clases

Clase	Atributos	Operaciones
Rol (enumeración)	Admin, Coordinador, Cliente	N/A
Usuario	<ul style="list-style-type: none">- idUsuario: Tipo Long. Es un valor entero autoincremental que identifica cada usuario con un valor único.- nombre: Tipo String. Almacena el nombre que el usuario escoge para sí mismo.- contraseña: Tipo String. Almacena la contraseña que el usuario escogió para su cuenta.- rol: Tipo Rol. Almacena el rol del usuario de entre las diferentes posibilidades de la clase de enumeración Rol.	Esta clase es una enumeración cuyos valores representan los distintos tipos de categorías de eventos (Familiar, Niños, Adultos). Las distintas categorías de los eventos pueden ser agregadas, modificadas o eliminadas con tan solo editar esta clase, lo que permite la rápida adaptación a cambios de manera segura.
Cliente	<ul style="list-style-type: none">- eventos: Tipo Lista de valores Long. Almacena una lista de las ids de los eventos que el cliente ha reservado.- serviciosEvento: Esta clase es inyectada con una referencia a la clase ServiciosEvento para acceder a las	<ul style="list-style-type: none">- reservarEvento: Recibe la lista de eventos del cliente así como el id del evento a reservar. Regresa la nueva lista de eventos del cliente con el nuevo evento agregado. Hace uso del método “reservarEvento” de la clase ServiciosEvento.- cancelarReservación: Recibe la lista de eventos del cliente así como el id del evento del cual la reservación será cancelada. Regresa la nueva lista de eventos en donde se ha

	funcionalidades de los eventos.	removido el id de la reservación cancelada. Hace uso del método “cancelarReservación” de la clase ServiciosEvento.
Coordinador	<ul style="list-style-type: none"> - eventos: Tipo Lista de valores Long. Almacena una lista de las ids de los eventos que este usuario coordina. - serviciosEvento: Esta clase es inyectada con una referencia a la clase ServiciosEvento para acceder a las funcionalidades de los eventos. 	<ul style="list-style-type: none"> - crearEvento: Recibe la lista de eventos del coordinador así como un objeto de tipo Evento con todos los datos requeridos. Regresa la nueva lista de eventos en donde se ha agregado el evento creado. Hace uso del método “crearEvento” de la clase ServiciosEvento. - editarEvento: Recibe la lista de eventos del coordinador así como un objeto de tipo Evento con los datos modificados de un evento existente, mediante el atributo “activo” del evento se puede cancelar el evento. Regresa un valor booleano que indica si la edición del evento fue hecha exitosamente. Hace uso del método “editarEvento” de la clase ServiciosEvento.
Admin	<ul style="list-style-type: none"> - serviciosEvento: Esta clase es inyectada con una referencia a la clase ServiciosEvento para acceder a las funcionalidades de los eventos. - serviciosUsuario: Esta clase es inyectada con una referencia a la clase ServiciosUsuario para acceder a las funcionalidades de los usuarios. 	<ul style="list-style-type: none"> - crearUsuario: Implementación del método de la clase de ServiciosUsuario. - buscarUsuario: Implementación del método de la clase de ServiciosUsuario. - eliminarUsuario: Implementación del método de la clase de ServiciosUsuario. - editarUsuario: Implementación del método de la clase de ServiciosUsuario. - login: Implementación del método de la clase de ServiciosUsuario. - crearEvento: Implementación del método de la clase de ServiciosEvento. - editarEvento: Implementación del método de la clase de ServiciosEvento.

		<ul style="list-style-type: none"> - reservarEvento: Implementación del método de la clase de ServiciosEvento. - cancelarReservación: Implementación del método de la clase de ServiciosEvento. - verEventosUsuario: Implementación del método de la clase de ServiciosEvento. - verEventos: Implementación del método de la clase de ServiciosEvento. - verEventosFiltrado: Implementación del método de la clase de ServiciosEvento.
TipoEvento (enumeración)	Actividad, Tour, Festival	N/A
CategoríaEvento (enumeración)	Familiar, Adultos, Niños	N/A
GéneroEvento (enumeración)	Rock, Bohemio, Cómico	N/A
Evento	<ul style="list-style-type: none"> - idEvento: Tipo Long. Es un valor entero autoincremental que identifica cada evento con un valor único. - idCoordinador: Tipo Long. Almacena el id del coordinador del evento. - nombre: Tipo String. Almacena el nombre del evento. - fecha: Tipo DateTime. Almacena la fecha y hora de inicio del evento. -cupo: Tipo int. Almacena el cupo máximo de clientes al evento. -disponibilidad: Tipo int. Almacena el número de lugares restantes en el evento. -Ubicación: Tipo String. Contiene la dirección del evento. 	

	<p>-tipo: Tipo TipoEvento. Almacena el tipo de evento de entre las diferentes posibilidades de la clase de enumeración TipoEvento.</p> <p>-categoría: Tipo CategoríaEvento. Almacena la categoría de evento de entre las diferentes posibilidades de la clase de enumeración CategoríaEvento.</p> <p>-género: Tipo GéneroEvento. Almacena el género del evento de entre las diferentes posibilidades de la clase de enumeración GéneroEvento.</p> <p>-activo: Tipo Booleano. Indica si el evento sigue activo o no. Un evento inactivo puede ser que ya haya terminado o que haya sido cancelado.</p>	
ServiciosEvento	N/A	<p>- crearEvento: Recibe un objeto de tipo Evento. Regresa el evento creado. Identifica al evento con un id único y almacena los detalles del mismo.</p> <p>- editarEvento: Recibe un objeto de tipo Evento. Regresa el evento con sus valores actualizados. Sobrescribe los nuevos valores sobre el evento enviado usando como referencia el id del evento.</p> <p>- reservarEvento: Recibe el id de un usuario y el id de un evento. Regresa un booleano indicando si la reservación fue exitosa. Agrega a la lista de eventos del usuario el id del evento en cuestión y reduce por 1 el valor</p>

		<p>de disponibilidad en el evento asociado al id del evento proporcionado.</p> <ul style="list-style-type: none"> - cancelarReservación: Recibe el id de un usuario y el id de un evento. Regresa un booleano indicando si la cancelación de la reservación fue exitosa. Remueve de la lista de eventos del usuario el id del evento en cuestión y aumenta por 1 el valor de disponibilidad en el evento asociado al id del evento proporcionado. - verEventosUsuario: Recibe el id de un usuario. Regresa una lista de objetos tipo Evento usando los ids de eventos de el atributo eventos del usuario. El atributo de eventos del usuario son los registrados su es ciente o los coordinados si es un coordinador. - verEventos: No recibe parámetros. Regresa una lista de objetos tipo Evento con todos los eventos existentes. - verEventosFiltrado: Recibe tres parámetros de clases enumeración; TipoEvento, CategoríaEvento o GéneroEvento. Regresa una lista de objetos tipo Evento con los eventos existente activos que cumplan con las características de los filtros.
ServiciosUsuario	N/A	<ul style="list-style-type: none"> - crearUsuario: Recibe el nombre, contraseña y rol del usuario (limitado a los roles no admin). Regresa un booleano si la creación del usuario fue exitosa. El usuario es creado con un id autoincremental único. - buscarUsuario: Recibe un id de usuario. Regresa un objeto de tipo Usuario que coincide con el id proporcionado.

		<ul style="list-style-type: none">- eliminarUsuario: Recibe un id de usuario. Regresa un booleano indicando si el usuario asociado al id proporcionado fue eliminado apropiadamente.- editarUsuario: Recibe un objeto de tipo Usuario. Regresa los datos del usuario asociado al id del objeto usuario proporcionado sobrescribiendo el resto de sus valores.- login: Recibe el nombre de usuario y una contraseña. Regresa un booleano indicando si los datos proporcionados coinciden con los del usuario.
--	--	--

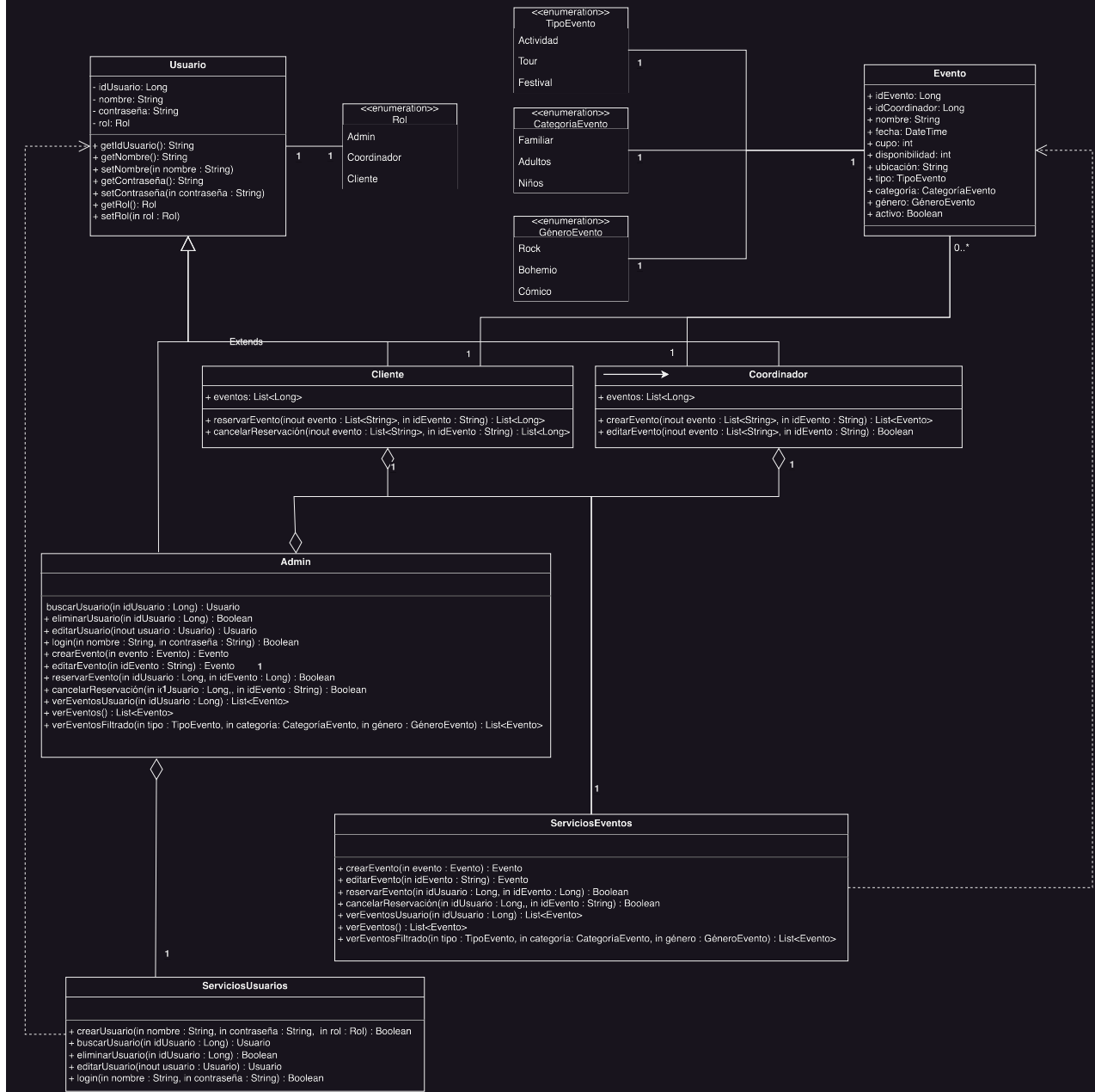
Relaciones entre Clases

Clase	Relaciones
Rol (enumeración)	- Una asociación simple con la clase Usuario de 1 a 1; cada objeto Usuario tendrá un objeto de Rol asignado. Es usada para mantener el control de roles en la clase de Usuario.
Usuario	- Es heredada a las clases Cliente, Coordinador y Admin. Contiene los atributos y métodos en común entre los distintos tipos de usuarios - La clase ServiciosUsuario tiene dependencia de esta clase debido a que hace uso de la manipulación de objetos de esta clase.
Cliente	- Hereda de la clase Usuario los atributos y métodos en común con el resto de Usuarios. - Tiene una asociación de uno a cero o más objetos de la clase Evento, indicando que un Cliente puede tener múltiples Eventos asociados. - Una asociación con la clase ServiciosEvento de 1 a 1 ya que hace uso de los métodos presentes en dicha clase.
Coordinador	- Hereda de la clase Usuario los atributos y métodos en común con el resto de Usuarios. - Tiene una asociación de uno a cero o más objetos de la clase Evento, indicando que un Coordinador puede tener múltiples Eventos asociados. - Una asociación con la clase ServiciosEvento de 1 a 1 ya que hace uso de los métodos presentes en dicha clase.
Admin	- Hereda de la clase Usuario los atributos y métodos en común con el resto de Usuarios. - Una asociación con la clase ServiciosEvento de 1 a 1 ya que hace uso de los métodos presentes en dicha clase. - Una asociación con la clase ServiciosUsuario de 1 a 1 ya que hace uso de los métodos presentes en dicha clase.
TipoEvento (enumeración)	- Una asociación simple con la clase Evento de 1 a 1; cada objeto Evento tendrá un objeto de TipoEvento asignado.
CategoríaEvento (enumeración)	- Una asociación simple con la clase Evento de 1 a 1; cada objeto Evento tendrá un objeto de CategoríaEvento asignado.
GéneroEvento (enumeración)	- Una asociación simple con la clase Evento de 1 a 1; cada objeto Evento tendrá un objeto de GéneroEvento asignado.

Evento	<ul style="list-style-type: none"> - Tiene una asociación de cero o más objetos a uno de las clases Cliente y Coordinador indicando que un Cliente o Coordinador puede tener múltiples Eventos asociados. - La clase ServiciosEvento tiene dependencia de esta clase.
ServiciosEvento	<ul style="list-style-type: none"> - Una asociación con la clase Cliente de 1 a 1. - Una asociación con la clase Coordinador de 1 a 1. - Una asociación con la clase Admin de 1 a 1. - Una dependencia con la clase Evento, sus métodos hacen uso de ella.
ServiciosUsuario	<ul style="list-style-type: none"> - Una asociación con la clase Admin de 1 a 1. - Una dependencia con la clase Usuario, sus métodos hacen uso de ella.

Diagrama de Clases

UML: Diagrama de Clases



Bibliografías

[1] Rumbaugh, J., Jacobson, I. y Booch, G. (2007). *El lenguaje unificado de modelado. Manual de referencia*. UML 2.0 (2^a ed.). Pearson Addison-Wesley.

[2] The Unified Modeling Language (10 Junio 2022)

https://www.uml-diagrams.org/#google_vignette

[3] UML Class Diagram Tutorial (10 Junio 2024)

<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/;WWWSESSIONID=F9E0E37B743E10DBBED5900AF6287C56.www1>