

# Diagrama Entidad-Relación

En este documento se recopilara la información sobre el diagrama de Entidad-Relación realizado para el proyecto. En este documento se especificaran e identificaran las principales entidades, los atributos de estas mismas y las diferentes relación entre las entidades. También se especificara el porque y el como se ha llegado al diagrama, dejando así claro nuestras decisiones.

El fin de este documento es tener una documentación clara y explicatoria del diagrama para tanto nuevos integrantes como los ya presentes en el proyecto.

Para dejar clara la estructura de este documento, primero se presentara la entidad, seguida de una explicación la cual justifica su función. Después se explicaran los atributos de dicha entidad y esto se hara para cada entidad. Al acabar, ya explicada cada entidad, se pasara a desarollar las diferentes relaciones entre entidades.

## Entidades y Atributos

### **Usuario:**

Usuario es una de las entidades mas importantes de nuestro diagrama, este tiene la función de representar el usuario que utilizara la plataforma y es aquella entidad la cual guardara los datos mas importantes de los usuarios. Es primordial que esta exista, ya que cualquier transacción parte de un usuario y como ahora veremos sus atributos son necesarios para la correcta gestión y comunicación con estos.

Como clave primaria tiene el atributo DNI el cual es único para cada usuario y funcionara como el identificador. También contaremos con otros atributos como nombre y apellidos del usuario, los cuales son obligatorios pero no únicos, ya que puede existir diferentes personas con mismo nombre y/o apellidos. Por último, dispone de los atributos email y SPAM, el cual también sera obligatorio y único ya que cada correo es diferente para cada persona. Aclaramos que SPAM se trata de un flag booleano. Estos atributos son necesarios para poder mandarles correos, ya sean recordatorios o anuncios de eventos similares.

Como ahora veremos, un usuario debe tener la capacidad de registrarse y de comprar un ticket, por tanto necesitaremos otras dos entidades las cuales estarán relacionadas con usuario.

### **Registrado:**

Registrado es una subentidad de usuario. Esta entidad representa a los usuarios que se registran formalmente en la plataforma, la cual proporcionan datos adicionales. Esta entidad es necesaria ya que los usuarios tienen la capacidad de registrarse, función que no es obligatoria.

Al principio tuvimos ciertas dudas de como representar a los usuarios registrados y de si crear un

usuario “empresa” para la compra de entradas de las empresas para sus empleados. Al final decidimos implementar los usuarios registrados como una subentidad de usuario. Esto se debe a que llegamos a la conclusión de que es simplemente un usuario, pero con datos adicionales y registrado. El tema de la compra de ticket como empresa se explicara más adelante, concretamente en la entidad “ticket”

Sus únicos atributos son teléfono y dirección postal los cuales simplemente guardan información adicional del usuario registrado.

### **Evento:**

Evento otra entidad indispensable en nuestro diagrama. La entidad tiene como función representar los eventos disponible en la plataforma. La entidad tiene gran importancia ya que tanto las entradas como los patrocinios giran en torno a los eventos. Los atributos de esta entidad permiten detallar la información necesaria para su gestión y publicación.

Como clave primaria tenemos a ID. Esta ID sera único para cada evento y estará vinculado al evento. No se repetirán IDs. Para esta entidad quizás podríamos haber utilizado una clave primaria compuesta formada por el nombre del evento y el lugar por ejemplo, pero por simplicidad preferimos crear un ID único. Esta decisión también la tomaremos en otras entidades. Los demás atributos de esta entidad simplemente añadirán información necesaria de dicho evento. Como atributos contamos con nombre, el cual sera simplemente el nombre del evento. La fecha y el lugar, que son necesarios para saber cuando se realizará el evento. Otro atributo que hemos añadido se llama tipo el cual es un identificador numérico que sirve para poder identificar los diferentes tipos de eventos como carreras, conciertos, rifas etc. Esta decisión se ha tomado por encima de crear subentidades debido a que no creemos que hayan suficiente información que almacenar para los diferentes tipos de eventos. No obstante, a nivel lógico de clases esto si que se tendrá en cuenta. Y por último tenemos la descripción el cual tendrá una breve información adicional relevante sobre el evento.

Un evento tiene unas etiquetas, ofrecen unas entradas y puede tener o no, unos patrocinadores. Por tanto necesitaremos una entidad para cada uno dichos de los aspectos.

### **Patrocinador:**

Como hemos mencionado, necesitaremos una entidad para poder representar a los patrocinadores. Los patrocinadores serán aquellos que patrocinan un evento. Es necesario ya que necesitamos establecer una relación entre los eventos y los patrocinadores.

Como atributos tenemos la clave primaria ID que, igual que los eventos, cada patrocinador tendrá una y será única, y como el otro atributo tendremos el nombre de dicho patrocinador.

### **Etiquetas:**

Esta será una entidad la cual solo estará relacionada con evento. Cada evento está asociada con una etiqueta. Esta etiqueta será necesaria para dar información adicional del evento (saber qué tipo de evento estamos acudiendo). También es útil ya que necesitaremos saber qué tipo de eventos acude

cada usuario para poder mostrarle a estos, evento similares. Sin la etiquetas, no podríamos aplicar esta funcionalidad.

La clave primaria sera ID que sera un único para cada etiqueta. Y como otro atributo, el nombre de cada etiqueta.

### **Entrada:**

Esta entidad es una entidad débil de evento, ya que no puede existir sin estar asociada a un evento. Representa los diferentes tipos de entradas a eventos. Es necesaria ya que no todos los eventos tiene el mismo tipo de entrada y por lo tanto es necesaria para poder tener controlado y asociado la entrada con el evento, y como veremos, con el ticket que dispondrá el usuario.

Cada evento singular puede tener distintos tipos de entradas. Por ejemplo un concierto puede tener entradas a distinto precio en función de la localidad. Distinguir estos tipos es el objetivo fundamental de esta entidad.

Cada diferente tipo de evento tendrá su tipo de entrada, la cual estará identificada por su clave primaria, ID, la cual asignara a cada tipo de entrada un identificador único. Dispondrá también de la clave foránea ID de evento. Esto se debe a que entrada es entidad débil de evento y necesitaremos saber a que evento pertenece la entrada. El atributo información sirve para concretar la información sobre el tipo de entrada. Esta información variara según el tipo de evento al que se acuda, siendo este diferente para un concierto, rifa, carrera etc. Por simplicidad hemos decidido no crear subentidades para los diferentes tipos de entrada, ya que nos parece mas simple y cómodo. Esto si que se tendrá en cuenta en el modelado lógico. Y como atributo final tenemos el precio, el cual guardara la información del precio de la entrada.

Cada tipo de entrada estará asociada a un ticket que es aquel que comprara y dispondrá el usuario. Por lo tanto, entrada tendrá que estar relacionada con ticket.

### **Ticket:**

Ticket es otra entidad indispensable en nuestro diagrama. Esta entidad tiene la función de llevar el control de las compras realizadas por los usuarios y sus atributos son necesarios para controlar y personalizar correctamente cada ticket.

Ticket es una entidad débil de usuario y de entrada ya que este no puede existir sin un usuario y sin el tipo de entrada. Realmente se trata de una relación muchos a muchos con atributos.

Esta entidad nace de la necesidad de necesitar algo que relacione los tipos de entrada con el usuario. A su vez esta entidad nos generaba un debate. ¿Las entradas son intransferibles?. Si la respuesta es sí, entonces, ¿qué pasa con las empresas que desean comprar muchas entradas para sus trabajadores? La solución a esto la resolvimos implementando un atributo DNI\_beneficiario el cual, es opcional y se explicara en el párrafo de los atributos.

Su clave primaria será id, la cual será única para cada ticket. Esta entidad contará con dos claves foráneas, tanto la clave principal de usuario que servirá para identificar el comprador (el cual será el que recibiría el certificado de donación en caso de proporcionar sus datos), que será el usuario y tanto la clave foránea de entrada, la cual identificara el tipo de ticket que estamos comprando. Como atributo opcional contamos con DNI\_beneficiario. Este atributo esta puesto como opcional

ya que contamos con que hayan empresas que compren entradas en masa ya sean para sortearlas, o regalárselas a sus empleados, y por lo tanto, serán transferibles y no nominales. En cambio habrán usuario las cuales solo compren entradas para ellos mismos u otras personas identificadas, las cuales sí serán nominales y estarán registradas con este atributo. Cabe la opción de que el atributo de la clave foránea del usuario como DNI\_beneficiario sea el mismo ya que aquel usuario que compre una entrada y se ponga como beneficiario a si mismo, tendrá el mismo DNI.

También hemos añadido un atributo pago extra para reflejar la cantidad que un usuario puede llegar a pagar adicionalmente al comprar una entrada, la cual se reflejaría después en el certificado de dicho usuario como una donación.

Por ultimo también hemos añadido un atributo información análogo al atributo información de la entidad entrada, la cual especificara información adicional en base al tipo de entrada. La justificación de este atributo es prácticamente la misma que en Entrada

Con esto finalizamos la explicación de cada entidad y sus atributos.

## Relaciones

Ahora pasaremos a explicar las relaciones entre las diferentes entidades.

**Usuario 1 – N Ticket:** Un usuario esta relacionado con ticket ya que el usuario es aquel que compra un ticket y el ticket es de un usuario. Un usuario no esta obligado a comprar un ticket, por tanto, es opcional pero tiene la capacidad de comprar mas de un ticket. En cambio, si existe un ticket, este debe haber sido comprado por un usuario, por tanto la relación es obligatorio y un ticket solo puede ser comprado por un usuario.

**Ticket N -1 Entrada:** Un ticket solo puede ser un tipo de entrada, y es obligatorio por lógica, de que debe existir un tipo de entrada para tu ticket. Por otro lado, pueden existir muchos tickets que sean de un mismo tipo de entrada pero no es obligatorio que exista, ya que puede ocurrir de que nadie compre un tipo específico de entradas.

**Evento 1 – N Entrada:** Un evento siempre dispondrá de al menos un tipo de entrada lo cual es obligatorio. Una entrada solo estará relacionada con un evento, y esta relación tiene que ser obligatoria.

**Evento N – N Patrocinador:** Un evento puede o no tener uno o varios patrocinadores. Un patrocinador puede o no estar patrocinando uno o varios eventos. Es una relación de muchos a muchos optionales

**Evento N – N Etiquetas:** Un evento puede tener una o más etiquetas, las cuales no son obligatorias. Una etiqueta puede estar relacionada con uno o mas eventos. A su vez, si existe una etiqueta esta tiene que tener un evento asociado por lo cual seria opcional la relación.