

Aprendizaje servicio (voluntariado)

Elementos de la aplicación.

Para una mejor explicación de los componentes que forman el proyecto, agregaremos estos en dos categorías. Las **actividades**, que hacen referencia a las ofertas que se muestran en la aplicación y los **usuarios**, personas que manipulan la aplicación.

Actividades.

Introducción a las actividades.

Las actividades son aquellas ofertas que se muestran en la aplicación.

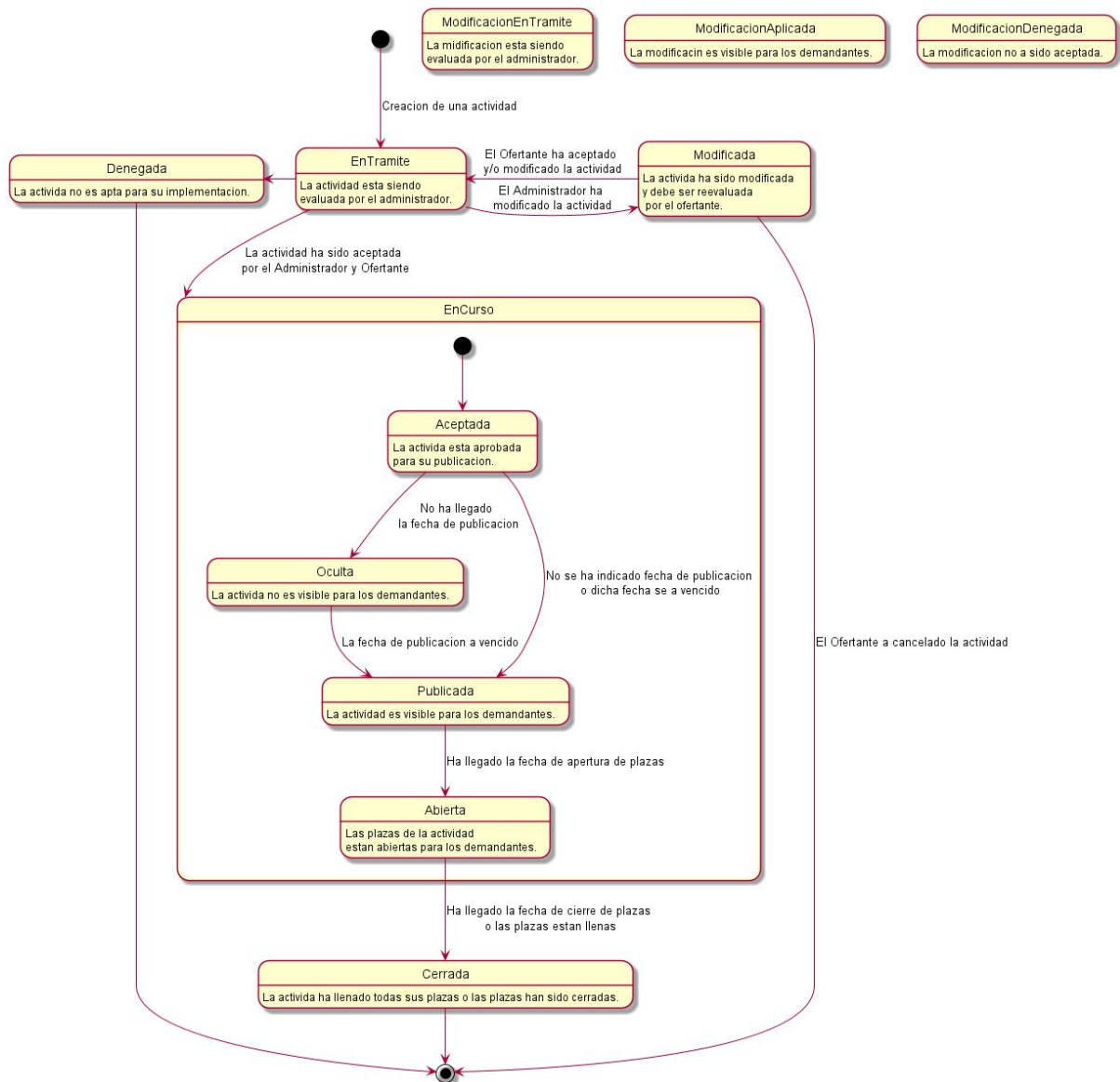
Cualidades de las actividades.

- Cada actividad se deberá clasificar en una o más etiquetas, dependiendo del ejercicio que se proponga.
- Las actividades contendrán un valor indicando la valoración de los usuarios. Dicho valor se almacenará en la base de datos como un decimal.
- Las actividades de estarán sujetas a un ciclo de vida dependiendo del estado en el que se encuentren.
- Las actividades se podrán actualizar, en cualquier estado pero dicha actualización debe de ir acompañada de una razón y debe de ser aprobada por el administrador de la actividad

Datos de las actividades.

1. **Estado:** Las actividades tienen **8 estados** posibles pero solo pueden pertenecer a un estado en cada instante de tiempo. Se refiere a la **clase estado**.
2. **Título:** Este atributo que almacena el título de la actividad, para poder mostrárselo al usuario.
3. **Descripción:** Para poder mostrar a los demandantes de que se trata la actividad, y destacar caracteres relevantes de la actividad.
4. **Imagen:** Almacena una posible imagen.
5. **Comentario:** Almacena los comentarios de los usuarios.
6. **Fecha de publicación:** Con este parámetro los Oferentes indican a los administradores cuando desean que se publique la actividad. (*Que pase del estado Oculta a Publicada*). En caso de que no se indique la actividad para de Aceptada a Publicidad.
7. **Fecha de apertura de plazas:** Indica la fecha límite en la cual las plazas están abiertas (*Traspaso del estado Publicado a Abierto*). Si no se especifica la fecha de apertura de plazas será la misma que la fecha de publicación.
8. **Fecha de cierre de las plazas:** Indica la fecha límite en la cual las plazas se cierran.
9. **Horas invertidas:** Almacena el número de horas estimadas que dura el ejercicio de actividad.
10. **Calificación:** Almacena la puntuación de calidad de los usuarios.
11. **Número de plazas:** Almacena el número de plazas disponible. Si no se rellena este parámetro indica que no hay un número máximo de plazas.
12. **Eventos:** Este parámetro referencia a la **clase eventos** para poder gestionar los calendarios de la actividad.
13. **Clasificación:** Referencia a la **clase etiquetas**, usada para clasificar la actividad en relación a la rama de conocimiento. Nota: Las etiquetas deben de ser concretadas por el cliente.
14. **Convalidación:** Referencia a la **clase Convalidación**, en esta se guardan los datos de relevancia para una convalidación por créditos. Este parámetro puede no ser introducido si la actividad no es evaluable.
15. **Solicitudes:** Son las solicitudes que es el usuario y se relaciona con la clase solicitudes para poder gestionarlas.

Ciclo de vida de la actividad.



Clases relacionadas con las actividades.

Clase Etiqueta.

Esta clase almacena las etiquetas que se podrán elegir en las **actividades** y los **demandantes**.

Clase comentario.

En esta clase se almacenan los comentarios de los usuarios con las actividades.

Datos de la clase comentario.

1. **Demandantes:** Guarda la relación uno a uno con la el demandante.
2. **Comentario:**

Clase Estado,

La relación de esta clase con la clase actividad nace de la necesidad de implementar un mensaje para notificar los cambios de estados de las clases.

Datos de Estado.

- **Estado Type:** Que almacena el estado actual.
- **Mensaje:** que almacena un string para referenciar al mensaje.

Clase Evento.

Esta clase nece como ayuda de la gestión de la actividad, en relación a las reuniones o el calendario de la actividad

Datos de Evento:

1. **Fecha:** Indica la fecha y la hora en el que se desarrolla el evento.
2. **Título:** Nombre del evento
3. **Descripción:** Almacena la descripción del evento.

Clase Convalidación.

En esta clase se almacena si es necesario los datos de la convalidación,

Datos de convalidación:

1. **Descripción:** Se describen los datos importantes de la convalidación como a qué asignatura convalidada o cuales son los requisitos para convalidar.
2. **Evaluación:** Cómo se va a evaluar o lo que se tendrá en cuenta a la hora de puntuar.

3. **Créditos máximos:** El número de créditos máximos de la actividad.

Clase solicitud.

La clase solicitud nace para suplir los requisitos funcionales de **evaluación del demandante, presentar solicitud, y gestionar solicitudes**.

Los datos que contiene esta clase son el estado de la solicitud, si el demandante no querer pertenecer más a la actividad o es expulsado este valor pasa a denegado, y la calificación obtenida en esta actividad.

Los estados posibles de la solicitud son.

1. **En Trámite:** cuando la solicitud está siendo revisada
2. **Aceptada:** cuando ocupas una plaza de la actividad.
3. **Denegada:** cuando renuncias a la plaza, eres expulsado o se niega tu participación.

Datos de solicitud:

1. **Estado:** Referencia a la clase estado.
2. **Nota:** Nota del alumno en esa actividad, si no tiene nota se marca como null.

Usuarios.

Descripción de la clase usuario.

La clase usuario es una clase abstracta que de la cual Administrador, Usuario y Ofertante, los datos que en ella aparece

Datos de usuario:

1. **DNI:** En caso de ser una entidad no es necesario rellenar este campo
2. **Nombre:**
3. **Apellido:** En caso de ser una entidad no es necesario rellenar este campo
4. **Contraseña:**
5. **Email:** Único y obligatorio
6. **Telefon:**
7. **Actividades:** Una lista de actividades, en cada subclase tiene un uso diferente, salvo en usuario.
8. **Imagen Perfil:**

Demandantes.

Son aquellos usuarios que demanda la participación de una actividad. Hereda de la clase usuario y añade más datos.

Datos de demandante:

1. **Fecha de nacimiento:**
2. **Interés:** donde el demandante podrá elegir las etiquetas disponibles de las actividades para recibir sugerencias de las actividades que le interesan.
3. **Datos de interés:** Dato que el usuario considera relevante para mostrar.
4. **Solicitudes:** Es una lista que almacena todas las solicitudes del demandante, el estado de la solicitud y si existe la posibilidad, recibida por la actividad.

Además la clase usuario implementa una relación con la clase Comentario que le permite al usuario introducir un comentario en la actividad, solo uno para generar una realimentación hacia los creadores y administradores de la actividad.

Ofertantes

Los ofertantes son aquellos usuarios que generan la actividad. Esta clase implementa la relación actividades como una forma de poder gestionar las actividades que han creado.

Datos de ofertante:

1. **Calificación:** almacena una calificación extraída de la calificación de sus actividades para mejorar los servicios que se muestran.
2. **Dirección:** La opción de poder añadir una dirección física como método de verificación de la entidad.
3. **Contacto:** En el caso de querer añadir más contactos.
4. **nif:** número de identificación fiscal, es una forma más de asegurarse de la veracidad de la entidad que ofrece el servicio.

Administrador.

Esta clase hace uso de la entidad actividades, relacionar al administrador con las actividades que requiere su gestión.

Esta clase no implementa atributos extra, ya que casi toda la cualidades son funciones.

Resumen.

Con este modelo de relaciones y datos abarcamos todos los requisitos.

Resumen de requisitos.

1. Registrarse como solicitante(demandante) o ofertante: Con la clase usuario cubrimos este requisito, generando una separación entre los demandantes y los ofertantes, ya que estos últimos requieren de información única como entidad como el nif.

Nota: Los administradores son usuarios con un rango de administrador y debe de ser otorgado por los gestores de la aplicación.

2. Crear, modificar, evaluar una actividad: Al tener una relación entre usuario y actividades y estar separados los rangos de demandante, oferente y administrador, podemos asignar los privilegios necesarios a cada uno para poder gestionar eficientemente las actividades.
3. Evaluar la calidad y comentar una actividad: Para ellos se crearon la clase comentario y el parámetro calificación en la clase actividad. de este modo podemos generar una realimentación de la experiencia de los usuarios.
4. Gestión de los participantes: Con la relación de solicitud podemos tanto evaluar al participantes como evaluar su solicitud además de que el participante pueda retirar su participación.
5. Modificar los datos de usuario: Los usuarios tendrán acceso a sus datos.

GitHub

<https://github.com/CarlosRA97/AprendizajeServicioVoluntarioServidor/tree/6e002dcd3ea0d80df60c08edb21e71a1527d46b0>

Modelo ER

En la siguiente página.

