

Entrega 2: Diagramas UML

PADSOF

Grupo 5

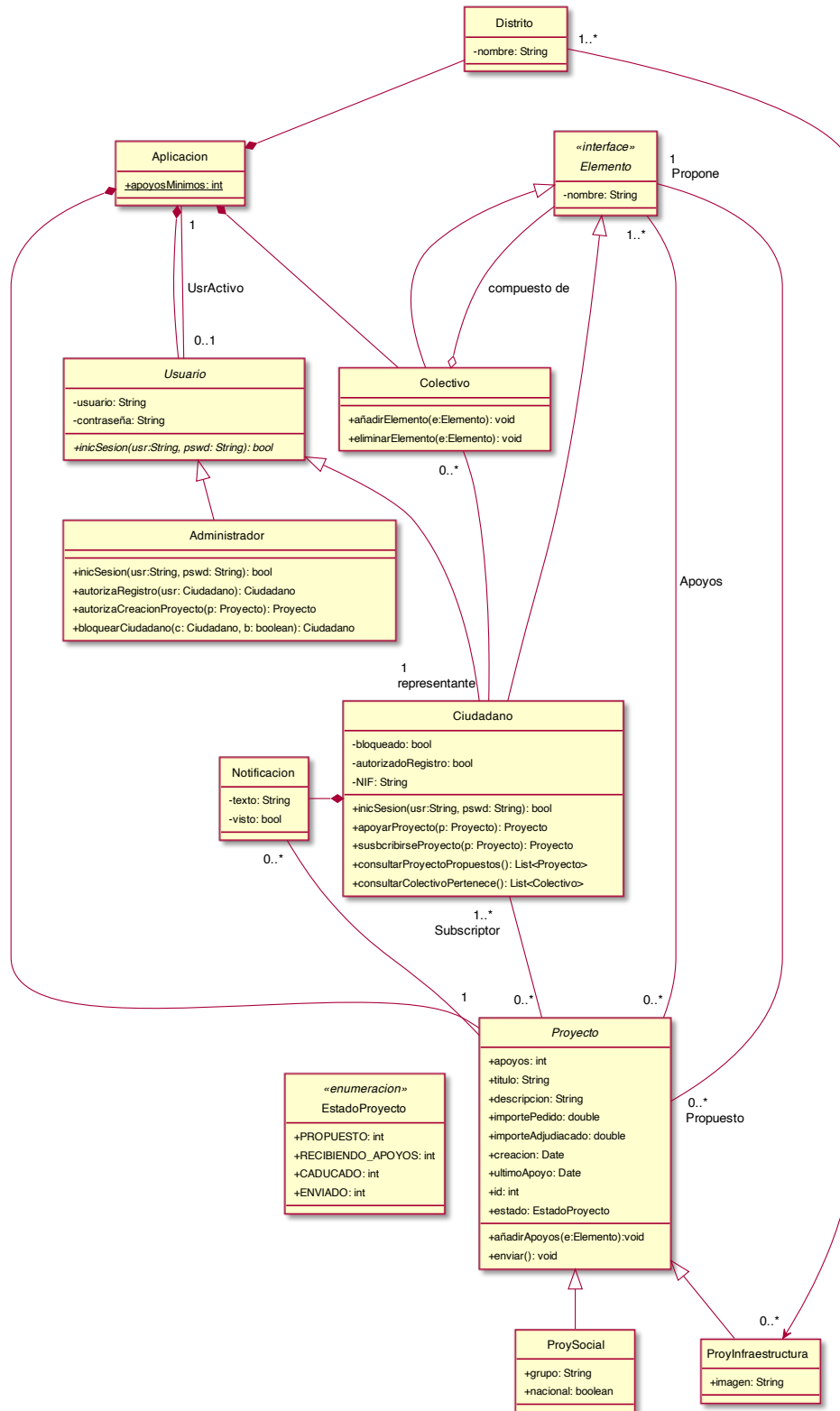
David del Val,
Junco de las Heras,
Jorge Fernández

Doble grado Informática-Matemáticas

Grupo 2201

UAM, Marzo 2020

1. Diagrama de clases



El diagrama de clases es el primer diagrama que realizamos en esta fase del diseño ya que la definición de las clases es necesaria para el resto de los diagramas

El elemento central de nuestro diseño es la clase Aplicación. Esta almacenará todas las instancias de las clases principales (**Usuario**, **Colectivo** y **Proyecto**). Respecto a la clase abstracta *Usuario*, cabe destacar que la utilizamos para encapsular las clases **Ciudadano** y **Administrador** ya que estas comparten parte de su funcionalidad y atributos.

En segundo lugar, queremos comentar la relación entre los colectivos y ciudadanos. Para el representate, hemos incluido una relación de 1 a muchos mientras que los demás integrantes del colectivo se relacionan con este a través de la clase *Elemento*. Esta nos permite expresar que tanto los ciudadanos como los colectivos (sub-colectivos) pueden formar parte de un colectivo. Entre **Colectivo** y *Elemento* hemos establecido una relación de agregación ya que podemos considerar que los elementos están contenidos en el colectivo.

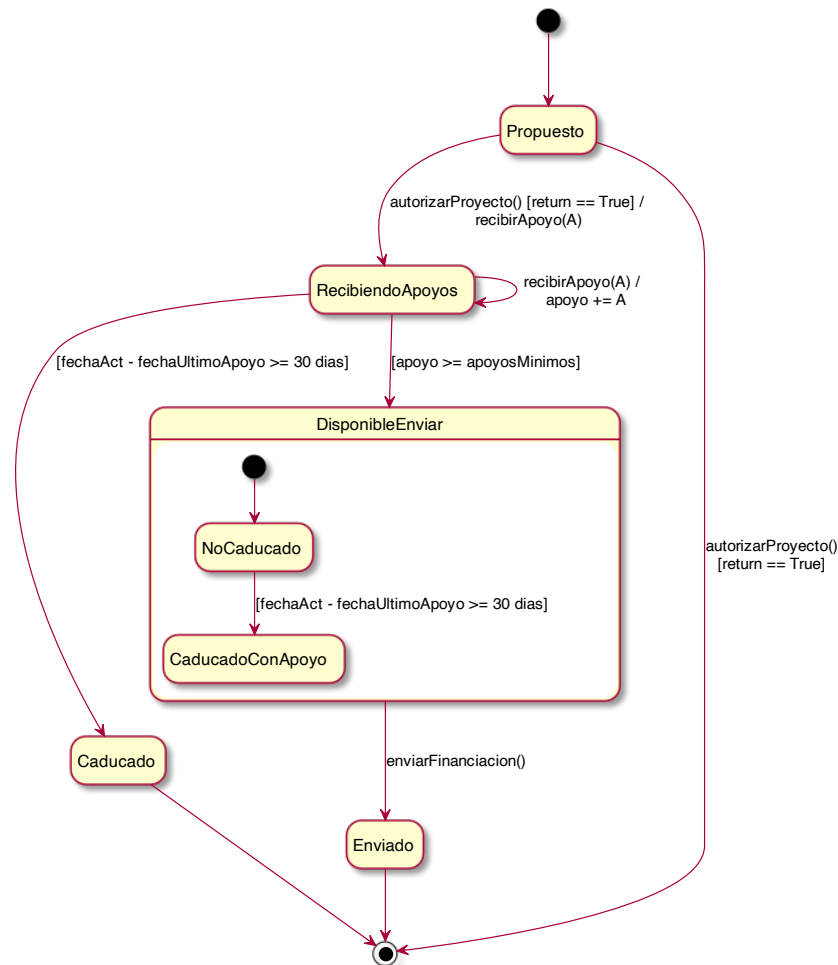
Por otra parte, la interfaz *Elemento* se relaciona con **Proyecto** mediante dos tipos de asociaciones. En primer lugar, hemos de representar que un elemento (colectivo o ciudadano) puede apoyar a una serie de proyectos. Para ello, hemos utilizado una asociación de muchos a muchos. En segundo lugar, un colectivo o ciudadano puede proponer varios proyectos. Como un proyecto solo puede estar propuesto por una entidad, esta relación es uno a muchos.

Para representar los proyectos decidimos implementar una clase abstracta *Proyecto* que contenga los atributos y la funcionalidad común a todos los proyectos mientras que las particularidades de cada tipo se implementan en la clase correspondiente (**ProySocial** o **ProyInfraestructura**).

Ya que los ciudadanos pueden subscribirse a proyectos para que se les informe de los cambios en el estado de estos, hemos creado una clase **Notificación**. Cada ciudadano tendrá sus propias notificaciones que se crearán cuando se produzcan cambios en proyectos que tengan al ciudadano en la lista de subscriptores

Por último, queremos señalar que los distritos se han implementado con una clase en vez de una enumeración para permitir añadir nuevos distritos durante el ciclo de vida de la aplicación. En la fase de captura de requisitos, decidimos que al crear un proyecto de infraestructura, los distritos a los que afecta se elegirían de una lista de todos los distritos de la ciudad. Esta lista la hemos incluido en la clase aplicación con la relación de contenido.

2. Diagrama de transición de estados de Proyecto

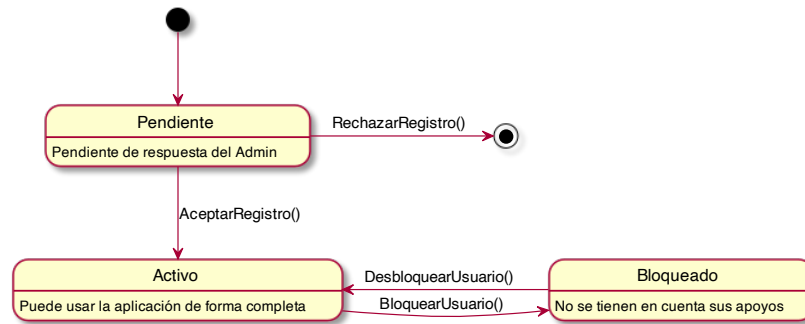


Nuestro primer diagrama de transición de estados describe la clase **Proyecto**. Elegimos esta clase ya que sus objetos son probablemente los que atraviesan más estados.

Al crear el proyecto este empieza en el estado de **propuesto**. En este estado, espera la acción del administrador, que tiene dos opciones, o deniega el proyecto (caso en el que el diagrama acaba) o lo acepta, pasando al estado de **‘RecibiendoApoyos’**. En este permanece hasta que caduca (han pasado al menos 30 días desde que recibió el último apoyo) o llega al número mínimo de apoyos que necesita (**apoyosMinimos**) para poder ser enviado.

En este último caso entra en el estado jerárquico **‘DisponibleEnviar’**. Este es el estado desde el que un proyecto puede ser enviado. Dentro de él, el objeto empieza como **no caducado**, pero pasa a **‘CaducadoConApoyo’** si pasan 30 días desde el último apoyo. Está caducado o no, si ya ha recibido el número mínimo de apoyos, este proyecto puede ser enviado al sistema externo.

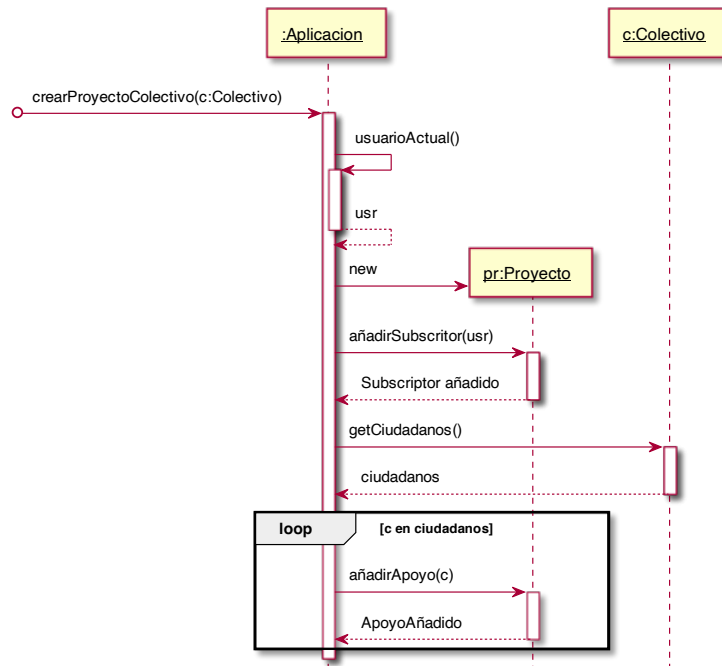
3. Diagrama de transición de estados de Ciudadano



En el caso de la clase **Ciudadano**, sus instancias se crean cuando un individuo se registra en la aplicación. Al registrarse, se le solicitan los datos necesarios para crear un nuevo usuario y este pasará al estado de 'Pendiente'. Permanecerá en este estado hasta que el administrador se pronuncie sobre el registro. Si el administrador rechaza el registro, el objeto pasará a un estado final de eliminado ya que esa instancia de **Ciudadano** ya no es necesaria.

Por otra parte, si el administrador decide aprobar el registro, el usuario pasa a estar activo. Este es el estado en el que se prevé que el objeto esté durante la mayor parte del tiempo. Desde este estado, el administrador puede bloquear al ciudadano, lo cual lo pasaría al estado de 'Bloqueado', donde sus apoyos ya no se tienen en cuenta. Del mismo modo, el administrador puede devolver al usuario al estado de activo.

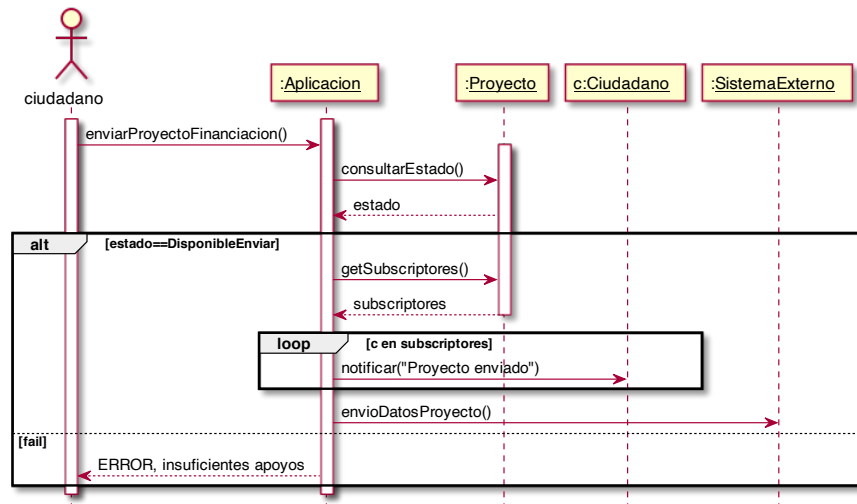
4. Diagrama de secuencia de crearProyectoColectivo



La situación que vamos a estudiar es la creación de un nuevo proyecto por parte de un colectivo. Esta comienza con una petición a la aplicación en la que se indica el colectivo que está solicitando la creación del nuevo proyecto.

En primer lugar, la aplicación solicita el usuario actual (que es el representante del colectivo) ya que este quedará suscrito al proyecto al haber sido el creador. Tras subscribir al usuario, la aplicación obtiene la lista de ciudadanos que pertenecen al colectivo `c` dado. Todos ellos deben de apoyar al proyecto ya que este está siendo creado en nombre del colectivo. Por tanto, la aplicación recorre la lista de ciudadanos y añade sus apoyos al Proyecto.

5. Diagrama de secuencia de enviarProyecto



En este último diagrama vamos a analizar el escenario en el que un ciudadano envía un proyecto al sistema externo de financiación. En primer lugar, el ciudadano envía una petición de enviar proyecto a financiación a la aplicación, la cual consulta el estado del proyecto.

Si su estado no es el de 'DisponibleEnviar', la aplicación devuelve un mensaje de error al usuario. Si, por el contrario, el proyecto puede ser enviado, se solicita la lista de los subscriptores. Esta está compuesta por la persona que propuso el proyecto (o el representante del colectivo en su defecto) y los ciudadanos que se subscribieron individualmente. A cada uno de los ciudadanos en esta lista se les envía una notificación de que el proyecto ha sido enviado y, finalmente, los datos del proyecto son entregados al sistema externo.