

Caso Práctico

¡Apóyanos! Financiación colectiva de proyectos

Requisitos

Se desea implementar una aplicación de escritorio inspirada en las numerosas plataformas de micro-mecenazgo (*crowdfunding*) que han surgido en los últimos años. Estas plataformas ofrecen a sus usuarios la posibilidad de proponer proyectos y que estos puedan ser apoyados económicamente para conseguir la financiación necesaria para su puesta en marcha (<http://es.wikipedia.org/wiki/Micromecenazgo>). A continuación se presenta una especificación de la funcionalidad que debe soportar la aplicación *¡Apóyanos!* propuesta como caso práctico de la asignatura y que se basa en la plataforma *Lanzanos* (<http://www.lanzanos.com/>).

Esquema general de funcionamiento

Cualquier **usuario** registrado puede crear un **proyecto** indicando, entre otras, las siguientes informaciones: una descripción, la financiación necesaria, el plazo para conseguir esta financiación y las recompensas. Un proyecto puede establecer varias **recompensas**, cada una de las cuales es definida por una cantidad de dinero y una descripción de qué obtendría el usuario si apoya el proyecto con esa cantidad. A partir de su creación los proyectos entran en la **fase de votación** en la que pueden ser votados por los usuarios (mecenazgos) y si consiguen un número determinado de **votos** (100 en el caso de lanzanos.com) entonces pasan a la **fase de financiación**. En esta fase los usuarios pueden **financiar (apoyar)** un proyecto eligiendo una recompensa. Una vez acabado el plazo establecido para la financiación, los usuarios reciben una notificación del sistema que les indica si el proyecto ha conseguido o no la financiación necesaria. Si se ha alcanzado dicha financiación se carga a los mecenazgos el pago de la cantidad con la que apoyaron. La realización de los pagos, que pueden realizarse a través de Paypal o tarjeta bancaria no será considerada en la aplicación. La entrega de las recompensas queda fuera del ámbito de este tipo de plataformas y no debe ser considerada en el desarrollo de *¡Apóyanos!*

Por ejemplo, se podría crear un proyecto para solicitar ayuda económica para escribir un libro y podría tener recompensas tales como: recibir un ejemplar del libro firmado y aparecer en los agradecimientos si se apoya con 100 euros, recibir una lámina con la portada del libro si se apoya con 50 euros y aparecer un agradecimiento en la web del autor si se apoya con 5 euros. En www.lanzanos.com se pueden ver ejemplos de proyectos y recompensas.

Detallamos a continuación cada una de las funciones que se deben implementar.

Entrada al sistema

Para utilizar los servicios del sistema, los usuarios deben realizar un *login* con su nombre y contraseña. Un usuario deberá registrarse tanto si desea lanzar un proyecto como ser mecenazgos. Una ventana de *login* permitirá acceder a la ventana principal de la aplicación o bien a la ventana de registro si el usuario que accede no está ya registrado. En los seminarios de prácticas de Swing se abordará la implementación de esta ventana de *Login*.

Registro

En el registro se debe proporcionar la siguiente información: datos personales (nombre y apellidos, dni), nombre de usuario, contraseña (se introduce dos veces y se comprueba que es la misma) y email, y además se deberán aceptar los términos de uso. En los seminarios de prácticas de Swing se abordará la implementación de esta ventana de *Registro*.

Crear Proyecto

La información necesaria para crear un proyecto es la siguiente: título, categoría, descripción, cantidad de dinero para iniciar el proyecto, plazo para conseguir la financiación (máximo 3 meses), una o más recompensas, modo de pago: paypal o abono en tarjeta bancaria. Cada recompensa viene dada por una cantidad con la que se puede apoyar, un título y una descripción que detalla la recompensa a obtener por el mecenas. Una recompensa puede tener un valor máximo de apoyos permitidos, por ejemplo se puede establecer que una recompensa de regalo de un ejemplar de libro podrá tener como máximo 10 apoyos. Las categorías podrían ser: Música, Libros, Cine, Social, Software y Deportes.

Al crear un proyecto esté entra en la fase de votación y no pasa a la fase de financiación hasta conseguir 50 votos.

Votar un proyecto

Para votar un proyecto un usuario selecciona primero una categoría (o puede seleccionar ver todos los proyectos en fase de votación) y luego un proyecto en fase de votación. Cuando un usuario vota, el sistema actualiza el número de votos y comprueba si el proyecto ha alcanzado 50 votos para pasar a la fase de financiación (Nota: Se probará esta funcionalidad con un número menor de votos).

Apoyar un proyecto

Para apoyar un proyecto un usuario (mecenas) selecciona primero una categoría (o puede seleccionar ver todos los proyectos en fase de financiación) y luego un proyecto en fase de financiación. Entonces el mecenas elegirá una recompensa para apoyar con una determinada cantidad. Cuando un usuario apoya un proyecto, el sistema actualiza la cantidad financiada y comprueba si se ha alcanzado la cantidad establecida en su definición en cuyo caso es cerrado por el sistema y se envía una notificación a los mecenas que han apoyado.

Finalización de la fase de financiación

El sistema comprobará cada 24 horas (se podrá simular haciendo click en un botón) qué proyectos han expirado su plazo para obtener la financiación, el cual es establecido por el usuario que crea el proyecto. Cuando expire el plazo se les enviará una notificación a los usuarios que han financiado el proyecto.

Un proyecto puede recibir financiación desde varias plataformas crowdfunding. Por ello, antes de comprobar la financiación actual de un proyecto se actualiza su valor cargando y sumándole la financiación recibida en otras plataformas para lo que se creará un componente según se indica más adelante.

Cálculo de comisión

La plataforma ¡Apóyanos! cobra una comisión a los usuarios que consiguen la financiación para su proyecto. La comisión es del 4% excepto para los proyectos sociales que es del 2% y los proyectos de cine que solicitan más de 6000 euros que es del 5%. Otras formas de calcular la comisión pueden ser establecidas en el futuro.

Servicio interno de mensajería

El sistema ofrece un servicio de mensajería interno que permite a los usuarios realizar preguntas al creador de un proyecto. Los usuarios podrán consultar las preguntas realizadas y las respuestas obtenidas en una ventana que es la misma desde la que se pueden enviar preguntas. La implementación de este servicio es opcional aunque debe ser considerado en la creación del modelo del dominio.

Arquitectura de la aplicación

Se organizará la aplicación de acuerdo a un modelo de tres capas (Presentación, Lógica de Negocio y Almacenamiento) que será descrito en un seminario de prácticas y se entregará una documentación con un ejemplo de caso práctico (Terminal de Punto de Venta, TPV).

Se proporcionará a los alumnos un servicio de persistencia que usarán para almacenar objetos. Se entregará documentación sobre el funcionamiento y uso de este servicio de persistencia y se explicará en un seminario de prácticas.

Se utilizará un patrón DAO para independizar a la aplicación de un servicio de persistencia concreto de acuerdo a las explicaciones proporcionadas en las clases de teoría y prácticas.

Cuando se entregue a los alumnos el diagrama de clases de la aplicación se indicará cuáles son las clases persistentes.

Interfaces de usuario

A continuación se detallan las ventanas que formarán parte de la aplicación con el objetivo de facilitar el desarrollo de la aplicación. No obstante, los alumnos tienen libertad para optar por otros diseños.

- **Ventana Login.** Se entregará a los alumnos y se discutirá en un seminario de prácticas. Incluirá los botones “Entrar” y “Registro usuario”.
- **Ventana Registro.** Se entregará a los alumnos y se discutirá en un seminario de prácticas. Incluirá los campos de entrada de los datos de un usuario.
- **Ventana Principal.** Desde ella el usuario podrá visualizar un listado de proyectos en forma de tabla. Para un proyecto en fase de votación se muestra el título, 120 caracteres de la descripción, los días restantes y el número de votos conseguido. Los proyectos se listarán en orden decreciente de votos conseguidos. Para un proyecto en fase de financiación se muestra el título, 120 caracteres de la descripción, días restantes, dinero recaudado y porcentaje recaudado. En cualquier caso, cada proyecto incluirá un botón para mostrar una ventana de información sobre el proyecto que se describe abajo. La ventana principal también incluirá un botón para abrir la ventana de Creación de un Proyecto.
- **Ventana Creación de un Proyecto.** Incluirá dos pestañas. La primera mostrará el panel de introducción de datos básicos de un proyecto: título, descripción, fecha de finalización, importe y categoría. La segunda muestra un panel con dos áreas. En una aparecen los campos para la entrada de los datos de una recompensa: título, cantidad y descripción, un botón para insertar la recompensa y otro para limpiar los campos. En la otra área se mostrará una tabla con todas las recompensas introducidas y botones para editar la tabla y borrar una entrada. El contenido de la tabla deberá ser actualizado al insertar un nuevo elemento, editar uno existente o borrar una entrada. Este segundo panel también incluirá en una tercera área los botones para registrar y cancelar el proyecto.
- **Ventana Información de un proyecto en votación.** Se accede a ella desde la ventana principal como se indicó arriba. Mostrará la información de un proyecto organizada en dos áreas. A la izquierda los datos del proyecto (título, descripción, categoría, importe, fecha de finalización, días que restan para finalizar la votación, número de votos conseguidos) y a la derecha una columna con todas las recompensas. Debajo de estas dos áreas aparecerá un botón para votar. Cuando se haga click sobre este botón se mostrará una ventana de dialogo de confirmación.

- **Ventana Información de un proyecto en financiación.** Se accede a ella desde la ventana principal como se indicó arriba. Mostrará la información de un proyecto organizada en dos áreas. A la izquierda los datos del proyecto (título, descripción, categoría, importe, fecha de finalización, días que restan para finalizar la financiación, financiación recibida, porcentaje del total) y a la derecha una columna con todas las recompensas. Cada recompensa incluirá un botón para apoyar. Cuando se haga click sobre uno de estos botones se mostrará una ventana de confirmación.
- **Ventana Notificaciones.** Mostrará una tabla con las notificaciones recibidas por un mecenas sobre los proyectos que ha apoyado. Se mostrará el título del proyecto, el nombre del creador y un texto del tipo “Proyecto supera/no supera la fase de financiación con % financiación de un total de X euros”.
- **Ventana de Preguntas/Respuestas.** Mostrará una ventana con dos pestañas. La primera mostrará un panel con la lista de títulos de las preguntas que ha realizado un usuario y al seleccionar un ítem de esta lista se mostrará el título, y texto de la pregunta y la respuesta. También incluirá un botón para insertar una nueva pregunta que aprovechará los elementos que muestran la información de la pregunta. La segunda pestaña mostrará un panel con un listado de preguntas contestadas o pendientes de contestar (el título aparecerá en un color diferente si está pendiente de contestación). Al seleccionar un ítem se mostrará la información y si la pregunta no está contestada, el usuario podrá completar la respuesta.
- **Ventana apoyos de un usuario.** Mostrará una ventana con un listado de los apoyos de un usuario a recompensas de proyectos. Se mostrará una tabla con el título del proyecto y de la recompensa y la cantidad financiada. Un botón permitirá abrir la ventana de información sobre un proyecto.

Componentes Java Beans

Los alumnos deberán **crear un componente propio** y **usar componentes de terceras partes**.

- Para **introducir la fecha** de terminación de un proyecto se usará algunos de los componentes Javabeans disponibles por la red que ofrecen esta funcionalidad como por ejemplo JCalendar (<http://toedter.com/jcalendar/>).
- A partir de un cargador de datos XML de financiación que se proporcionará a los alumnos, se deberá construir un componente FinanciacionProyectos que notificará a la aplicación qué **financiación externa** han recibido los proyectos. Los detalles sobre la creación de este componente se proporcionarán en un documento aparte.
- El componente Luz que se construirá en las clases prácticas será utilizado para **activar la comprobación de las finalizaciones** de las fases de votación y financiación de los proyectos.

Grupos de prácticas

Los alumnos deberán formar **grupos de dos** que deberán ser del mismo grupo de teoría salvo casos excepcionales. Se recomienda organizar el trabajo de forma que los dos miembros de un grupo puedan trabajar en paralelo, por ejemplo un reparto de las clases de la interfaz de usuario o de las clases del negocio o que se repartan la implementación del controlador o del acceso al servicio de persistencia. Los dos alumnos deben tener un conocimiento de todos los aspectos de la práctica y deberían haber programado en cada capa de la arquitectura.

Gestión del proyecto

Cada grupo se deberá registrar en **Assembla** (repositorio **Subversion**) y creará un proyecto **Maven**. Se dedicará un seminario de prácticas a explicar Maven y otro a Subversión. En el proyecto Eclipse debe existir una carpeta con las librerías usadas (configurar el *build path* con archivos locales en vez de externos) con el fin de facilitar su ejecución por el profesor.

No es necesaria una documentación usando javadoc, pero sí deben incluirse comentarios en el código, como mínimo para explicar lo que hace cada método cuyo comportamiento no sea obvio. También deberían comentarse los atributos y variables que no tengan un significado claro.

Patrones de Diseño

Los conocimientos sobre patrones se valoran principalmente en el examen de teoría de la asignatura. No obstante, la realización de esta práctica conlleva el uso de algunos patrones aplicados directamente por el alumno o indirectamente al utilizar alguna construcción o funcionalidad de Java o de los servicios proporcionados por los profesores. El alumno deberá indicar en la documentación cuáles son estos patrones y comentar brevemente su uso.

Como ejercicios adicionales a ser discutido en la documentación, el alumno debe (i) analizar el uso de patrones para que la aplicación fuese independiente del sistema de pagos y (ii) para que se manejaran recompensas predefinidas y cada tipo incluyese la información a aportar por el usuario, por ejemplo la dirección si debe recibir un regalo, la talla si va a recibir una camiseta, etc.

Hitos

La práctica se organizará en las siguientes fases:

1. Realización del diagrama de clases inicial que será entregado al profesor del grupo. La fecha de entrega será el **viernes 24 de octubre**.
2. En un seminario se proporcionará el diagrama de clases a seguir.
3. Diseño de la interfaz de usuario a partir de los conocimientos adquiridos en los seminarios de Swing. El diseño de ventanas expuesto arriba pretende facilitar el desarrollo de la aplicación a los alumnos.
4. El grupo podrá aplicar una estrategia de desarrollo iterativo completando de forma incremental las diferentes capas.
5. Entrevista con el profesor para seguimiento del trabajo práctico (**primera y segunda semana de diciembre**). Para esta entrevista el grupo debería haber abordado ya parte del desarrollo de cada capa.
6. Uso y desarrollo de componentes según se explicó arriba.
7. Se implementarán test unitarios para, al menos, una clase del modelo.

Fecha de entrega: 12 de enero

Documentación

Constará de las siguientes partes:

1. Índice de contenidos
2. Diagrama de clases del dominio.
3. Un diagrama de colaboración o secuencia para la operación de apoyar un proyecto.
4. Una breve explicación de la arquitectura de la aplicación y decisiones de diseño que se consideren de interés para la comprensión del trabajo.
5. Explicación de los patrones de diseño utilizados.
6. Explicación sobre los componentes utilizados.
7. Tests unitarios implementados
8. Un pequeño manual de usuario que explique cómo usar la aplicación
9. Observaciones finales (deben incluir una estimación del tiempo dedicado)

Se creará una tarea en el Aula Virtual para la entrega de la documentación como archivo pdf y la aplicación como con un fichero zip con el proyecto Eclipse.

Evaluación

Cada parte se valorará con el siguiente porcentaje:

- Modelo del dominio de clases (10%)
- Diseño de la interfaz (20%)
- Uso de componentes (15%),
- Ejercicios de patrones (5%),
- Código (separación en tres capas, legibilidad, uso de principios básicos de programación OO, implementación de patrones de diseño, uso de swing, comentarios) (35%),
- Uso de JUnit (5%),
- Documentación entregada (10%).
- Es obligatorio el uso de Maven y Subversión.