

## Orientación a Objetos II – 2021

### Trabajo Práctico 1

El trabajo práctico nº 1 es el diseño e implementación de una aplicación Web interactiva. Este TP 1 es de entrega obligatoria y debe hacerse en grupos de dos personas. Su enunciado está dividido en 3 partes:

**Parte I:** Diseño, implementación y testing del dominio de la aplicación.

**Parte II:** Construcción de aplicación Web interactiva básica.

**Parte III:** Construcción de la aplicación Web interactiva completa.

Fecha límite de entrega del TP 1 (Partes I, II y III): **Viernes 16 de Abril de 2021 23:59hs**

---

#### Parte I: Diseño, implementación y testing del dominio de la aplicación

##### *Descripción del dominio de la aplicación*

CuOOra es una red social de preguntas y respuestas. Los usuarios pueden crear ambas cosas (es decir, crear nuevas preguntas o crear respuestas para preguntas de otros usuarios), como así también votarlas mediante mecanismo de like/dislike. Las preguntas deben pertenecer al menos a un tópico.

A través de CuOOra debe proveer la siguiente funcionalidad:

- **Crear un nuevo usuario:** indicando su nombre de usuario, contraseña. Debe registrarse fecha y hora de creación de la cuenta.
- **Agregar tópicos:** indicando su nombre y descripción.
- **Agregar pregunta:** indicando los tópicos a las que se relaciona (al menos uno), un título, una descripción de la pregunta. Debe registrarse el usuario que la realiza y hora y fecha de creación.
- **Agregar una respuesta:** se indica pregunta que se responde, el texto de la respuesta, hora y fecha de creación. Debe registrarse el usuario que la realiza.
- **Agregar un usuario como follower de otro.**
- **Registrar Like/Dislikes a preguntas y respuestas:** además se indica el usuario que realiza esa acción y debe saberse el momento en el que realizó la acción. Un usuario puede realizar un solo like o un solo dislike para una pregunta o respuesta. Si un usuario dio like a una pregunta o respuesta, y luego da un dislike, lo que debe suceder es que se elimina el like anterior, y se agrega el dislike.
- **Registrar un tópico de interés para un usuario determinado.**
- **Obtener todas las preguntas de un tópico:** se indica un tópico y debe retornar todas las preguntas que se realizaron relacionadas a ese tópico.
- **Obtener preguntas relevantes para un usuario determinado:** Se considera relevante una pregunta cuando ésta pertenece a un tópico de interés para el usuario en cuestión o bien cuando fue realizada por alguno de los usuarios que sigue en la red social.

- **Obtener las respuestas a una pregunta:** se indica la pregunta y se retornan todas las respuestas ordenadas de mayor a menor por cantidad mayor puntaje, el que está dado por la diferencia de likes y dislikes.
- **Borrar una pregunta:** elimina del sistema la pregunta y todas sus respuestas.
- **Conocer las preguntas que realizó un usuario.**
- **Conocer las respuestas de un usuario.**
- **Calcular el puntaje de un usuario:** se indica un usuario, y se retorna un puntaje que se calcula en función de su actividad en la red social. Por cada pregunta realizada, suma 20 puntos, y por cada respuesta a una pregunta que realizó otro usuario, suma 50 puntos. Además, se suma puntaje determinado por la diferencia entre likes y dislikes de todas sus participaciones, que de tener mas dislikes que likes, puede ser negativo.

**Tareas:**

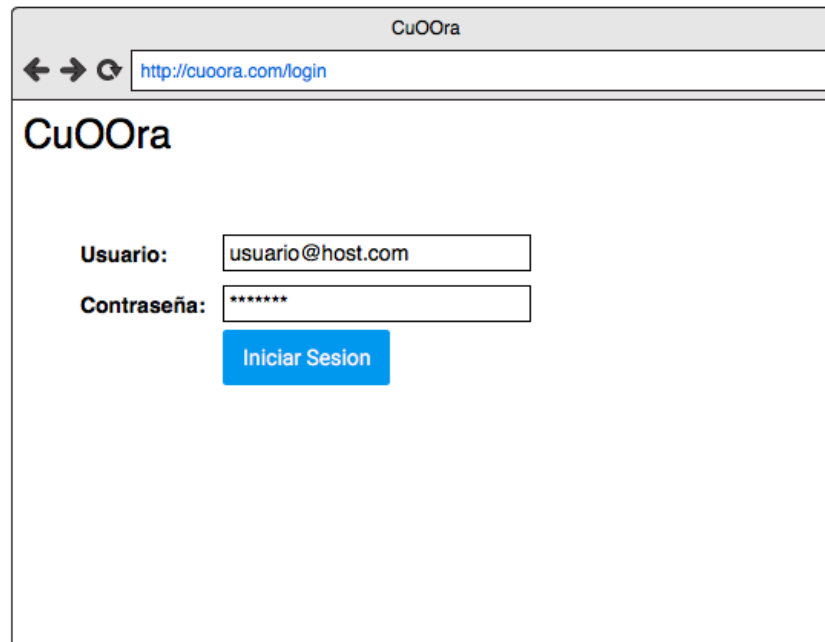
- Diseñar una solución y documentar con diagrama de clases UML.
- Implementar su diseño en Pharo.
- Diseñar e implementar test de unidad para su solución.

---

## Parte II: Construcción de CuOOra Web: login y home.

En esta parte del trabajo se espera que desarrollen las primeras partes de la aplicación Web CuOOra, que estará basada en el framework Seaside, y de la cual se brinda una especificación a continuación.

El login de la aplicación es muy simple, se muestra en la Figure 1. Se espera que la página principal del sitio CuOOra sea similar al mockup de la Figura 2. Una vez que el usuario ingresa haciendo el login, el *home* del sitio le muestra el listado de las últimas 5 preguntas relevantes ordenadas según su fecha de creación. Por cada pregunta, se muestra su autor, cuántas respuestas tiene, y la cantidad de likes y dislikes. Debe ser posible desde esta vista votar las preguntas listadas y también crear una pregunta nueva.



CuOOra

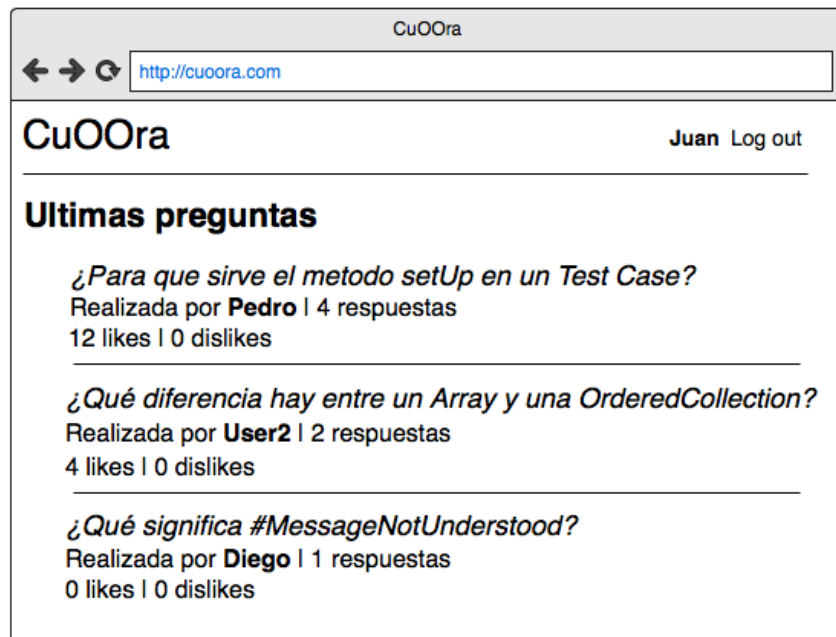
← → ↻

CuOOra

Usuario:

Contraseña:

**Figura 1.** Login de la aplicación



**Figura 2.** Diseño (final) orientativo del Home de la aplicación.

**Tareas:**

- Diseñar e implementar una aplicación Web con Seaside que implemente la UI siguiendo los mockups de este enunciado.

### Parte III: Construcción de CuOOra Web completa

En esta parte del enunciado, se refina el mockup para el home de la aplicación (mostrado en la Figura 1), y se definen nuevos mockups para el resto de la aplicación Web esperada.

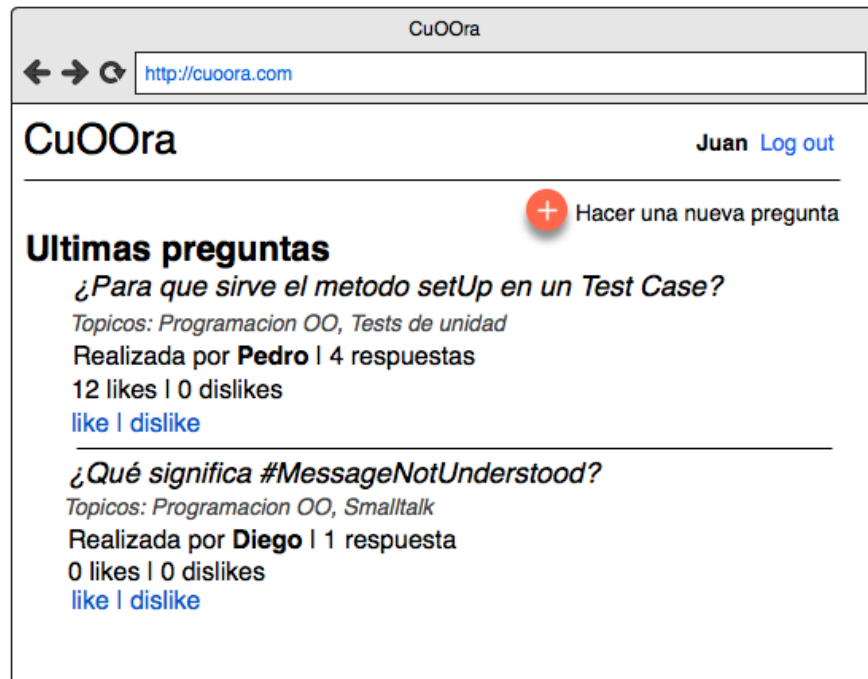


Figura 1. Diseño (final) orientativo del Home de la aplicación.

Cuando el usuario hace click en alguna de las preguntas, la aplicación debe mostrar el detalle de la misma, que se muestra en la Figura 2. Como puede apreciarse, se muestra la información de la pregunta, se ofrece la posibilidad de que el usuario la vote, y se listan todas sus respuestas.



Figura 2. Diseño orientativo de la vista de detalle de una pregunta.

Al hacer click en **Agregar Respuesta**, debe mostrarse una vista similar a la de la Figura 3.

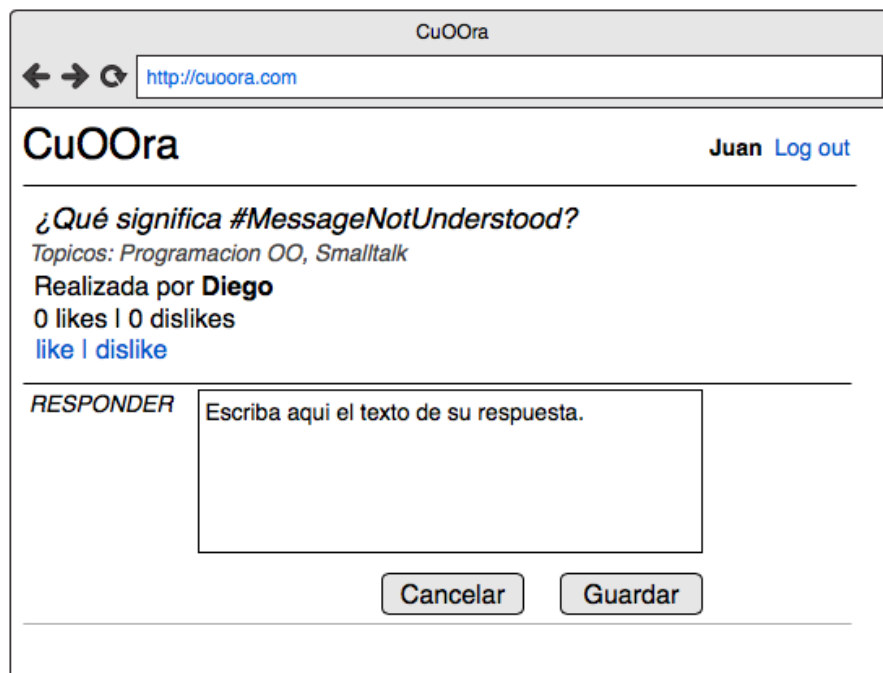


Figura 3. Creación de respuesta para una pregunta existente.

Cada vez que aparece el nombre de un usuario en pantalla, este debe ser un link que permite navegar al perfil de dicho usuario, donde se muestran solamente las preguntas realizadas por él,

ver Figura 4. Notar que en esta pantalla, el usuario que está navegando puede empezar a seguir (follow) al usuario cuyo perfil se está mostrando.



**Figura 4.** Perfil de un usuario.

Finalmente, cuando el usuario hace click en “Hacer una nueva pregunta” (ver Figura 1), se debe mostrar un formulario de creación similar al de la Figura 5, donde el usuario puede escribir el título de la pregunta, su desarrollo y los tópicos. Los tópicos se ingresan separados por coma.

The mockup shows a web browser window with the address bar displaying 'http://cuoora.com'. The page title is 'CuOOra'. In the top right corner, the user 'Juan' is logged in, with a 'Log out' link. The main heading is 'Nueva pregunta'. Below this, there are three input fields: 'Titulo' with the placeholder 'Escriba aqui el titulo de su pregunta.', 'Desarrollo' with the placeholder 'Escriba aqui el desarrollo de su pregunta.', and 'Tópicos' with the placeholder 'Ingrese tópicos separados por coma'. At the bottom right, there are two buttons: 'Cancelar' and 'Guardar'.

Figura 5. Creación de nueva pregunta

**Tareas:**

- Diseñar e implementar una aplicación Web con Seaside que implemente la UI siguiendo los mockups de este enunciado.

**Notas:**

- Los mockups son orientativos. Siempre y cuando se muestre la misma información, usted puede organizar el contenido como crea más conveniente (simple es mejor). No hace falta una pantalla para la creación de usuarios, pero sí una para el login de los mismos usuarios.
- Respecto de los mockups presentados en la Parte II, en este nuevo enunciado se modifican las UIs correspondientes al home de la aplicación y al detalle de una pregunta en pos de que el usuario pueda votar por like/dislike, como así también se muestran los tópicos de las preguntas.



### Preparando la entrega:

- Solo uno de los integrantes de cada grupo debe subir la entrega en la plataforma.
- Toda entrega debe incluir un archivo de texto con nombre "Información del Grupo" donde quede claro quienes conforman el mismo con nombre y número de alumno. En el mismo txt debe quedar escrita claramente la URL para acceder a su aplicación.
- La entrega debe incluir: modelo del dominio, test de unidad y aplicación Seaside. En lo posible en un solo .st generado mediante file out de un paquete correspondiente. No hay que subir la imagen. Si tiene dudas de cómo hacer el fileout del paquete que contiene su código, consulte en el foro. **Incluir en un archivo de imagen (JPG, PNG, etc) o PDF el diagrama de clases UML del modelo y de la aplicación Web.**
- Respecto de los tests de unidad: Se espera que haya tests de unidad para cada clase de su modelo, de las que deben probarse los métodos principales de su modelo (excluyendo getters y setters). No se requiere el uso de mock objects.
- Respecto de la aplicación Seaside:
  - Debe entregar su aplicación con una instanciación de prueba de su modelo que incluya:
    - Usuarios: pedro@cuoora.com, diego@cuoora.com, juan@cuoora.com con contraseñas idénticas al nombre de usuario (en caso que requieran contraseñas).
    - Las relaciones entre ellos son:
      - Juan sigue a Diego
      - Pedro sigue a Juan
    - Preguntas y respuestas:
      - Pregunta de Pedro:
        - ¿Para qué sirve el método SetUp?
        - Tópicos: OO1, Test de unidad.
        - Respuesta de Diego: Sirve para instanciar los objetos que son evaluados por el test en un único método y que se ejecute siempre antes de cada test.
      - Pregunta de Diego:
        - ¿Qué significa #messageNotUnderstood?
        - Tópicos: OO1, Smalltalk.
        - Respuesta de Pedro: Significa que el objeto que recibió el mensaje no encontró ningún método para ejecutar en respuesta.
  - En el comentario de clase del componente inicial de su aplicación (p.e., QuooraHomeComponent), incluya la expresión que hay que evaluar para registrar la aplicación (WAAdmin register: ... asApplicationAt: ...)