

Apellido y Nombre: ..... Legajo: .....

## Verdimeter

### Contexto

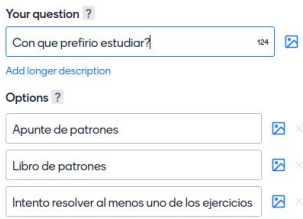
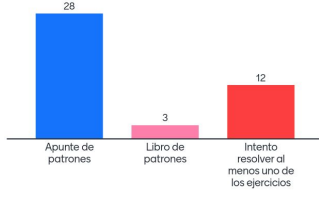
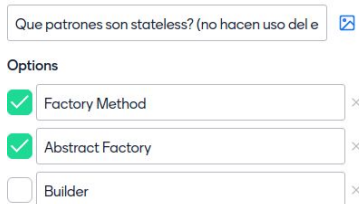
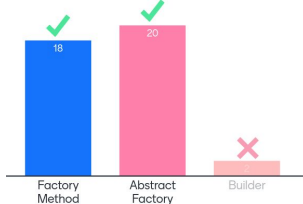
Verdimeter es un sistema de presentaciones online interactivas, el cual brinda funcionalidades para elaborar presentaciones y exponerlas al público de cierto evento. Permite que los asistentes al evento interactúen en tiempo real con la misma y que luego se analicen los resultados de dichas interacciones.

Un caso típico es el de un docente (en rol de presentador/expositor) que la utilizará para presentar el contenido de sus clases (evento) y que a su vez podría requerir la participación de los alumnos (en rol de participante) para que respondan a preguntas en tiempo real.

El sistema debe cumplir con la siguiente funcionalidad:

### Preparación

1. El expositor crea una presentación, la cual tiene un título y una descripción. Una presentación se compone de una secuencia de filminas. Solo el autor de la presentación es quien podrá editarla y/o ejecutarla.
2. El contenido de las filminas se puede presentar en dos modalidades: estático e interactivo.
  - a. Las estáticas son solo texto y gráficos (interpreta sólo html).
  - b. Las interactivas en principio tienen los siguientes formatos (se prevé que a futuro puede agregarse otros):

	Descripción	Resultado de la filmina
Encuesta	<p>Se14 presenta una pregunta y un listado de respuestas posible. La selección simple o múltiple. Se guardan las respuestas solo para generar una estadística.</p> 	<p>Se muestra el conteo de las respuestas de cada participante.</p> 
Pregunta	<p>Igual que la anterior, pero las respuestas pueden ser correctas o incorrectas. Cada pregunta correcta suma un punto y cada incorrecta resta uno.</p> 	<p>Igual al anterior, solo que se marcan las opciones como in/correcta. Se va guardando y acumulando el puntaje de los asistentes.</p> 
Texto Libre	<p>A partir de una pregunta, los participantes pueden escribir la respuesta en formato texto de a lo sumo 140 caracteres, solo una vez.</p>	<p>Se muestra lo que los usuarios escribieron.</p>

		<p>What would you like to say?</p> <div> <div>I am really glad that I got to voice my opinion during this interactive presentation.</div> <div>This was a very well prepared presentation, thanks for sharing your knowledge with us! It was great.</div> <div>I am really glad that I got to voice my opinion during this interactive presentation.</div> </div> <div> <div>I would like to ask the presenter about the first topic mentioned during this presentation.</div> <div>I think interaction is really fun.</div> <div>Could you please repeat that last part? I didn't quite understand exactly what you meant.</div> </div> <div> <div>I had a great time today!</div> <div>Thanks for the presentation.</div> <div>Good job!</div> </div> <div> <div>This was a really interesting way of conducting a presentation. I appreciate that you asked for</div> <div>This was a very well prepared presentation, thanks for sharing your knowledge with us! It was great.</div> <div>I am really glad that I got to voice my opinion during this interactive presentation.</div> </div>
--	--	--

## Exposición

1. Llegado el momento del evento, el expositor da inicio a una presentación de su autoría e invita a los participantes a unirse a la misma. Para realizar la invitación, el presentador comparte un link con un identificador único a la presentación, el cual se genera al ejecutar la presentación. No pueden haber exposiciones de una misma presentación en simultáneo.
2. Cuando un interesado entra al link, se registra a la presentación con un nombre y pasa a ser un participante de la misma, mostrándole la filmina actual. Esto puede suceder en cualquier momento mientras dure la presentación.
3. La exposición tiene una filmina actual, que es la que todos están viendo en un momento dado, y a su vez con diferentes perspectivas según el rol.
4. Según el caso, el participante interactúa con la filmina respondiendo a la consigna o solo observa el contenido. Cuando se envió la respuesta, esta es definitiva, no puede volver a responder y se queda esperando a la próxima filmina.
5. Cuando el presentador lo decide cierra la interacción con la filmina actual y se muestran los resultados de dicha filmina a ambos roles.
6. El presentador pasa a la siguiente filmina, dando por finalizada la filmina anterior y actualizando la visualización de todos. Los participantes no deciden que filmina ver, siempre deben ver el contenido de la actual. Los participantes no pueden interactuar con filminas ya finalizadas.

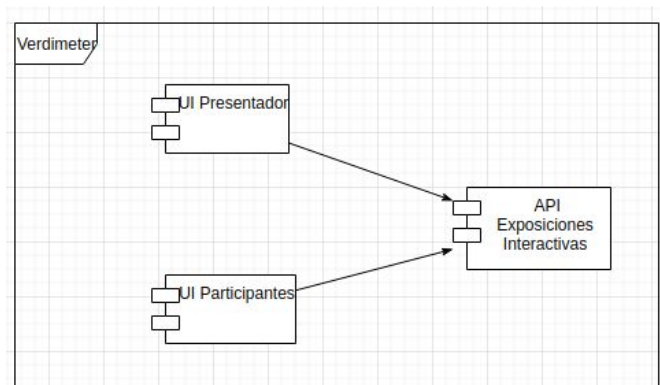
## Revisión de resultados

Cuando finaliza la presentación, el expositor puede conocer el estado final de cada una de las filminas interactivas. Estos datos deben permitirle al usuario comparar, ya por fuera del sistema, los resultados de distintas ejecuciones de la misma presentación, por ejemplo exportando los resultados en formato json.

El sistema tiene los siguientes componentes:

- **UI Presentador:** resuelve como se visualiza una exposición y manejan las interacciones para un presentador.
- **UI Participante:** resuelve como se visualiza una exposición y manejan las interacciones para un participante.
- **API Exposiciones Interactivas:** implementa la lógica de negocio de una exposición.

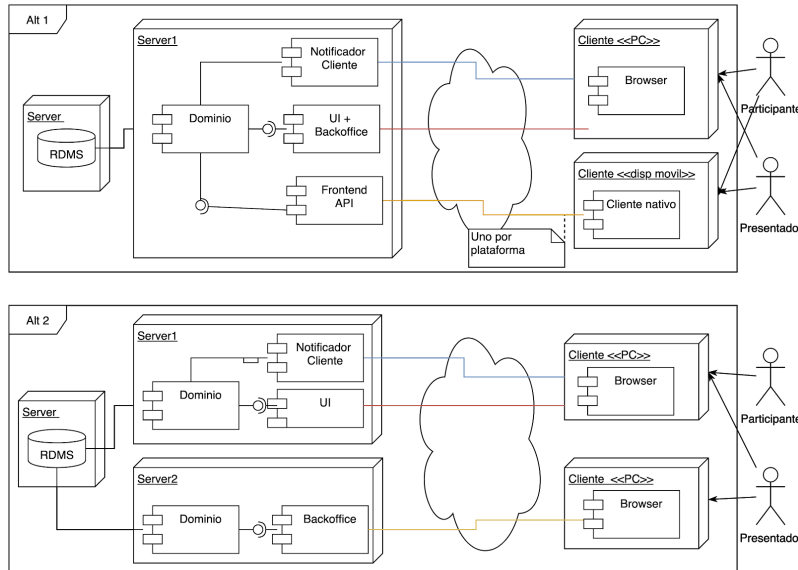
UI Presentador y UI Participante serán desarrolladas paralelamente por otro equipo.



## Punto 1 - Arquitectura (30 puntos)

Defina los siguientes atributos de calidad: portabilidad, usabilidad y fiabilidad (tolerancia a fallos).

Escoja un escenario que cuantifique cada atributo; luego muestre su aplicación en el contexto; y, por último, compare las alternativas.



Descripción de componentes:

- **UI:** Componente que tiene implementadas UI Presentador y UI Participantes
- **Backoffice:** ABM de presentaciones y consulta de resultados post exposición.
- **Notificador cliente:** Componente que avisa a los clientes cuando el presentador termina la interacción o pasa a la siguiente filmina.
- **Cliente nativo:** implementado para una tecnología en particular (Android, iPhone), permite visualizaciones más estéticas y puede almacenar datos.
- **Frontend API:** fachada / api REST para el cliente Nativo

## Punto 2 - Modelo de Dominio (40 puntos)

Realizar y comunicar el diseño del componente que resuelve la lógica de negocio aplicando el paradigma OO. Dando solución a la funcionalidad explicitada en el contexto.

- Documentar la solución utilizando diagramas UML (clases obligatorio). Si recurre a , prosa o pseudocódigo utilizarlo dentro de notas. (20)
- Justificar las decisiones de diseño que se tomen, por ejemplo haciendo referencia a los principios que guían al diseño vistos en clase o las consecuencias de aplicar un determinado patrón. (10)
- Realizar un diagrama de secuencia para comunicar cómo se resuelve el escenario “mostrar el contenido al participante Y de la filmina actual Z para la presentación X, que se configuró como tipo Pregunta” (10)

## Punto 3 - Persistencia (30 puntos)

Diseñar el modelo de datos del punto anterior para persistirlo en una DB relacional. Comuníquelo mediante un DER e indique:

- Qué elementos del modelo es necesario persistir;
- Las claves primarias, las foráneas y las restricciones según corresponda;
- Indicar cómo resolvió los impedance mismatch para persistir el modelo.
- Las estructuras de datos que deban ser desnormalizadas, si corresponde.

## Ejemplo

Momento / Vista	Presentador	Participante
Presentador inicia y envía link		
Participante se une con el link y filmina actual estática.		
Presentador pasa a la siguiente		
Inicia filmina interactiva encuesta		
Contesta el usuario		
Presentador termina interacción (resultados final de la filmina)		
Presentador pasa a la siguiente		
....		
Finaliza la presentación al pasar la última filmina		