

**Escuela Colombiana de Ingeniería**  
**Arquitecturas de Software – ARSW**  
**Parcial Segundo Tercio**

**Se permite:**

- Usar material hecho en el curso.
- Acceder a documentación técnica en línea.

**Implica anulación:**

- Tener abierto un cliente de correo electrónico o cualquier sistema de mensajería (redes sociales).

Una vez terminado el ejercicio, suba el código en formato .ZIP al espacio correspondiente en moodle.

## ***Da-freelancers***

“Da-freelancers” es una plataforma que permite poner en contacto a empresas que requieren de algún tipo de tarea especializada, con aquellas personas que las realizan bajo modalidad *freelance* (programadores, escritores, ingenieros, etc, que trabajan en sus casas con su propio horario).

Por ahora se quiere manejar una modalidad de subasta, en la que los ‘freelancers’ al enterarse de una oferta para realizar un trabajo, responden (en caso de que les interese) con la suma de dinero por la que ellos lo realizarían. La plataforma se encarga de procesar dichas respuestas y elegir automáticamente la más favorable teniendo en cuenta factores como experiencia, opiniones de otros empleadores, etc, aunque por ahora se limitará a elegir a la mejor alternativa económica.

La versión actual de la plataforma consta de una herramienta de escritorio a través de la cual los oferentes publican las tareas que requieren (proyecto ManejadorOfertas). Por otro lado, los ‘*freelancers*’ o potenciales candidatos a realizar tareas tienen una herramienta a la que se le notifica (bajo un esquema de mensajería de tipo publicador/suscriptor) cuando se haya publicado una nueva oferta. Por ahora, la herramienta de los freelancers no tiene interfaz gráfica, y es básicamente una simulación que en la medida que recibe ofertas, automáticamente realiza una propuesta económica aleatoria para la misma (proyecto FreeLancerApp), siguiendo un esquema cliente/servidor (a través del servicio remoto ‘agregarOferta’ de ManejadorOfertas). Tenga en cuenta que para

distinguir un ‘freelancer’ de otro, cada cliente genera una identidad aleatoriamente al iniciarse.

La versión actual de la plataforma está incompleta, pues aunque ManejadorOfertas está enviando eventos cada vez que se crea una tarea, las instancias de FreeLancerApp no están haciendo nada con los mismos una vez lo reciben.

1. [En el código][60%] Complete lo que haga falta para que:
  - a. Los *freelancers* registren su propuesta aleatoria una vez sean notificados de una nueva tarea.
  - b. El servidor (ManejadorOfertas) muestre en la ventana de ‘ofertas aceptadas’ la oferta ganadora una vez se hayan recibido las tres primeras propuestas (si esta información se muestra en la consola en lugar de la ventana, se valdrá 1/2 punto).
  - c. Los *freelancers* se enteren si fueron seleccionados para una tarea. Cuando una instancia del ‘freelancer’ se entere de que es ganador, éste debe imprimir en su pantalla “El freelancer XXXXX fue seleccionado para la tarea YYYY”.
2. [En la hoja de respuestas][10%] Bajo la configuración actual del servicio ofrecido por ManejadorOfertas, indique dos posibles inconsistencias que se podrían presentar, considerando que habrá muchas instancias de FreeLancerApp accediendo a dicho servicio.
3. [En el código][10%] Implemente una solución que prevenga las inconsistencias antes mencionadas, evitando (por eficiencia) hacer bloqueos innecesarios.
4. [En la hoja de respuestas][20%] Realice el diagrama de actividades UML de lo realizado (independientemente de que funcione o no), incluyendo las columnas ‘ManejadorOfertas’ y ‘FreeLancerApp’.