

Taller II – Curso 2018 - 2019

Trabajo de Laboratorio

Introducción

Se desea desarrollar una primera versión de un sistema de software a ser utilizado en el popular concurso televisivo “**Talento Uruguayo**”. Se trata de un certamen artístico donde los concursantes muestran sus talentos ante un jurado integrado por personalidades del ambiente artístico. Los participantes realizan performances (baile, canto, monólogo, etc.) y son evaluados por los jueces.



Los tres participantes que mayor puntaje reciban por parte del jurado tras realizadas todas sus performances serán puestos a elección del público en general, quien, por medio de su votación, elegirá finalmente al ganador (o ganadora) del concurso. El software a desarrollar servirá para que los jueces puedan evaluar y asignar puntos a las performances de los participantes y determinar finalmente cuáles son los tres participantes con mayor puntaje. También servirá para que el público pueda luego votar a través de Internet al ganador(a) del certamen entre esos tres participantes

Planteo del Problema

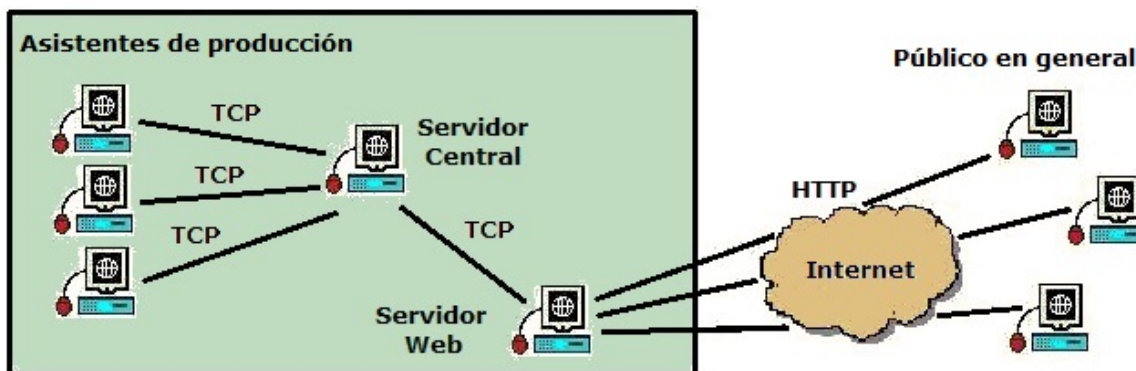
El certamen contará con dos tipos de participantes, que pueden ser nacionales o extranjeros. De cada participante se registrará su nombre artístico, su, foto, su nombre y apellido verdadero, su edad, su especialidad artística y la cantidad de votos que ha recibido del público (dicha cantidad será 0 al iniciar el concurso). El nombre artístico será el que identifique al participante dentro del certamen. Si se trata de un participante nacional, también se registrará el departamento del cual proviene junto con la localidad en la cual vive dentro del departamento. En caso de tratarse de un participante extranjero, lo que se registrará es su país de origen junto con la cantidad de años que lleva viviendo en Uruguay. No hay cota para la cantidad de participantes del certamen.

Cada participante realizará un total de 15 performances a lo largo del concurso. Cada vez que el concursante realice una performance, se registrará el número de su performance (de 1 a 15), una breve descripción de la misma y el puntaje (de 0 a 10) otorgado por cada uno de los 3 jueces a la performance. La primera performance realizada por el participante tendrá el número 1, la siguiente tendrá el número 2, y así sucesivamente.

Dado que los jueces pueden variar de una performance a otra (dependiendo de las características de la misma), el sistema no llevará registro alguno de los datos de los jueces. Es decir que lo único que se registrará de cada uno de los 3 jueces al momento de evaluar una performance es el puntaje que le otorga a la misma.

Arquitectura de la Aplicación

La aplicación a desarrollar tendrá dos tipos de usuarios: los asistentes de producción que ingresan los datos de concursantes y performances y el público en general. Los asistentes de producción la utilizarán desde las máquinas situadas junto al escenario donde se desarrollan las performances, mientras que el público en general accederá por Internet. En esta primera versión, no se requiere autenticación para ninguno de los usuarios. La siguiente figura ilustra la distribución de los diferentes componentes que conforman la aplicación.



Los equipos de los asistentes de producción estarán conectados mediante una L.A.N (Local Area Network) a través del protocolo TCP a un equipo que será el Servidor Central, el cual centralizará la lógica y la persistencia de toda la aplicación. Los equipos de los asistentes de producción correrán un programa cliente con una interfaz gráfica de ventanas que les permitirá acceder a sus funcionalidades. Habrá otro equipo que tendrá instalado un Servidor Web. El mismo recepcionará votos que el público realizará a través de Internet (mediante el protocolo HTTP) y ejecutará otro programa cliente que les permitirá emitir sus votos, el cual también accederá como cliente al Servidor Central mediante TCP. El Servidor Web responderá los pedidos mediante páginas JSP.

Requerimientos

La aplicación a desarrollar deberá resolver los siguientes requerimientos:

- 1) **Registro de un participante:** Se ingresarán los datos de un nuevo participante al sistema, incluida su foto, la cual será elegida por el asistente de producción al momento del ingreso. Este requerimiento será ejecutado en forma previa a comenzar el certamen. Se puede asumir que los datos de todos los participantes serán ingresados antes de que comience el concurso.
- 2) **Listado general de participantes:** Obtener un listado conteniendo nombre artístico, edad y especialidad artística de cada participante del certamen. Este listado debe realizarse ordenado alfabéticamente por nombre artístico. Servirá más tarde para que los asistentes de producción puedan seleccionar a qué participante le van a ingresar una performance o para acceder a listados detallados de los participantes.
- 3) **Listado detallado de un participante:** Dado el nombre artístico de un participante, obtener un listado detallado conteniendo todos sus datos personales. También se debe desplegar su foto, el total de puntos que el participante ha recibido de los jueces por las performances realizadas hasta el momento y la cantidad de votos recibidos por parte del público en general (dicha cantidad será cero mientras no se hayan definido los tres finalistas del concurso).
- 4) **Listado de performances de un participante:** Dado el nombre artístico de un participante, obtener un listado conteniendo cada una de las performances realizadas hasta el momento (número de performance, descripción de la misma y puntajes otorgados a la misma por los 3 jueces). Este listado debe realizarse ordenado por número de performance de menor a mayor.

- 5) **Cierre de performances y definición de finalistas:** Una vez que todos los participantes hayan hecho sus 15 performances, el sistema determinará cuáles fueron los tres participantes que obtuvieron más puntos por parte de los jueces (para ser luego colocados a disposición del público). En caso de que haya más de tres participantes con la mayor cantidad de puntos dados por los jueces, el sistema elegirá tres de ellos al azar. En caso de que al momento de ejecutar este requerimiento aún quede algún participante que no haya realizado sus 15 performances, el proceso no será llevado a cabo, emitiendo un mensaje de error apropiado.
- 6) **Ingreso de performance de un participante:** Dado el nombre artístico que identifica a un participante, la descripción de una performance y los puntajes otorgados por los 3 jueces, ingresar en el sistema la performance correspondiente hecha por ese participante. El número de la nueva performance siempre será consecutivo al de la última performance realizada por dicho participante. En caso de que ya se haya realizado el cierre de performances (y se hayan definido los 3 finalistas del certamen) o de que el participante ya haya realizado sus 15 performances, el proceso no será llevado a cabo, emitiendo un mensaje de error apropiado.
- 7) **Cierre de votación del público:** Registrar en el sistema que la votación por parte del público se acaba de cerrar. Este requerimiento será ejecutado una vez que la producción del certamen haya decidido finalizar el concurso. En ese momento, el sistema determinará cuál de los 3 finalistas obtuvo la mayor cantidad de votos del público y lo declarará como ganador. En caso de que haya dos finalistas con la mayor cantidad de votos, el sistema elegirá uno de ellos al azar. En caso de que aún no se hayan definido los tres finalistas, el proceso no será llevado a cabo, emitiendo un mensaje de error apropiado.
- 8) **Respaldo de datos:** Respaldo en disco todos los datos de la aplicación. Este requerimiento podrá ejecutarse toda vez que algún asistente de producción lo desee, especialmente luego de haber ejecutado funcionalidades que produzcan cambios en la información del sistema. Todos los datos se respaldarán juntos en un único archivo binario en disco (ubicado en el Servidor Central), para luego poder ser restaurados a memoria en una próxima ejecución.
- 9) **Listado de finalistas y ganador del certamen:** Obtener un listado con los tres finalistas del certamen. De cada uno de ellos se listará nombre artístico, edad y cantidad de votos del público recibidos hasta el momento. Opcionalmente, también se mostrará su foto. Este requerimiento servirá para que el público pueda elegir a quién votar o simplemente para saber cómo va la votación. En caso de que la votación del público ya haya finalizado, se listará además cuál de los 3 fue el ganador del certamen. En caso de que aún no se hayan definido los tres finalistas, el proceso no será llevado a cabo, emitiendo un mensaje de error apropiado.
- 10) **Votación de un ganador:** Dado el nombre artístico que identifica a un participante, otorgarle un voto para ayudar a que gane el certamen. En caso de que aún no se hayan definido los tres finalistas o la votación del público ya se haya cerrado, el voto no será registrado, emitiendo un mensaje de error apropiado. Este requerimiento le permitirá al público en general votar por un ganador. Toda persona podrá acceder al sitio y votar tantas veces como desee.

Los requerimientos 1 a 8 serán ejecutados por los asistentes de producción desde los equipos ubicados junto al escenario. Los requerimientos 9 y 10 serán ejecutados por el público en general por Internet. Estos serán los únicos requerimientos del sistema a los que el público tendrá acceso.

Lea con atención todos los requerimientos. No olvide hacer todos los chequeos que sean necesarios en forma previa a la resolución de cada requerimiento.

Primera Entrega - Análisis y Diseño

Fecha de entrega: **Lunes 11 o Martes 12 de Febrero de 2019** (según el día en que le toque la tutoría al grupo, en horario de clase). Se debe entregar la solución a las partes A y B. Las mismas deberán entregarse **impresas** en un documento debidamente organizado.

Parte A: Diagrama de clases conceptual de UML correspondiente a la realidad presentada en el planteo del problema hecho anteriormente.

Parte B: Elección de un diseño para resolver la lógica de la aplicación, desglose de requerimientos para el diseño elegido, especificación completa en notación UML (diagrama de clases de implementación) y elección de estructuras de datos apropiadas, justificando.

Segunda Entrega - Capa Lógica y Capa de Persistencia

Fecha de entrega: **Miércoles 27 de Febrero de 2019** (vía mail al docente, antes de las 24 horas). Se debe entregar, en un archivo zippeado, el código fuente que da solución a las partes C, D, E y F.

Parte C: Implementación en JAVA de todas las clases y métodos de la capa lógica definidos en las etapas de análisis y diseño. También se deben implementar las clases y métodos necesarios para respaldar y recuperar la información desde disco (capa de persistencia).

Parte D: Creación de programas de prueba que permitan verificar el funcionamiento de todas las clases y métodos implementados en la parte anterior. Para esta parte, no se pretende manejo de interfaz gráfica. Alcanza con desplegar información en la salida estándar.

Parte E: Implementación en JAVA del programa servidor de toda la aplicación mediante el uso de RMI para la comunicación remota de los programas cliente con el programa servidor.

Parte F: Creación de programas cliente de prueba que permitan testear la comunicación con el programa servidor desde los equipos cliente. Al igual que antes, no se pretende un manejo de interfaz gráfica. Alcanza con desplegar información en la salida estándar.

Tercera Entrega - Capa Gráfica

Fecha de entrega: **Martes 12 de Marzo de 2019** (vía mail al docente, antes de las 24 horas). Se debe entregar, en un archivo zippeado, el código fuente que da solución a las partes G y H (y también la totalidad de los archivos fuente de toda la aplicación).

Parte G: Implementación en JAVA de todas las clases necesarias para el manejo de la interfaz gráfica de ventanas para el programa cliente de los asistentes de producción. Se utilizarán paquetes con componentes gráficos de Java. Además se deberá realizar integración de la capa gráfica con el programa Servidor a efectos de verificar el funcionamiento de los requerimientos.

Parte H: Implementación de Servlets de Java que permitan ejecutar los requerimientos 9 y 10 al público en general y también la creación de páginas JSP que provean una interfaz web para el mismo. También se deberá realizar integración de la capa web con el programa Servidor a efectos de verificar el funcionamiento de los requerimientos.