



TALLER 2

ENTREGA FINAL

TRABAJO DE LABORATORIO

CURSO NOCTURNO 2018 - 2019

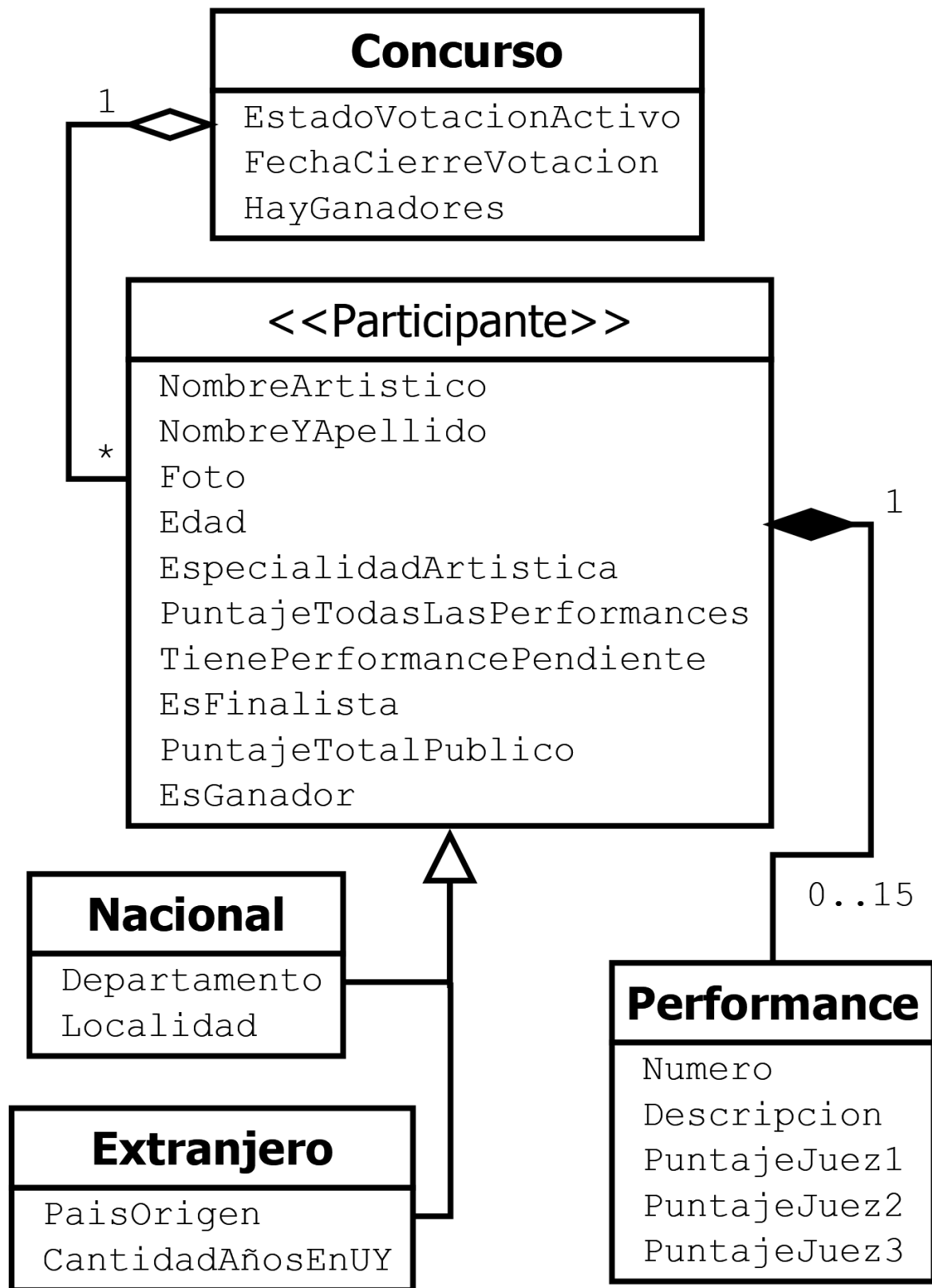
LICENCIATURA EN INFORMÁTICA

Tutor:	Suarez, Gabriel	
Integrantes:	Pías, Richard	- 1.924.591-2
	Segovia, Joaquín	- 4.739.544-4
	Torres, Mathias	- 4.223.291-4

Tabla de contenido

1. (Parte A) Diagrama de clases conceptual de UML	3
2. (Parte B) Elección del diseño	4
2.1. Análisis: tipos de datos abstractos (TAD)	4
3. (Parte B) Desglose de requerimientos	5
3.1. Registro de un participante	5
3.2. Listado general de participantes	5
3.3. Listado detallado de un participante	6
3.4. Listado de performances de un participante	6
3.5. Cierre de performances y definición de finalistas	7
3.6. Ingreso de performance de un participante	7
3.7. Cierre de votación del público	8
3.8. Respaldo de datos	8
3.9. Listado de finalistas y ganador del certamen	9
3.10. Votación de un ganador:	9
4. (Parte B) Especificación de diseño: Diagrama de clase de implementación	10
4.1. Estructuras de datos elegidas para representar las colecciones	10
4.2. Diagrama de clase de implantación (UML)	11
5. Wireframes para la interface tanto “win” como “web”	13
5.1. Pantallas “WIN”	13
5.2. Pantallas “WEB”	17

1. (Parte A) Diagrama de clases conceptual de UML



2. (Parte B) Elección del diseño

2.1. Análisis: tipos de datos abstractos (TAD)

A continuación, se presentan el resultado del análisis de los datos de tipos abstractos (TAD). Para ello se tuvieron en cuenta tanto la lógica como los requerimientos.

- a) Participantes = Diccionario (Participante)
- b) Performances = Secuencia (Performance)
- c) Finalistas = Diccionario (Participante)
- d) Participante = (NombreArtístico X NombreYApellido X Foto X Edad X EspecialidadArtística X PuntajeTodasLasPerformances X TienePerformancePendiente X EsFinalista X PuntajeTotalPublico X EsGanador X Performances)
- e) Nacional = (Participante + Departamento X Localidad)
- f) Extranjero = (Participante + PaisOrigen X CantidadAñosEnUY)
- g) Performance = (Numero X Descripcion X PuntajeJuez1 X PuntajeJuez2 X PuntajeJuez3)
- h) Concurso = (EstadoVotacionActivo X FechaCierreVotacion X HayGanador)

- i) {NombreArtístico, NombreYApellido, Foto, EspecialidadArtística, Departamento, Localidad, PaisOrigen, Descripcion} = String
- j) {Edad, PuntajeTodasLasPerformances, PuntajeTotalPublico, CantidadAñosEnUY, Numero, PuntajeJuez1, PuntajeJuez2, PuntajeJuez3} = Entero
- k) {TienePerformancePendiente, EsFinalista, EsGanador, EstadoVotacionActivo} = Bool
- l) FechaCierreVotacion = Fecha

Los siguientes puntos corresponden a la justificación.

- a) La colección de Participantes es representada como un diccionario ya que es un conjunto no acotado de elementos, no repetidos y no ordenados. Contiene a n Participante y cada uno de ellos se identifican por su nombre artístico (cadena de texto o String). La colección es única en todo el sistema. No hay duplicados de elementos Participante.
- b) La colección de Performances, se expresa como una secuencia acotada (de un total de 15 elementos en base a la realidad analizada), y son secuenciales, consecutivos; ya que esta estructura nos permite modelar correctamente la realidad de las performances que realiza un participante a lo largo de todo el concurso. Esta colección de performances se repite n veces como tantos participantes cargados en el sistema existan. Una colección por cada participante. La principal justificación de utilizar secuencia y no diccionario, está en la necesidad, de una búsqueda en concreto.
- c) La colección de Finalistas, es muy parecida a la de Participantes y se planteó como un diccionario puesto que por ejemplo, no está acotada en cuanto a cantidad de elementos, entre otras características mencionadas anteriormente en el punto a. Esta es una colección auxiliar para resolver el problema.
- d) El Participante, se representa como un producto cartesiano donde se guardan todos los datos inherentes a ellos tales como su identificador, nombre, foto, puntaje y estado respecto al concurso, entre otros.
- e) Los participantes de tipo Nacional, contienen los mismos datos que el producto cartesiano Participante más sus propios atributos, como lo son su Departamento y Localidad.
- f) Análogamente, los participantes de tipo Extranjero, contienen los de Participante más el nombre del país de origen y la cantidad de años que lleva viviendo en Uruguay.
- g) El producto cartesiano Performance, contiene los datos relacionados a cada performance realizada por un Participante, como son su número identificador, descripción de la misma y los puntajes de cada juez.
- h) Por último, el producto cartesiano Concurso, contiene el estado activado o desactivado del proceso de votación abierta al público, y cuándo fue el cierre este último mencionado. Además la fecha de dicha ejecución de cierre. También si hay un ganador.

3. (Parte B) Desglose de requerimientos

3.1. 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.

R1	
Entrada	Datos de un nuevo participante
Salida	Nuevo participante cargado al sistema
Proceso	A partir del identificador del participante, verifico si existe en el sistema SI ya pertenece ENTONCES Error, el participante ya está cargado al sistema SINO Creo el participante pidiendo datos extras si es Nacional o Extranjero Ingreso el participante al sistema FIN SI

3.2. 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.

R2	
Entrada	No tiene entrada
Salida	Listado reducido de participantes ordenados alfabéticamente
Proceso	MIENTRAS haya participante en la colección participantes HACER Agrego el participante con sus datos, a la lista, ordenado alfabéticamente FIN MIENTRAS Devuelvo la lista

3.3. 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).

R3	
Entrada	Identificador de participante (NombreArtistico)
Salida	"Listado" detallado de un participante
Proceso	A partir del identificador SI el participante no existe ENTONCES Error, el participante no existe SINO Obtengo el participante Agrego todos los datos del participante al "listado" Agrego los datos extras dependiendo el tipo al "listado" FIN SI Devuelvo el "listado" detallado de un participante

3.4. 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.

R4	
Entrada	Identificador de participante (NombreArtistico)
Salida	Listado de las performances realizadas hasta ese momento ordenado por su numero
Proceso	A partir del identificador SI el participante no existe ENTONCES Error, el participante no existe SINO Obtengo el participante del identificador ingresado Obtengo las performances del participante PARA CADA performance de la colección performances HACER Agrego datos de la performance al listado FIN PARA CADA FIN SI Devuelvo el listado

3.5. 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.

R5	
Entrada	No tiene entrada
Salida	Cierre de carga de performances y definición de finalistas
Proceso	<p>QuedaAlguno = false</p> <p>MIENTRAS (haya participantes) y (!QuedaAlguno: no haya participante con performance pendiente) HACER</p> <p> SI el participante tiene pendiente ENTONCES</p> <p> QuedaAlguno = true</p> <p> FIN SI</p> <p>FIN MIENTRAS</p> <p>SI (QuedaAlguno=true): hay algún participante que falte realizar alguna performance pendiente ENTONCES</p> <p> Error, aún faltan performances para realizar</p> <p>SINO</p> <p> Cargo la colección, obtengo los 3 finalistas (algoritmo: Ordena, valúa puntos y activa a los finalistas)</p> <p> Activo votación del público (Punto 3.10 – Requerimiento 10)</p> <p>FIN SI</p>

3.6. 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.

R6	
Entrada	Identificador de un participante y datos de una performance
Salida	Performance cargada al sistema y asociada al participante
Proceso	<p>SI (la votación está abierta) o (si hay ganador) ENTONCES</p> <p> Error, no se pueden ingresar más performances</p> <p>SINO</p> <p> A partir del identificador</p> <p> SI el participante no existe ENTONCES</p> <p> Error, el participante no existe</p> <p> SINO</p> <p> Obtengo el participante ingresado</p> <p> Obtengo las performances del participante</p> <p> SI no se pueden cargar más performance ENTONCES</p> <p> Error, ya tiene el máximo de performances permitidas</p> <p> SINO</p> <p> Creo la nueva performance</p> <p> Cargo la nueva performance</p> <p> Aumento el total de puntaje de performances del participante</p> <p> FIN SI</p> <p> FIN SI</p> <p>FIN SI</p>

3.7. 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.

R7	
Entrada	No tiene entrada
Salida	Registro de cierre y determinación de ganador
Proceso	<p>SI no está abierta la votación ENTONCES Error, la votación nunca se activó, no puede cerrar ningún proceso</p> <p>SINO</p> <p>Cierro votación del publico Obtengo el id del ganador (algoritmo: recorre, cuenta, determina) A partir del identificador del participante, lo determino como ganador (para el Punto 3.9 – Requerimiento 9)</p> <p>FIN SI</p>

3.8. 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.

R8	
Entrada	No tiene entrada específica (si todos los datos en memoria)
Salida	Datos guardados y persistidos
Proceso	<p>A partir de todos los datos en memoria Persisto todos los datos/cambios Retorno mensaje al usuario: "Cambios guardados correctamente"</p>

3.9. 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.

R9	
Entrada	No tiene entrada
Salida	Listado de los finalistas y eventual ganador
Proceso	<p>SI la colección de finalistas esta vacía Error, aun no se determinaron los finalistas</p> <p>SINO PARA CADA participante de la colección de finalistas HACER SI la votación está abierta ENTONCES Permito votarlo (Punto 3.10 – Requerimiento 10) Agrego los datos del finalista al listado</p> <p>SINO No permito votarlo SI el participante es ganador ENTONCES Agrego los datos del finalista al listado como ganador</p> <p>SINO Agrego los datos del finalista al listado</p> <p>FIN SI</p> <p>FIN SI</p> <p>FIN PARA CADA Devuelvo el listado</p> <p>FIN SI</p>

3.10. 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.

R10	
Entrada	Identificador de un participante
Salida	Participante con 1 votos del publico sumado
Proceso	<p>SI el estado de la votación está activo ENTONCES A partir del identificador de un participante finalista Obtengo al participante Agrego 1 voto al contador del participante</p> <p>SINO Error, no puede votar</p> <p>FIN SI</p>

4. (Parte B) Especificación de diseño: Diagrama de clase de implementación

4.1. Estructuras de datos elegidas para representar las colecciones

Diccionario de Participantes: Los colección Participantes se define como un diccionario, ya que es no acotada, no ordenada, y sus elementos se identifican por su código string, siendo que estos últimos no se repiten.

Como contiene elementos que corresponden a una herencia, podemos decir que es una colección polimórfica.

La estructura elegida para su representación, es el Árbol Binario de Búsqueda (ABB), ya que nos permite guardar sus elementos ordenadamente por su identificador, ya que un requerimiento nos solicita listar a los participantes ordenados alfabéticamente, por su Nombre Artístico (siendo este su identificador).

Cabe destacar que como el código puede contener, además de letras y números, caracteres “raros”, basamos el orden alfabético en el orden del código ASCII.

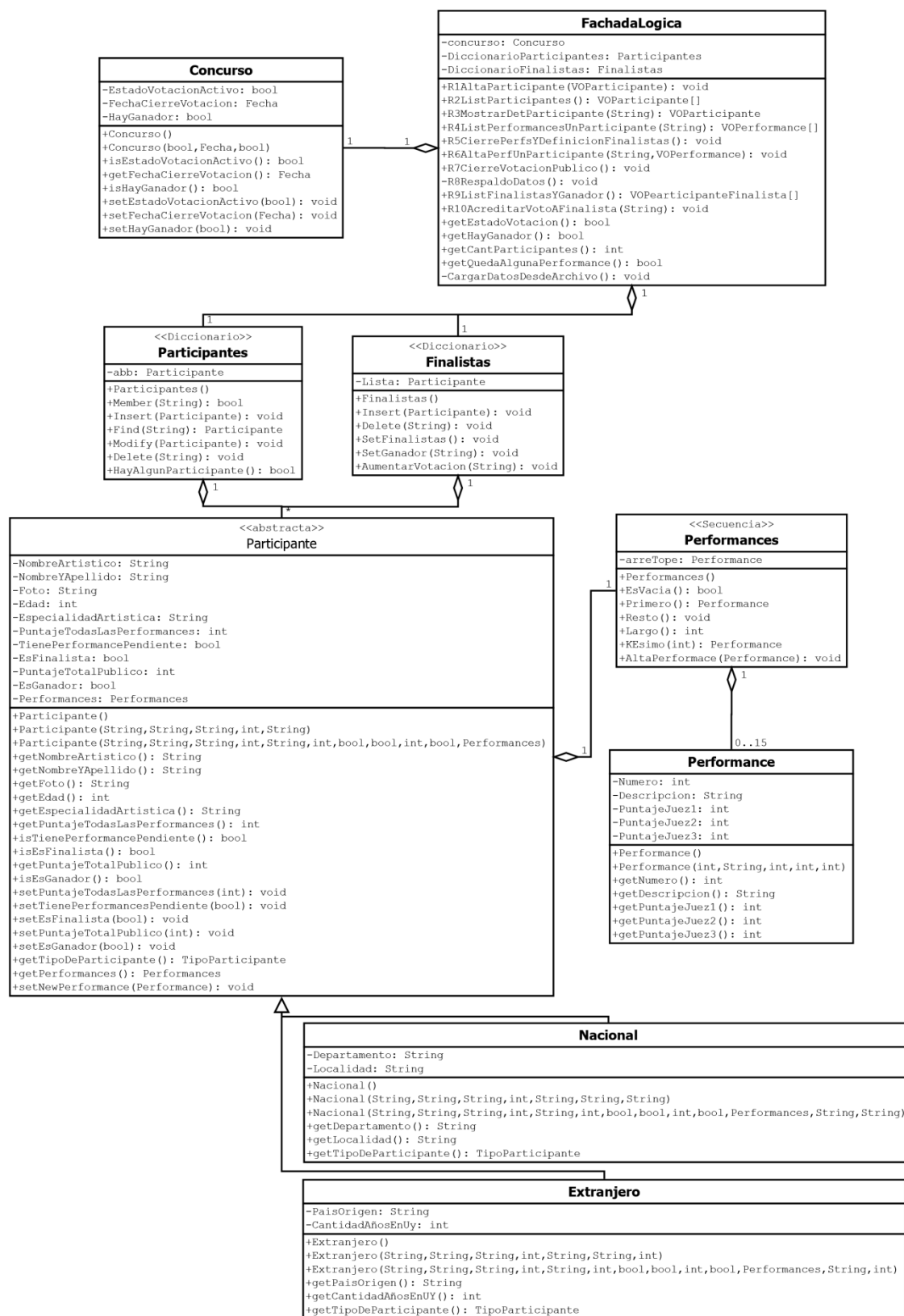
Secuencia de Performances: Las colección Performances se especifica como una secuencia, puesto que es acotada, sus elementos son secuenciales y consecutivos. Estos últimos, se identifican con un número de performance y no se pueden repetir.

Como cada colección tiene un máximo de 15 elementos, se optó por la estructura de arreglo con tope.

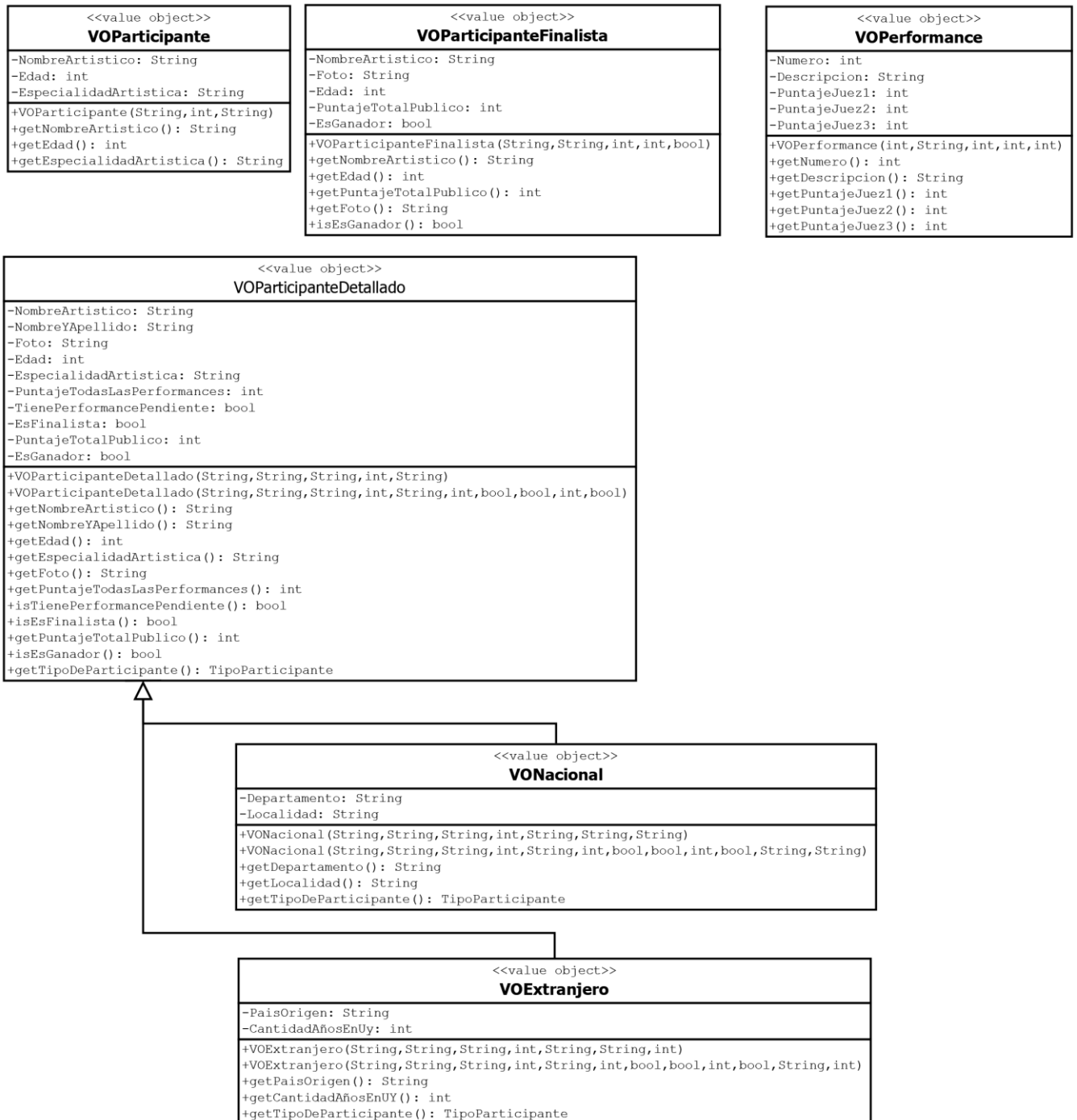
Esta estructura nos facilita el acceso a los datos de una performance para resolver uno de los requerimientos, donde no se recorre toda la colección, sino que con su identificador, se accede a sus datos directamente en su posición.

Diccionario de Finalistas: Colección auxiliar para trabajar con algoritmos. Es análoga a la primera, excepto que en lugar de ABB, es una Lista. No existe la necesidad de ordenar sus elementos por su id, sino por otro atributo.

4.2. Diagrama de clase de implantación (UML)



***CONTINÚA EN LA SIGUIENTE HOJA



****Se adjunta a este documento, un .pdf con los UML en formato vectorial para una mejor visualización.***

5. Wireframes para la interface tanto “win” como “web”

5.1. Pantallas “WIN”

Al abrir la aplicación “WIN”, esta muestra un contenedor principal, el cual contiene en la parte superior izquierda, un “menú” para iniciar efectivamente el programa cliente.

PANTALLA PRINCIPAL:

Alineados a los requerimientos, buenas prácticas de diseños de interfaces, y mejores heurísticas de experiencia de usuario, se decidió que la primera pantalla (Imagen 1) cumpla con un requerimiento esencial, a la hora de administrar el concurso, sin la necesidad de una acción extra del usuario.

Concurso - Mantenimiento de Participantes

Logo Concurso

Registrar Participante

Lista de Participantes

Nombre Artístico	Edad	Especialidad Artística	Seleccionar uno
Al - El Zorro de Salto	40	Guitarrista	<input checked="" type="radio"/>
Bee Juan Malambero	38	Malambo	<input type="radio"/>
Dou - El Pancho	31	Actor	<input type="radio"/>
La mas grande	25	Acordionista	<input type="radio"/>
The Norteño	28	Malabarista	<input type="radio"/>

Ver detalle del participante - Administrar performances

Cierre de performances, def. finalistas - Abrir Votacion

Cerrar votacion - Obtener Ganador

Imagen 1.

Al iniciar efectivamente, se abre la primera pantalla, la cual en la zona media, presenta una grilla con el listado general (simple) de los participantes, hasta el momento registrados, resolviendo así, el **requerimiento número 2**.

En la parte superior derecha, se encuentra un botón (“Registrar Participante”) el cual nos permite ingresar un nuevo participante al sistema; a continuación se explica la pantalla (Imagen 2) a la cual podemos navegar haciendo clic en dicho botón, cumpliendo así, el **requerimiento número 1**.

Volviendo a la grilla, ésta se encuentra en una zona encerrada (“Lista de Participantes”), junto a un botón (“Ver detalle del participante - Administrar performances”) en la parte inferior derecha, que nos permite navegar a otra ventana (Imagen 3) para ver los datos completos (**requerimiento número 3**) de un participante seleccionado en la grilla, ver las performances ingresadas hasta el momento (**requerimiento número 4**), y la posibilidad de agregar una nueva (**requerimiento número 6**). Justamente para que el flujo funcione, se deberá seleccionar un participante de la grilla.

Por último, esta primera ventana, presenta un botón (“Cierre de performances, def. finalistas - Abrir Votación”) ubicado en la parte inferior izquierda, el cual cierra el concurso, definiendo los finalistas y habilitando la votación vía web del público; resolviendo el **requerimiento número 5**.

También un botón (“Cerrar votación - Obtener Ganador”) ubicado en aparte inferior derecha el cual cierra la votación vía web del público; resolviendo el **requerimiento número 7**.

REGISTRAR UN PARTICIPANTE:

Como anteriormente se mencionó, el botón (“Agregar Participante”), despliega la pantalla (Imagen 2) que permite ingresar todos los datos inherentes a un participante, incluido sus datos característicos si es de tipo Nacional o Extranjero; cumpliendo así, el **requerimiento número 1**.

De entre todos los campos para ingresar texto, podemos destacar un botón (“Cargar foto”) el cual permite ingresar una imagen al participante. Además, un “combo box” (o “seleccionador” de opción), para poder agregar datos dependiendo del tipo de participante.

Por último, un botón (“Cancelar”) para salir de esta pantalla y volver a la principal, en contraste inferior con otro botón principal (“Guardar”).

Este último, esta acentuado en el diseño, ya que es el “call to action” principal de este formulario; guardar un nuevo participante en el sistema.


Mantenimiento de Participantes - Registrar un Participante

Datos Participante

Nombre Artístico

Nombre y Apellido

Edad

Fotografía 

Tipo Participante

Datos Participante Nacional/Extranjero

DatoExtra1

DatoExtra2

Imagen 2.

VISUALIZAR UN PARTICIPANTE:

En la pantalla principal, al seleccionar un participante de la grilla, podemos navegar mediante el botón (“Ver detalle del participante - Administrar performances”) hacia la siguiente pantalla.

The screenshot shows a web application interface titled "Mantenimiento de Participantes - Ver Participante". It is divided into two main sections: "Datos Participante" and "Lista de Performances".

Datos Participante: This section includes a placeholder for a profile picture and several fields for participant information:

Nombre Artístico	TheMagicMan	Nombre y Apellido	Pepe Perez
Tipo Participante	Nacional	Edad	25
Departamento	MVD	Localidad	Cordon
Puntaje de Jueces	131	Cumulo de votos	0

Lista de Performances: This section contains a table with 5 rows of performance data. Each row has columns for the performance number, description, and scores from three judges.

Numero	Descripcion	PuntajeJurado1	PuntajeJurado2	PuntajeJurado3
1	Baile	1	4	7
2	Canto	3	43	4
3	Baile	23	22	2
4	Pinta	3	3	3
5	canta	4	2	7

Below the table is a button labeled "Agregar Performance". At the bottom right of the screen is a button labeled "Volver".

Imagen 3.

Dicha pantalla, nos muestra todos los datos relacionados a un participante (**requerimiento número 3**) y también todas sus performances (**requerimiento número 4**).

Se despliega un botón (“Volver”), que no está “resaltado”, ya que no es una acción importante, sino que sirve para volver a la pantalla principal.

Por último, dentro de la sección “Lista de Performances”, se muestra un botón (“Agregar Performance”) el cual nos permite navegar hacia la pantalla (Imagen 4) para agregar una performance al participante (**requerimiento número 6**).

REGISTRAR PERFORMANCE:

El formulario para agregar una performance, solo es accesible desde el detalle de visualización de un participante, ya que el ingreso de una, está directamente relacionado a un participante.

Se muestran los datos básicos del participante para darle contexto al usuario en la sección “Datos Participante”. En la parte inferior a ésta, se muestra la zona principal llamada “Datos Performance”.

En ésta zona principal, se ingresan los datos relacionados a una performance. Cabe destacar, se optó que el número de performance será autogenerado, ya que son consecutivos en el sistema, evitando posibles errores.

Nuevamente, al igual que en la pantalla de ingreso de participante, se encuentra un botón (“Cancelar”) para salir de esta pantalla, pero esta vez, volver a al detalle de participante, respetando la navegación.

En contraste al botón anterior, también se muestra el botón principal (“Guardar”). Análogamente a la pantalla para ingresar participante, se enfatiza en el diseño, ya que es el “call to action” principal de este formulario; guardar una nueva performance al participante, en el sistema (**requerimiento número 6**).

The screenshot shows a web form titled "Mantenimiento de Participantes - Agregar Performance". It is divided into two main sections: "Datos Participante" and "Datos Performance".

Datos Participante: This section contains a placeholder icon for a profile picture, the "Nombre Artístico" (TheMagicMan), and the "Nombre y Apellido" (Pepe Perez).

Datos Performance: This section contains several input fields:

- Numero:** A text field containing "XX".
- Descripcion:** A large text area with the placeholder text "Descripcion de performance".
- Puntaje Jurado 1:** Three identical text fields, each containing the placeholder text "Puntaje 1".

At the bottom right of the form, there are two buttons: "Cancelar" and "Guardar".

Imagen 4.

5.2. Pantallas “WEB”

Al abrir la página web del concurso, si no se definieron los finalistas, se muestra un mensaje indicando al público que la etapa de performances no termino y que la votación aún no está abierta.

LISTADO Y VOTO DE FINALISTAS:

Una vez definido los finalistas y abierta la etapa de votación del público, se muestra una pantalla web con los n (en este caso 3) finalistas, cumpliendo con la primera parte del **requerimiento número 9**, junto a los datos más relevantes para el público, en función de esta etapa. O sea, poder identificar claramente el finalista que quiere votar y si su voto fue contado por el sistema (“Cantidad de votos”). Para esta última acción, se muestra un botón (“Votar!!!”) para que el público pueda otorgar un voto, cumpliendo con el **requerimiento número 10**.

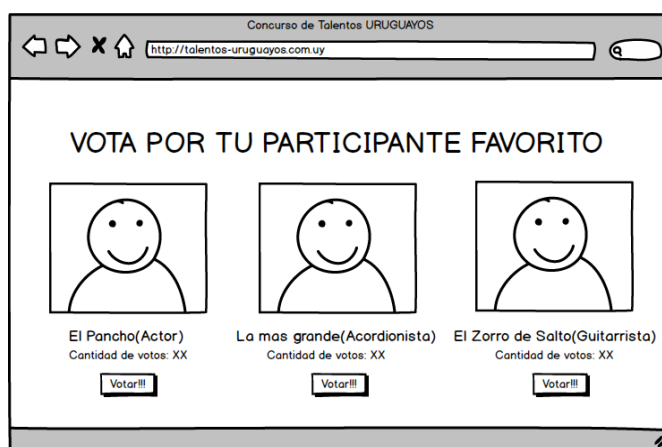


Imagen 5.

GANADOR DEL CERTAMEN:

Cumpliendo con la segunda parte del **requerimiento número 9**, en caso de que la votación del público ya haya finalizado, la opción de votar no estará disponible, ocultando el botón.

Además la interface cambia, exponiendo y dejando bien claro cuál fue ganador, o sea, cuál fue el finalista que obtuvo la mayor cantidad de votos del público.

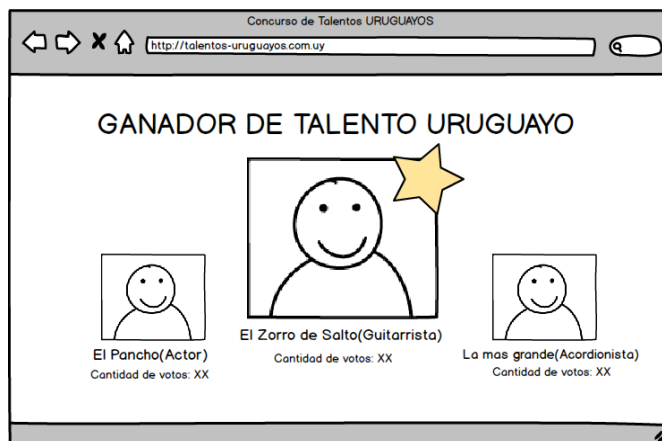


Imagen 6.

