

Grupo de trabajo 5

Semestre 2022-20

Análisis del proceso seguido para el desarrollo de los proyectos del curso

1) Procesos seguidos para los proyectos 1, 2 y 3

Proyecto 1

Para la realización del primer proyecto, como equipo nos centramos en primero realizar el análisis y diseño de manera conjunta. Para esto, comenzamos por identificar cuáles eran los roles necesarios para construir una solución al problema y qué elementos (modelados por medio de objetos de clases) debían poder cumplir responsabilidades asignadas a esos roles. Posteriormente, construimos un diagrama de clases inicial y le asignamos las relaciones entre clases, los tipos de relaciones (herencia, composición, agregación, delegación) y los atributos que inicialmente habíamos imaginado cada uno tendría.

Al definir luego las diferentes responsabilidades de cada tipo de actor dentro del mundo del problema e identificar funcionalidades necesarias que se debían cumplir, pasamos a pensar en los diferentes métodos que le debíamos asignar a cada clase.

Teniendo en cuenta esta información, hicimos un plan inicial para avanzar en la construcción de cada una de las clases necesarias para cumplir con la implementación de la lógica. Comenzamos a definir los atributos y métodos necesarios para realizar un registro exitoso de cualquier participante y luego las funciones que podrían permitir tanto a un administrador como a un participante iniciar sesión.

Para guardar la información de las credenciales de los participantes registrados y los avances del juego de futbol de fantasía, se decidió crear un sistema de base de datos por medio de serialización. Se decidió permitir la creación de solo un equipo por participante y la existencia de solo un administrador.

Existieron complicaciones para terminar la totalidad de las funcionalidades que debíamos implementar, pues no hubo claridad a la hora de construir el código de algunos métodos principalmente relacionados con la generación de estadísticas y reportes.

Proyecto 2

Para la entrega del proyecto 2, fue necesario comenzar a modificar el diseño de clases inicialmente realizado, pues debíamos añadir un nuevo conjunto de clases correspondiente a la interfaz del programa. Para esto, analizamos qué relaciones existen entre las clases JFrame, JPanel, la interfaz ActionListener y las clases que comenzamos a modelar.

Se realizó una división de trabajo para que un integrante se encargara de construir la interfaz y los otros dos terminasen la lógica junto a los documentos solicitados en la entrega.

Para esta entrega nos enfrentamos con un nuevo reto. Diseñar e implementar una GUI. Esto era nuevo para nosotros por lo que debimos aprender las funcionalidades de Swing y cómo acoplar y desplegar todo lo que habíamos realizado, pero de manera gráfica. Del mismo modo, en esta entrega debimos estructurar y modelar nuestra aplicación para cumplir con el modelo MVC. Al realizar esto nos dimos cuenta de que en varias ocasiones estábamos mezclando la lógica de la aplicación con la vista. Esto hacía que cualquier cambio o nueva función que agregábamos a la aplicación nos hiciera realizar grandes cambios en el código.

Por otra parte, al no tener conocimiento previo de interfaces gráficas, nos enfrentamos con algunas dificultades como lo fueron el manejo de layouts en la aplicación. Por medio de esta entrega pudimos reconocer en qué aspectos debíamos mejorar para la entrega 3. Así mismo, pudimos identificar soluciones a los problemas a los que nos encontramos en la implementación del proyecto 3.

Otro aspecto que pudimos mejorar en esta entrega es que creamos clases de distintos objetos para ayudar con la persistencia. Nos dimos cuenta que de esta forma leer los archivos y crear todos los equipos de fantasía era mucho más fácil. Además, de esta forma podíamos organizar todas estas clases dentro del paquete de controlador.

Proyecto 3

En la elaboración de este proyecto, debido a que se realizaron cambios en las reglas del juego y se debían implementar funciones que permitieran visualizar reportes y progresos del juego y el desempeño de equipos y jugadores, nuevamente tuvimos que modificar el mundo del problema por medio del diagrama de clases.

Adicionalmente, debido a los problemas que enfrentamos en la implementación de la interfaz para mostrar resultados de consola, decidimos cambiar completamente el mecanismo de comunicación con los usuarios. Ahora en lugar de modelar una “consola” para la comunicación, modelamos diferentes ventanas emergentes que permitiesen de formas gráficamente distintas, registrarse, ingresar y jugar.

Debido a que no se logró dimensionar correctamente la cantidad de tiempo que iban a tomar estos cambios e implementaciones, al final no se logró dejar la aplicación en un estado final deseable. Un problema grande también fue la mala distribución del trabajo, pues se quiso dejar que una persona terminase la lógica, otra la parte de interfaz gráfica y otra los documentos y esto demostró no ser lo más eficiente.

2) ¿Qué aspectos del diseño salieron bien?

A continuación, se presenta una lista de aspectos del diseño del problema que pudieron ser implementadas de manera exitosa:

- Fue posible terminar la implementación de las funciones de registro, inicio de sesión y salir de la aplicación junto a una interfaz pertinente como parte del primer panel y primeras funcionalidades.
- Debido a que se logró el modelamiento de clases para leer y cargar archivos (en formato csv y objetos serializados), fue posible contar con persistencia de datos. Con la persistencia de datos, que es un elemento fundamental del programa, se pudieron lograr las implementaciones del punto anterior y llevar un progreso de cada juego iniciado.
- Modelar las clases de Admin y Participante junto a sus métodos diferenciadores mostrados en interfaces diferentes fue completamente posible.
- Se logró asignarle la posibilidad a la clase Admin de cargar los archivos con los datos de una temporada y luego subir un archivo de resultados para cierta fecha.
- Se pudieron construir diferentes frames y diferentes paneles que contenían las diferentes vistas necesarias de la aplicación. Sin embargo, hizo falta construir paneles para funcionalidades más relacionadas con la visualización de un equipo de fantasía y los reportes del progreso del juego.

3) ¿Qué aspectos del diseño salieron mal?

A continuación, se presenta una lista de aspectos del diseño del problema que no pudieron ser implementadas de manera exitosa:

- Algunas asignaciones de la visibilidad de los atributos y métodos se realizaron de forma incorrecta en el diagrama de clases, en el momento de la implementación pudimos notar que en realidad ciertas clases que habían sido modeladas como públicas, debían ser privadas y viceversa.
- Para la entrega de la implementación del proyecto 3, no fue posible definir las funcionalidades para generar y mostrar reportes de avances del juego de futbol de fantasía. Aunque nos imaginamos cuáles son algunas clases que se podrían modelar para generar estos reportes, al final no los incluimos en el diseño del mundo del problema.
- Debido a falta de tiempo y organización a lo largo del desarrollo de los proyectos, todas las funcionalidades no pudieron ser cubiertas y, por tanto, muchos métodos relacionados con las clases modeladas no fueron implementados.
- La dinámica del juego de futbol de fantasía en conjunto con la visualización de su progreso no pudo ser finalizada, sin embargo, con más tiempo se podría con seguridad terminar de definir los métodos necesarios y construir las interfaces gráficas que permitan la comunicación entre usuario y aplicación y visualizar los resultados.

4) ¿Qué decisiones fueron acertadas?

Algunas decisiones tomadas que resultaron acertadas para contribuir positivamente al desarrollo de la aplicación fueron:

- Haber modelado una clase interfaz llamada “Usuario” para desde allí extender los dos tipos de usuarios “Admin” y “Participante” nos permitió reciclar código y asignarle a cada clase atributos y métodos únicos que definitivamente debían quedar por separado.
- Haber separado clases “Reales” de clases “Fantasía” nos permitió desagregar algunas responsabilidades y reducir la complejidad excesiva e innecesaria de elementos como: Fechas, Jugadores, Partidos, Temporadas.
- El hecho de haber implementado el modelo MVC, nos permitió organizar de una mejor manera nuestra aplicación y posibilitar que pueda ser más fácilmente modificada en caso de ser necesario.
- Haber creado solo dos tamaños de ventana (JFrames) para luego poner los diferentes paneles ahí nos ahorró el modelamiento de más clases que extendieran de JFrame para hacer ventanas de diferentes características.
- Elegir layouts como BorderLayout en combinación con GridBagLayout resultó ser una decisión acertada para lograr diseños de GUI más estéticas y organizadas. Aunque administrar layouts dentro de otros layouts y sus características fue complejo, al final resultó en interfaces visualmente mejores.

5) ¿Qué decisiones fueron problemáticas?

Algunas decisiones tomadas que resultaron no ser acertadas para contribuir positivamente al desarrollo de la aplicación fueron:

- Al principio, haber modelado una clase para la base de datos fue una decisión que luego no pudimos implementar pues debíamos pensar mejor en una forma de generar y guardar la persistencia de datos para equipos de fantasía, fechas y jugadores de fantasía.
- Cuando comenzamos a trabajar en la entrega del proyecto 2, a la hora de construir la interfaz y querer diseñar la comunicación con el usuario por medio de inputs, nos dimos cuenta que la forma en la que estaban construidas las funciones lógicas para hacer consultas del modelo no era adecuada, pues en ellas se pedían entradas de texto por medio de la consola y estas mismas se usaban para hacer otros llamados. Sin embargo, al utilizar una interfaz, todos los inputs y outputs debían ser comunicados por medio de funciones y no era un proceso automático. La consecuencia fue tener que volver a construir todas estas funciones de conexión entre interfaz y modelo, lo cual tomó mucho tiempo y no terminamos completamente.
- No haber modelado con anticipación muchas excepciones en los métodos de las clases ocasionó que nuestro programa pueda presentar errores que terminen su ejecución

sin que esto sea informado al usuario, lo cual se termina percibiendo como bugs en la aplicación.

6) ¿Qué tipos de problemas se experimentaron durante la implementación del proyecto? ¿A qué se debieron?

El primer problema con el que nos enfrentamos en el proyecto fue con el desconocimiento de las GUI, al ser este un nuevo tema para nosotros cometimos varios errores por ejemplo en el redimensionamiento de ventanas por no tener experiencia en esto. Puede que hayan sido errores fáciles de solucionar, pero al momento no encontrábamos solución. La forma de evitar más errores fue investigando más sobre el framework de swing. Por otra parte, encontramos dificultad en el diseño de la aplicación. Al ser un programa con varias clases y responsabilidades, el agrupar estas para que cada una sea relevante a veces era difícil y más si teníamos en cuenta que debíamos separar el programa en vista, controlador y modelo. Sabíamos que luego de hacer esto realizar cualquier cambio iba a ser mucho mas fácil pero ese primer paso de diseño toma mucho tiempo y es de prueba y error. Por último, nos dimos cuenta que algunas veces nos encontrábamos con el problema de que no existían las relaciones necesarias entre clases para cumplir cierta funcionalidad. Este error se debe a que pasamos por encima dichas relaciones al hacer el análisis de dominio. Debido a esto tuvimos que volver a los primeros pasos que es entender cómo se comportan todas las clases de nuestro programa y quitar o agregar relaciones.

Consideración final

Como grupo, consideramos que la forma en la que estamos construyendo la aplicación nos puede permitir terminarla si le dedicamos más tiempo e implementamos cada método con sus respectivas excepciones y manejo de errores. Además, si quisiéramos modificarla más adelante también sería posible sin gran problema.

En general, comprendemos que no pudimos realizar un buen y completo trabajo a comparación de lo que se esperaba y pedía en esta entrega, pero confiamos en que con lo aprendido en el curso y con más práctica, podríamos terminarla con más dedicación de tiempo y distribución del trabajo.