

Documento de análisis: (Proyecto 1, Proyecto 2 y Proyecto 3)

¿Qué cosas salieron bien y qué cosas salieron mal?

Creemos que en general hubo ambigüedad para el entendimiento del enunciado, esto causó que las ideas de diseño probablemente fueran diferentes a las esperadas.

Cosas que salieron bien:

- Al tener un análisis y diseño tan exhaustivo antes de iniciar el proyecto a nivel de desarrollo, pudimos ser más eficientes y tener claridad acerca de las necesidades que debía solucionar el sistema y pudimos ser más efectivos a la hora de empezar a programar.
- Los tiempos de desarrollo del proyecto pudieron ser cumplidos sin mayor problema.
- El diseño del UML de las clases fué acorde y pudo acoplarse bastante bien a la etapa de desarrollo, por lo cual nuestro análisis y diseño del proyecto fué acorde a las necesidades que se debían implementar.

Cosas que salieron mal:

- Un impedimento o retraso que tuvimos durante el desarrollo de la aplicación fué al momento de implementar gráficos y estadísticas sobre la misma, puesto que no encontrábamos la manera de guardar los datos eficientemente y que se pudieran almacenar los registros una vez cerremos la aplicación. Por lo cual tuvimos que recurrir al uso de JSON para cargar los progresos de los usuarios y poder dar continuidad a los datos dentro de la aplicación.
- Otro contratiempo que tuvimos fue que nuestro diseño de la interfaz de la aplicación era bastante ambicioso y era extremadamente complicado de implementar en la tecnología de Java Swing (y más sin apoyo de un maquetador), por lo cual el desarrollo del mismo quedó con bastantes diferencias al modelo original realizado en el diseño.

¿Qué decisiones resultaron acertadas y qué decisiones fueron problemáticas?

Fue de gran ayuda realizar los pasos anteriores a la puesta en práctica (fase de programación), es decir, fue acertada la creación y generación del diseño de lo que se tenía como objetivo trabajar para el proyecto (analizado desde diversos ángulos como: las funcionalidades a las que tendría acceso el usuario y aquellas de estarían ocultas pero serían necesarias para el éxito de la aplicación; las historias de usuario, para saber que se pretendía tener al final; y las relaciones entre los diferentes elementos del sistema). Dado que esto permitió un mayor entendimiento y más allá de esto, ayudó a la planeación de cómo se organizaría nuestro sistema para así llegar a un resultado efectivo y muy cercano al esperado.

Sin embargo, también tuvimos decisiones que aunque fueran analizadas y estudiadas desde el principio como buenas, al final resultaron bastante problemáticas y resultaron atrasando el desarrollo del proyecto. Un ejemplo particular de una de estas decisiones fué tratar de implementar una imagen de cada jugador dentro del sistema de comprar y vender jugadores. Esto era bastante complicado porque no teníamos una base de datos que incluyera imágenes de los jugadores, aparte era difícil de insertar en el listado de jugadores. Al final esta decisión no aportaba mayor relevancia al desarrollo del sistema y su desarrollo era bastante engorroso, por lo cual la terminamos desechando.

¿Qué tipo de problemas tuvieron durante el desarrollo de los proyectos y a qué se debieron? *(En este último punto, sería conveniente discutir aspectos como los problemas con las estimaciones del trabajo necesario debido al desconocimiento de la tecnología, los problemas realizando el análisis del dominio, o la dificultad de diseñar en un entorno incierto).*

Desde un inicio se tuvieron ciertas dificultades en el planteamiento de la estructura, al momento de realizar modelo UML, ya que el enunciado donde se daba el contexto del producto final que se quería no era muy claro. Sin embargo, en clase se asumieron ciertos factores para que así todos tuviéramos la misma idea de a lo que se quiere llegar.

También, había alta complejidad en algunas partes para el desarrollo de código. Por ejemplo, fue difícil implementar el formato “JSON”, ya que no estábamos familiarizados con su forma de uso, por lo tanto, fue necesario investigar de qué manera podíamos sacar provecho de él, en cuanto a eficiencia y utilidad.

Por otro lado, teniendo en cuenta que no estábamos del todo familiarizados con el uso de las interfaces gráficas (ya que no se habían visto en anteriores cursos y el tiempo de clase no era suficiente para su total entendimiento) al momento de poner en práctica, fue trabajoso llegar al diseño que queríamos que le mostrará al usuario (en cuanto a posición de texto, el tamaño de los ítems en pantalla, etc.). Como complemento de esto, otro reto importante de mencionar, fue programar las interfaces desde 0% usando la librería de Swing, sin embargo, no se nos fué permitido usar frameworks o maquetadores visuales que hicieran el proceso de desarrollo mucho más veloz y eficiente. Esto era entendible debido a que la idea del proyecto era aprender con profundidad el uso de estas tecnologías.

En general, es coherente mencionar que al no estar en contacto anteriormente con este lenguaje Java y al ver una nueva forma de programar, es decir de designar los elementos del sistema (objetos...), fue un gran desafío desglosar cada cosa que queríamos implementar en nuestro sistema (por ejemplo, las relaciones establecidas en el modelo UML y sus elementos implícitos).