

Documento de seguimiento proyecto

Conseguir los requisitos del nivel señalado.

Justificación.

La planificación de esta entrega recoge:

- Primera mitad de una partida (primera fase completa, segunda y tercera en desarrollo)
- Módulo de estadísticas
- Funciones del administrador
- Implementación de patrones de diseño
- Depuración de código y correcciones varias

Análisis retrospectivo del sprint. Reflexión.

General

En este tercer sprint, nos hemos dedicado a avanzar en gran medida el desarrollo de una partida, creando un sistema de turnos funcional y su conexión entre el juego en directo y en la base de datos; así como completar aspectos del entorno de la aplicación como el módulo estadístico y la funcionalidad de los administradores. Además, se fue implementando paulatinamente el patrón de diseño “estados”.

Generalmente, hemos sabido organizarnos pese a la carga del resto de asignaturas y a estar en periodo de Navidades. Destacamos el tiempo empleado en solucionar errores que no comprendíamos y depurar el código existente para poder avanzar.

Con la dinámica de las nuevas implementaciones, se busca terminar de probar y desarrollar las últimas historias de usuario para la entrega final.

Daniel Díaz - Pedro Parrilla: Turnos con fases

Dado que ya teníamos implementada una primera versión de interfaz con websockets y servidor, y con la sugerencia de meter patrón estado a las fases del turno, creamos dos líneas de desarrollo conjuntas buscando poder seguir avanzar con las fases del turno al mismo tiempo que se diseñan las mismas fases con la estructura de patrón de estado.

Dentro de las diferentes fases, se diseñan las diferentes interfaces que van llamándose para cada fase del turno. Estas mandan la información de cada turno en específico y el servidor lo interpreta para registrar, procesar y devolver la petición de nueva fase al jugador. La

información de los movimientos también es interpretada para mostrar los movimientos en fichas y actualizar los diferentes tableros.

Al mismo tiempo que esto funciona dentro del canal de turnos, se añade un nuevo canal que es llamado cuando un cliente actualiza la página o entra por primera vez la partida, mande la información de la partida actual para que pueda sincronizarse al turno. Una vez queda sincronizada la partida, se cierra la conexión con este canal por el cliente. Esto formula dos canales (turn y gameload) por los que el servidor contacta con el cliente, y el mismo cliente trata de actualizar su interfaz de la misma manera.

El patrón estado se ha adaptado hasta la fase en la que se encuentra la rama, por lo que lo primero que se trabajará es implementarlo para continuar desarrollando las diferentes opciones de fase restantes y restringir el uso de los turnos de cara a la entrega final.

La versión presentada no incluye el patrón ya que no incluía todas las fases que se habían desarrollado hasta el momento. Todo debido a que tuvimos que trabajar en paralelo para no estancarnos ambos en algún error que no nos dejase avanzar, como por ejemplo que la tecnología hibernate no admite interfaces de Java como forma de tipado, no como en Java vanilla que se puede abstraer un tipo como instanciación de un objeto que implemente la interfaz.

Esto hizo que el tratamiento de las fases se hiciera en memoria y no se guardara en el servidor, provocando así diferencias con la implementación sin el patrón.

Por otro lado, las fases del juego no están ordenadas en ciertos momentos sino que quedan a elección del jugador de forma que el patrón estado no puede saber qué estado será el siguiente, esto genera ciertos problemas y diferencias con la implementación habitual del patrón de diseño las cuáles estamos trabajando para solventar de la mejor manera posible.

Álvaro Vázquez - Luis Rodríguez: Tableros, fichas (movimiento tanto en servidor como en interfaz), modificación de entidades como GameBoard y GamePiece, interacciones de monedas, panel de errores y sus respectivas pruebas.

En el inicio del sprint nos dedicamos a hacer que las fichas y tableros se visualizarán correctamente en la interfaz, la primera idea era hacer que la base de datos estuviese perfectamente conectado a la interfaz del juego, a lo largo del desarrollo, vimos que era inviable, pues el modelo que se recibe en la vista solo se carga al cargar la página. Por ello no se podía tener en cuenta en la interfaz los datos en tiempo real. Planteamos una solución y fue que por websocket se mandasen los datos, tanto en el servidor, para que se modificase la base de datos, como en la interfaz, en la que se iban a cargar los datos y modificar de manera local en el dispositivo de cada usuario. También se ha hecho limpieza en todo lo que se guardaba en la base de datos, que con la nueva estructura no hacía falta almacenar (por ejemplo las listas de posiciones X e Y de las fichas). Añadimos también las interacciones de las monedas en el juego con sus respectivas restricciones por lo que fue necesario modificar cómo funcionaban las fases. Una vez terminamos estas funciones añadimos un notificador de errores en las vistas de registro y edición de usuarios.

Gonzalo Martínez - Álvaro Miranda: Módulo de estadísticas, funciones del administrador.

Al inicio del sprint, discutimos la apariencia de la vista del módulo de estadísticas con el resto compañeros a la vez que las estadísticas a calcular; tras la aprobación de todos, Gonzalo empezó a programar la vista en jsp, mientras que Álvaro empezó con las funciones para calcular las estadísticas. Tras varios errores que se fueron solucionando, ambos conseguimos terminar con la vista del módulo pero la parte funcional no pudimos terminarla por completo por falta de tiempo, quedando así únicamente sin ser funcional el apartado de logros.

Para implementar las funciones de administración de la aplicación, modificamos la interfaz y añadimos funcionalidades: la opción de crear un usuario nuevo en la pantalla de buscar usuarios, editar un usuario accediendo a su perfil (en nuestro caso, modificar la contraseña) y la posibilidad de borrar el usuario, también desde su perfil. El borrado de usuarios nos llevó bastante tiempo y fue un quebradero de cabeza porque salían bastantes errores y no dejaba borrarlos. Tras profundizar e ir arreglando errores, tuvimos que reestructurar algunas entidades para permitir el borrado en cascada al eliminar un usuario.

Tipo	Gonzalo Martínez	Álvaro Vázquez	Luis Rodríguez	Álvaro Miranda	Daniel Díaz	Pedro Parrilla
Tableros y fichas (interfaz en funcionamiento)	10	29	17	5	15	1
Monedas	-	8	19	-	-	-
Estadísticas	25	2	-	14	-	1
Administración	6	-	3	20	-	-
Turnos	-	-	-	-	30	38
Corrección de errores	4	6	6	6	-	5
Total:	45	45	45	45	45	45

La unidad de medida son las horas de trabajo.