

INFORME

En esta última mejora del proyecto hemos cambiado bastantes cosas respecto a la estructura del proyecto, tenemos 2 paquetes principales, una para el modelo y otra para las vistas pero en esta última implementación hemos añadido 4 paquetes más en el modelo. Para la gestión de los bloques del minijuego TamaDigOut tenemos una clase por cada tipo de bloque, que luego explicaremos con el patrón factory. Respecto al tamagotchi tenemos un paquete para los elementos que se le puedes dar de alimento y otro para los distintos tipos de evoluciones, que también explicaremos posteriormente. Por último, para el minijuego del tres en raya tenemos un paquete para gestionar los jugadores y otro para las fichas del jugador y del tamagotchi.

Respecto a los paquetes principales, tenemos uno llamado “PackModelo” donde además de incluir lo previamente dicho, también se incluye la gestión de la partida, los bloques del tres en raya, los patrones, el jugador principal, la lista de los jugadores y la gestión de los minijuegos.

En el otro paquete “PackVista” tenemos todo lo relacionado con las vistas, como puede ser la pantalla principal, la de inicio o las de los minijuegos. Dentro de estas clases se encuentran las clases del controlador.

Ahora vamos a explicar los patrones que hemos utilizado en todo el proyecto, empezando por el MVC. Para toda la estructura del programa hacemos uso del Modelo Vista Controlador, este patrón consiste en separar todo en 3 partes importantes. Las clases visuales con las que el usuario interactúa son las clases del vista. Estas clases no se pueden comunicar directamente con el modelo, que son las que gestionan las cosas, por eso las vistas tienen sus controladores. por ejemplo en la pantalla de inicio tenemos el botón jugar, que llama a la clase del modelo Juego y ejecuta iniciarPartida(). Relacionado con el minijuego TamaDigOut cuando se pulsa el botón w,a,s,d el controlador manda

información al modelo, este actualiza la posición del tamagotchi y se actualiza la vista. Un último ejemplo también es a la hora de clicar un panel en el minijuego del tres en raya, el controlador manda la información al modelo, este la gestiona y actualiza el vista, por eso cuando clickamos en un panel la ficha se pone correctamente.

Para notificar a la vista los cambios del modelo utilizamos el patrón observer. Este patrón utiliza los métodos `setChanged` y `notifyObserver`. Por ejemplo en `ListaJugadores` llamamos a `notifyObserver` para enviarle las puntuaciones de los jugadores, mientras que la clase `PantallaPrincipal` recibe los datos con el `update` los coloca en la pantalla.

Para gestionar las evoluciones del tamagotchi utilizamos el patrón state, para ello el tamagotchi cuando tenga que evolucionar la clase "Tamagotchi" llama a la interfaz "Evoluciones". En el paquete de evoluciones están todas las clases a las que puede evolucionar el tamagotchi. Cada una de estas clases implementa la interfaz previamente dicha.

El patrón factory lo hemos utilizado para generar los bloques del primer minijuego, como hay distintos tipos de bloques este patrón es perfecto, al añadir un patrón como el factory es más fácil ampliar el minijuego en un futuro.

Por último hemos utilizado el patrón composite para la gestión de los alimentos del tamagotchi, es posible hacer conjuntos de comida y sopas, de esta manera el programa es mucho más fácil de ampliar y más cómodo de gestionar.