

Test Innovamat

Marc Castells

El repositorio se puede ver aquí: [GitHub - Castlemark/Innovamat_Test](https://github.com/Castlemark/Innovamat_Test)

También se ha compilado el test para web: https://castlemark.github.io/Innovamat_Test/

Características

Con tal de poder facilitar la configuración de los diversos parámetros del test, existe un `ScriptableObject` llamado `GameConfig` de configuración dentro de la carpeta de `Assets`. Desde este fichero podemos configurar el rango de números deseado y el idioma del texto. El programa soporta el rango entero de numeros de `Int32`.

Tener un fichero de configuración nos permite reutilizar el contenido del test de manera más fácil. Por ejemplo, podemos ofrecer varias dificultades creando nuevas configuraciones con valores distintos y mostrándolas como "niveles" difíciles.

El programa está también diseñado para que sea fácil añadir soporte para nuevos idiomas. Simplemente se debe crear una clase que implemente la interficie `INumberToText` y añadirla a `GameConfig` junto al nombre del idioma. Debido a las peculiaridades de cada idioma es imposible crear una sola implementación que cubra los idiomas requeridos. Esta implementación ofrece facilidad de extensión y la flexibilidad necesaria para cualquier futuro idioma. En el proyecto vienen implementados el español y el ingles.

La lógica de juego está implementada utilizando una máquina de estados. La máquina de estados es `GameController`. Esta posee varios estados como `BeginRoundState`, `CorrectAnswerState`, etc. Esta arquitectura nos permite tener la lógica separada y contenida. De este modo podemos extender y modificar el flujo del juego sin añadir complejidad.

Todas las animaciones se han hecho a traves de codigo. Se ha creado un sistema de `Tweening` muy basico. Simplemente hace falta que una clase herede de `Tweenable` y tendra acceso al metodo `Tween`. Se ha hecho esto en vez de utilizar una libreria externa porque el enunciado prohíbe explícitamente el uso de assets externos.