Ejercicio N°1

**Objetivo**: Implementar iteraciones, condicionales, estructuras de datos.

El texto que está escrito en el html se encuentra en el archivo **texto.txt** y **no requiere intervención de vue.**

Se debe inicializar desde un **archivo** llamado **js la lógica necesaria para generar la instancia de vue y montarla sobre una clase asociada a un contenedor dentro del html.**   
Ejemplo: <div class =”contenedor”>

Dentro de la **instancia root** de vue, en el **objeto data** deberán inicializar el **array de objetos** los juegos, (estos se encuentran en el archivo **juegos.txt)**

Nuestra representación en el html debe hacerse sobre una **lista** desordenada. En cuyo caso, tendremos una **iteración** sobre un list-item que deberá mostrarle al usuario la **propiedad nombre y anio** de cada juego, pero solo debe mostrar por medio de la **directiva v-if aquellos que tengan una calificación mayor a 6.** Tanto el **nombre** como el **año** se mostrarán **dentro de un h2** escrito en el html dentro del mismo list-item.  
Además deberá tener el html una etiqueta **<img />** que deberá mostrar gracias a la iteración cada una de las portadas, para esto usaremos la **directiva v-bind y afectaremos al atributo src y alt.**

Para mostrar cada una de las **consolas anidamos otro v-for sobre la etiqueta <p>** e iteraremos el array para que nos muestre cada una de ellas.

Por último, un **segundo párrafo el cual nos mostrará la calificación del juego**. Por medio de **v-bind afectaremos a su atributo class y generaremos un operador ternario en base a su calificación**, si esta es **>= 8 le daremos la clase “green”**, **caso contrario se aplicará la clase “red”** (ambas clases ya desarrolladas en la hoja de estilos.)

**No deben tener errores en la consola y debe estar cargado en un repositorio público en github.**

En la siguiente hoja queda la representación visual del navegador.

