SETUP

1. Visual Studio abierto con la begin 1 del tatetí
2. Tener en los proyectos recientes la begin 2 del tatetu, la end del tatetí, y la de tankster (poner nombres fáciles de identificar)
3. CMD abierto en la carpeta del worker role de Node.js
   1. Tener CURL.exe disponible
4. Tener el Notepad++ abierto con los .js y los .html abiertos
5. Windows Explorer abierto en la carpeta del worker role de Node.js

DEMO FLOW

1. [INTRO] Presentacion 🡪 mati
2. [DEMO] Mostrar Tankster en modo practice 🡪 mati
3. [PPT] Anatomía de un Social Game 🡪 mati
4. [PPT] High-Level de componentes de social games 🡪 mati
5. [PPT] Diagrama de componentes JS 🡪 angel
6. [DEMO] Ta-te-ti 🡪 marce/mati
   * Mostrar HTML5 canvas (breakpoint antes de dibujar)
   * Mostrar que al juego le falta avisar si ganas/empatas
   * Mostrar con TDD como implementar hasWinner() isTie()
   * Mostrar el juego funcionando con los cambios
7. [Demo] Node.js 🡪 angel
   * Levantar node.js
   * Levantar un servidor de node.js desde la consola
   * Mostrar un simple-user client
   * Transformarlo en un multi-user client
8. [Demo] Ta-te-ti 🡪 marce/mati
   * Pasar a solución intermedia del Ta-te-ti
   * Agregarle los snippets de socket.io
   * Correr node.js desde la consola
   * Correr el juego en 2 browsers
9. [Demo] Ta-te-ti en Azure 🡪 marce/mati
   * Mostrar la solución del tatetí con el worker
   * Jugar al Ta-te-ti con el browser y el iPhone
10. [Demo] Tankster en azure
    * Mostrar la solución de tankster (sin node js) 🡪 angel
    * Mostrar un blob
    * Mostrar tankster con autenticación, unirse a un game, jugar. 🡪 marce/mati
11. [PPT] Closure, links, preguntas, etc. 🡪 mati