



Constantino Calvo Mascar s

2 DAW

17/05/2023

ÍNDICE

1. Resumen del proyecto.....3

2. Lineas futuras de trabajo.....13

3. Justificación del proyecto5

4. Desarrollo del proyecto.....6

5. Conclusiones.....12

6. Bibliografía15

1. Resumen del proyecto

Temdle es una aplicación web basada en dos juegos independientes a mi proyecto: ‘wordle’ y ‘Temtem’.

[Temtem](#)¹ es un videojuego español desarrollado por Crema Games que consiste en la captura de criaturas llamadas ‘temtem’.

Temdle es un pequeño juego web que consiste en adivinar una de estas criaturas usando como pistas algunos datos clave que obtienes de intentos anteriores, como por ejemplo, el peso, la altura o los tipos de dicha criatura.

La criatura objetivo cambia cada día y hay que tratar de adivinarla en la menor cantidad de intentos posibles.

A fondo sobre Temtem...

Temtem se lanzó con acceso anticipado en el año 2020 y su versión completa se lanzó el 6 de septiembre de 2022. Es un juego multijugador masivo en línea de captura y combate de unas criaturas llamadas Temtem. Este juego es muy similar a los juegos de la franquicia Pokemon.

En su salida, Temtem, consiguió la cantidad de 21,967 jugadores simultáneos, cifra que ha ido disminuyendo con los años, hasta alcanzar picos de mil o dos mil jugadores simultáneos cuando el juego recibe nuevo contenido.

A fondo sobre Wordle...

Wordle es un juego en formato web que se popularizó en 2022. Wordle consiste en adivinar una palabra de cinco letras comenzando sin ninguna pista, y obteniendo pistas relacionadas con la última palabra que probaste, a día de hoy se puede jugar en el siguiente enlace: <https://lapalabradeldia.com/> ².

La versión original del Wordle fue creada por Josh Wardle y adquirida por el periódico New York Times tiempo después.

A raíz del éxito del Wordle, muchas versiones alternativas fueron creadas, como por ejemplo, una versión basada en [Pokemon](#) ³ o una versión sobre adivinar [películas](#) ⁴.

2. Justificación del proyecto

¿Por qué un wordle basado en Temtem?

Temtem es un videojuego el cual se vuelve más atractivo cuanto más comunidad de jugadores tenga, así que los jugadores suelen ser muy cooperativos y propensos a realizar actividades juntos.

Considero que este principio encaja a la perfección con la razón del crecimiento en popularidad del Wordle: un juego simple en el cual es muy fácil compararte con tus amigos y tratar de obtener un mejor resultado que ellos cada día.

Pensé que crear un proyecto que aunase estos dos conceptos sería muy beneficioso para la comunidad y entretenido para las personas que forman parte de ella.

¿Cuáles son los objetivos del proyecto?

Desde el comienzo del proyecto, uno de los objetivos más importantes que me llevaron a escoger esta idea fue realizar un juego en formato web completamente funcional, con el que los usuarios pudiesen entretenerse.

A nivel personal, mis objetivos son afianzar el conocimiento obtenido a lo largo del curso académico y aprender más sobre las tecnologías utilizadas y el proceso de creación de un proyecto completo de manera individual.

3. Desarrollo del proyecto

Metodologías utilizadas

Para poder organizar el proyecto correctamente desde un inicio, he tenido claro que utilizar metodologías ágiles era esencial.

Para la organización de hitos del proyecto he usado [Trello](#)⁵, un tablero Kanban que permite organizar proyectos tanto individuales como grupales.

Componentes de la aplicación

La aplicación se divide principalmente en dos partes:

La parte del front, donde se carga toda la lógica del juego y la parte del back, donde se conecta a la base de datos y cada día devuelve un objetivo diferente.

Estas dos partes, se conectan gracias a una api, la cual el front llama y el back le devuelve el identificador del objetivo de ese día en particular.

Tecnologías utilizadas

Para realizar los primeros bocetos del diseño del proyecto he utilizado [figma](#)⁶, una herramienta web que permite diseñar interfaces de software con un alto grado de profundidad.

He escogido figma frente otras opciones como [adobe XD](#)⁷ debido a que es la herramienta que se utiliza en la empresa en la que estoy realizando las prácticas y también debido a que es una herramienta enfocada tanto a diseñadores como a programadores, ya que genera código css en base al diseño creado.

En cuanto a las tecnologías principales de la aplicación he optado por: HTML, CSS, Angular y Symfony.

Tanto HTML como CSS son tecnologías esenciales para cualquier proyecto web y he optado por usar Angular y Symfony debido a que son las tecnologías estudiadas durante el curso escolar.

En cuanto a librerías de CSS, he optado por el uso de Tailwind y sus respectivos plugins por encima de Bootstrap debido a varios motivos:

Por un lado, Tailwind es mucho más ligero que Bootstrap, ya que solo carga las clases que se estén utilizando en el proyecto, en lugar de cargar toda la librería, permitiendo cargas mucho más rápidas del proyecto.

Otro motivo que me ha hecho decantarme por Tailwind es su versatilidad, su amplitud de clases permite una mayor personalización del estilo final. También he usado plugins como Tailwind elements o Tailwind scroll bar.

Además, Tailwind elements también permite hacer gráficos, los cuales estaban presentes en uno de los primeros esbozos de la aplicación web y me hicieron decantarme por tailwind.

En adición a Angular, he decidido usar la librería de Angular llamada Angular Materials, creada por Google, ya que me permite crear componentes más complejos como, por ejemplo, la barra de búsqueda o el 'dialog' que se ejecuta cuando aciertas el objetivo diario en el proyecto.

En cuanto al back, he optado por usar symfony y MySQL, debido a que son las tecnologías estudiadas durante el curso escolar y de las que más conocimiento tengo.

Sin embargo, también estuve barajando usar una base de datos no relacional, como por ejemplo MongoDB, ya que haría que la aplicación fuese más simple. Al final me decanté por la combinación de symfony y MySQL ya que, al conocer

estas tecnologías me resulta más fácil trabajar con ellas y este conocimiento me permite personalizar más el funcionamiento.

Retos encontrados

El primer reto al que me enfrenté durante la realización del proyecto fue el diseño de la página principal del mismo.

Al comenzar a plantear el diseño, decidí buscar aplicaciones similares para analizar su diseño, pero encontré que todos los diseños eran muy similares: fondos de colores sólidos y el apartado principal era muy minimalista.

Decidí que debía diferenciarme de estos diseños usando un degradado como fondo de la página para darle un toque más colorido y usar imágenes del propio juego para tratar de conseguir una identidad visual única.

Otro reto al que me he enfrentado es el diseño del juego en sí.

Al principio había diseñado la interacción con el usuario de manera muy distinta. Mientras que en el resultado final las pistas que obtiene el jugador son el peso, la altura y los tipos de la criatura, el planteamiento inicial era mostrar un gráfico con todas sus estadísticas de combate (Ataque, estamina, ataque especial, defensa, defensa especial y velocidad).

Este planteamiento viene dado a que es un juego muy enfocado en los combates, por lo que la gente suele familiarizarse más con las estadísticas de combate que con el peso o la altura de las criaturas. Sin embargo, a la hora de hacer los primeros bocetos, me dí cuenta de que mostrar toda esa información suponía varios problemas: Si trataba de mostrar toda la información de cada intento en pantalla, acababa por haber demasiada información a la vez y forzaba al usuario a tener que hacer scroll constantemente para poder verla toda.

Otra solución que se planteó fue hacer que cada intento contase con un desplegable, en el cual apareciese toda la información. Este último

planteamiento, pese a que inicialmente sonaba bastante bien, me dí cuenta de que era poco práctico, ya que para poder comprar resultados de distintos intentos habría que colapsar y expandir muchas veces los elementos y al final sería más complicado jugar.

Por estos motivos, al final decidí simplificar mucho más el juego, mostrando solamente el peso y la altura, Ya que con solamente estos datos es fácil hacer una imagen mental del objetivo y poder avanzar en el juego.

Otros dos retos que he tenido presentes desde el comienzo del proyecto han sido el aprendizaje de nuevas técnicas de programación y la gestión del tiempo, que considero que están relacionadas entre sí.

Desde el comienzo del proyecto, siempre he tenido en cuenta la gestión del tiempo y la organización del proyecto. Cuando planteé como organizarme, tuve en cuenta las tecnologías que iba a utilizar, tanto las tecnologías que ya conocía como aquellas que estaba aprendiendo. Por ejemplo, Tailwind es una tecnología que comencé a utilizar una semana antes de comenzar el proyecto, así que sabía que aprender a utilizarla iba a ser un reto ya no sólo a nivel de programación, si no también en cuanto a la gestión del tiempo. Algo similar pasó con el uso de las bases de datos no relacionales, pero en el momento del planteamiento decidí descartar esta opción ya que supondría un cambio sustancial en la manera de trabajar y no podía asegurar que me fuese a dar tiempo a aprender lo suficiente sobre esta tecnología en el tiempo que dura este proyecto.

A lo largo de los distintos proyectos que he realizado durante estos dos últimos cursos escolares he tenido que aprender de cero sobre como organizar un proyecto de programación y como estimar el tiempo que voy a tardar en realizar cada apartado del proyecto, sin embargo, gracias a lo aprendido en este tiempo, he conseguido realizar la planificación de este proyecto, que considero que ha sido la más acertada de mis proyectos recientes.

Por último, otro reto al que me he enfrentado, fue la organización del backend de la aplicación.

En resumen, el objetivo que cumple el backend de mi aplicación, es generar un objetivo distinto para cada día y devolver este objetivo al frontend de la aplicación. Por 'objetivo' se entiende la criatura que, al adivinarla, se acaba el juego y aparece la pantalla de victoria.

Una de las primeras ideas para realizar este apartado, era generar cada día un número aleatorio, el cual se vinculase con el identificador de cada criatura y devolviese una aleatoria, pero esta idea se descartó rápidamente ya que presentaba varios problemas:

- Las criaturas podían repetirse: con este modelo, existía la posibilidad de que la misma criatura apareciese dos días seguidos (o más) o con poco tiempo de diferencia, lo cual resta valor a la experiencia del usuario
- Algunas criaturas podían tardar mucho en salir: Al ser completamente aleatorio y poderse repetir criaturas, también existe la posibilidad de que algunas criaturas tarden mucho en aparecer como objetivos, lo cual hace el juego menos variado y con un tiempo de vida menor a lo esperado.

Dado estos problemas, decidí descartar esta idea y pasar a la siguiente.

La siguiente idea que tuve, ya era bastante similar a la idea definitiva que ha sido implementada, pero también presentaba problemas.

Este modelo consistía en crear una base de datos en la que, en lugar de ser completamente aleatorio, seguiría siendo aleatorio pero con ciertas condiciones, como por ejemplo, que no se pueda repetir el mismo número hasta que hayan salido el resto.

Pese a que esta idea iba mejor encaminada que la anterior, me presentaba un problema a nivel técnico:

- Necesitaba alguna manera de ejecutar una función de mi código cada 24 horas, para cambiar el objetivo del día.

Pese a que esto se pueda hacer, todas las opciones que barajé presentaban distintos problemas, como por ejemplo, utilizar cron, que es una herramienta que nunca había utilizado y no me acababa de convencer, ya que no quería depender del sistema operativo de la máquina en la que estuviese ejecutando la aplicación. Así que después de valorar estas opciones, se me ocurrió la manera de gestionar este problema que acabó siendo definitiva.

Para la versión definitiva, mantuve la regla de aleatoriedad de la anterior, haciendo que los números no se pudieran repetir, pero que los identificadores de las criaturas estuviesen desordenados.

A esta tabla desordenada, se añade otro campo, llamado fecha, el cuál guarda la fecha de los días, de manera que asigna un día a cada criatura. De esta manera, no es necesario ejecutar un código cada 24 horas, ya que cuando la fecha cambie, el backend de la aplicación, gracias a la función `DateTime()` es capaz de sacar la fecha del día de hoy y compararla con la base de datos, dando así el cambio por cada día.

Esta solución también presenta otros problemas, como por ejemplo, que solo he generado la misma cantidad de días como criaturas hay, por lo que hay que volver a generar la base de datos cuando estos se acaben, pero ya tengo contemplado como abordar este problema.

4. Conclusiones

A nivel profesional

De lo que más orgulloso me siento de la realización de este proyecto es de mi estimación y organización del tiempo, tal y como he mencionado en una de las páginas anteriores.

Ser capaz de estimar cuanto tiempo va a ocupar cada parte del proyecto me ha permitido establecer prioridades y poder realizar un mejor producto.

Por otro lado, este proyecto me ha servido para afianzar todo el conocimiento aprendido durante este último curso escolar, ya que, pese a que había adquirido muchos conocimientos, hasta que no los he puesto en práctica todos juntos en la realización de este proyecto no he sido consciente de todo lo que había aprendido.

A nivel personal

Trabajar en este proyecto, personalmente, me ha hecho mucha ilusión, ya que es un proyecto que quería llevar a cabo desde hace bastante tiempo, así que poder trabajar en él ha sido gratificante.

Por otro lado, compaginar este proyecto con las prácticas ha sido un reto al que he tenido que enfrentarme.

Las prácticas han ocupado una gran parte del tiempo del que disponía para realizar el proyecto, Pero al final he sido capaz de organizarme teniendo las prácticas en cuenta.

5. Lineas futuras de trabajo

Debido a que me ha sobrado algo de tiempo respecto al plazo del proyecto, he preparado algunas funcionalidades o mejoras de cara a futuro:

- Tutorial: Al igual que en otras aplicaciones basadas en wordle o incluso el propio wordle, una de las ideas de cara a futuro es añadir un tutorial que explique tanto el funcionamiento del juego, como los efectos visuales que indican los aciertos o los fallos.

Este tutorial será accesible desde el header de la aplicación, justo debajo del logo.

- Reajuste de la dificultad del juego: Una de las diferencias principales entre este proyecto y el resto de trabajos que he realizado durante el curso escolar, es la interacción que tiene el usuario con la aplicación. Mientras que en los anteriores proyectos que había realizado la interacción del usuario con la aplicación consistía en navegar u obtener información, en este proyecto consiste en jugar. Por ende, durante el proyecto, he tenido que diseñar la jugabilidad del mismo.

Sin embargo, durante el proceso final del proyecto, cuando ya era completamente jugable, me he dado cuenta de ciertas mejoras que podrían perfeccionar la interacción con el usuario:

- Informar cuando se han acertado todos los tipos. En temtem, las criaturas pueden tener un tipo o dos tipos, por ende, cuando el usuario adivina dos tipos, es consciente de que ha acertado todos los tipos posibles, sin embargo, si el objetivo solo dispone de un tipo, el usuario puede pensar que no es así, y seguir probando tipos cuando ya no es necesario.
- Mostrar el peso y la altura de cada criatura en la barra de búsqueda. Como he mencionado en los retos afrontados, los usuarios jugadores de Temtem, suelen relacionar más las habilidades de combate con las criaturas que su altura y su peso,

por eso, creo que incluir el peso y la altura de cada uno en la barra de búsqueda, antes de probarlo, ayudaría a recordar estos valores sin tener que hacer pruebas innecesarias.

- Otro punto que ha sido valorado desde el principio es añadir un límite máximo de intentos, al igual que tiene el wordle original, pero no lo considero la mejor idea de diseño. Mientras que un límite añade un reto al juego, considero que Temdle ya es suficientemente complicado, a demás, añadir un límite de intentos, tiene como objetivo que el usuario intente adivinar la criatura en el menor número de intentos posible, pero las bases del propio juego ya llevan al usuario a intentar jugar de esta manera, por lo que creo que a efectos prácticos un límite solo serviría como 'corte' a mitad de una partida de un usuario que está disfrutando de la aplicación.
- Otro plan a futuro que tengo pensado añadir, es la generación automática de la base de datos. Debido a la manera en la que funciona ahora mismo la base de datos, existe una función de php que genera 164 días de objetivos aleatorios, pero ejecutar esta función es manual. Mi plan de implementación es comprobar si el día en el que se ejecuta la aplicación es el último día registrado en la base de datos, y en ese caso, volver a generar las tablas, ya he preparado parte del código para implementar esta funcionalidad.

6. Bibliografía

1. Página oficial de Temtem: <https://crema.gg/games/temtem/>
2. Página oficial de Wordle en Español: <https://lapalabradeldia.com/>
3. Página de Wordle basado en Pokemon:
<https://squirdle.fireblend.com/daily.html>
4. Página de Wordle basado en películas: <https://framed.wtf/>
5. Kanban Trello: <https://trello.com/>
6. Figma: <https://www.figma.com/>
7. Adobe XD: <https://helpx.adobe.com/es/xd/get-started.html>