

MEMORIA PRÁCTICA 1 - INGENIERÍA DEL CONOCIMIENTO

DETALLES DE IMPLEMENTACIÓN

Se trata de una aplicación **Electron**, ***(los motivos son los mismo que tenía en la práctica anterior) elegí esta tecnología ya que una forma sencilla podía desarrollar aplicaciones multiplataformas de manera sencilla. Otro factor que tuve en cuenta para desarrollar una aplicación multiplataforma es muchas de las aplicaciones más populares son aplicaciones **Electron**, esto me daba varias ventajas significativas, como una documentación de calidad y una amplia comunidad que responde a las preguntas frecuentes del desarrollo.

Una de las ventajas que da **Electron** es que ejecuta **NODEJS** que para mí es una ventaja significa ya que esto significa que se van a poder utilizar tecnologías web dentro de Electron, algo que en este momento prefiero ya que desarrollar con JAVA me cuesta.

Ya que usamos **NODEJS** y todo Electron está basado en tecnologías WEB, los lenguajes de programación que use han sido: **JAVASCRIPT, HTML, CCS**.

Para manejar los paquetes y dependencias en proyectos Node lo normal hasta ahora es usar **NPM**, pero en su lugar yo elegí **YARN** ya está destinado a convertirse en el nuevo estándar para manejar paquetes y dependencias, además que la velocidad de ejecución es menor en comparación con npm.

Durante los últimos años estamos viendo el auge de las tecnologías web, donde podemos encontrar muchos frameworks que nos ayudan a que nuestras aplicaciones se vean bien.

Elegí **React Native** como framework para la aplicación debido a lo fácil que es implementar componentes.

En resumen, esta es una aplicación **Electron**, que dentro corre un proceso **NodeJs** que nos permite ejecutar el framework **React Native** qué es el que nos muestra toda la parte visual y el que se encarga de ejecutar la lógica del modelo.

Procedimiento

Para poder desarrollar la práctica el primer paso fue entender qué es y cómo funciona el algoritmo ID3. Para ello me fue muy útil el documento (Curso-weka) adjunto en la documentación de la práctica

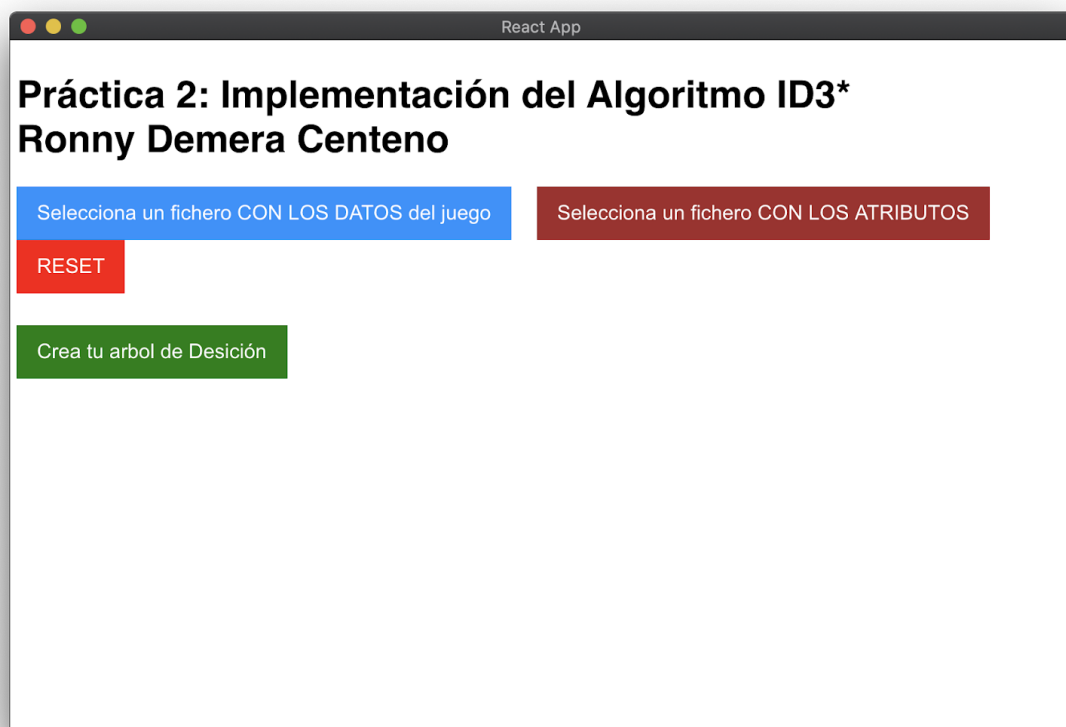
Gracias a las estructuras de datos que proporciona javascript **Javascript** he implementado el algoritmo con **Arrays, Maps y colas de prioridad**. Me gustaría destacar la importancia de los Maps, ya que facilitan el manejo de datos.

MANUAL DE USUARIO

Se adjunta una aplicación ejecutable en sistemas Windows (se puede proporcionar el ejecutable para sistemas Mac OS si hiciera falta).

Descomprimos la carpeta **win-unpacked**, dentro se encuentra el ejecutable **practica2.exe** aceptamos ejecutar aplicaciones desconocidas (se muestra esto porque en la build no hay posibilidad de firmar, ya que está no es una aplicación oficial).

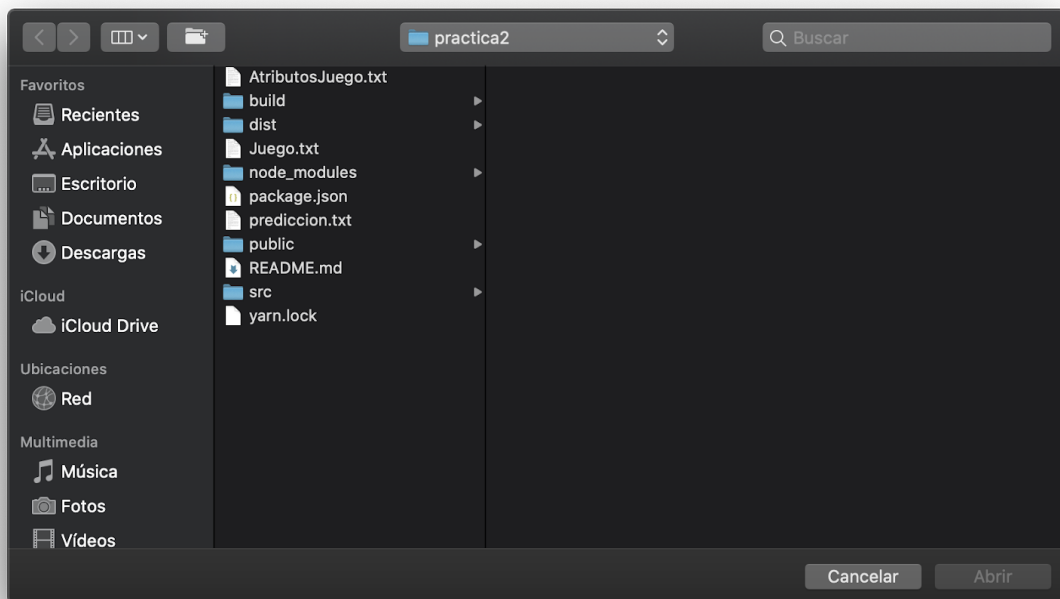
La pantalla que veremos al iniciar la aplicación es esta:



Dispones de varios botones, que nos ayudarán a mostrar nuestro árbol.

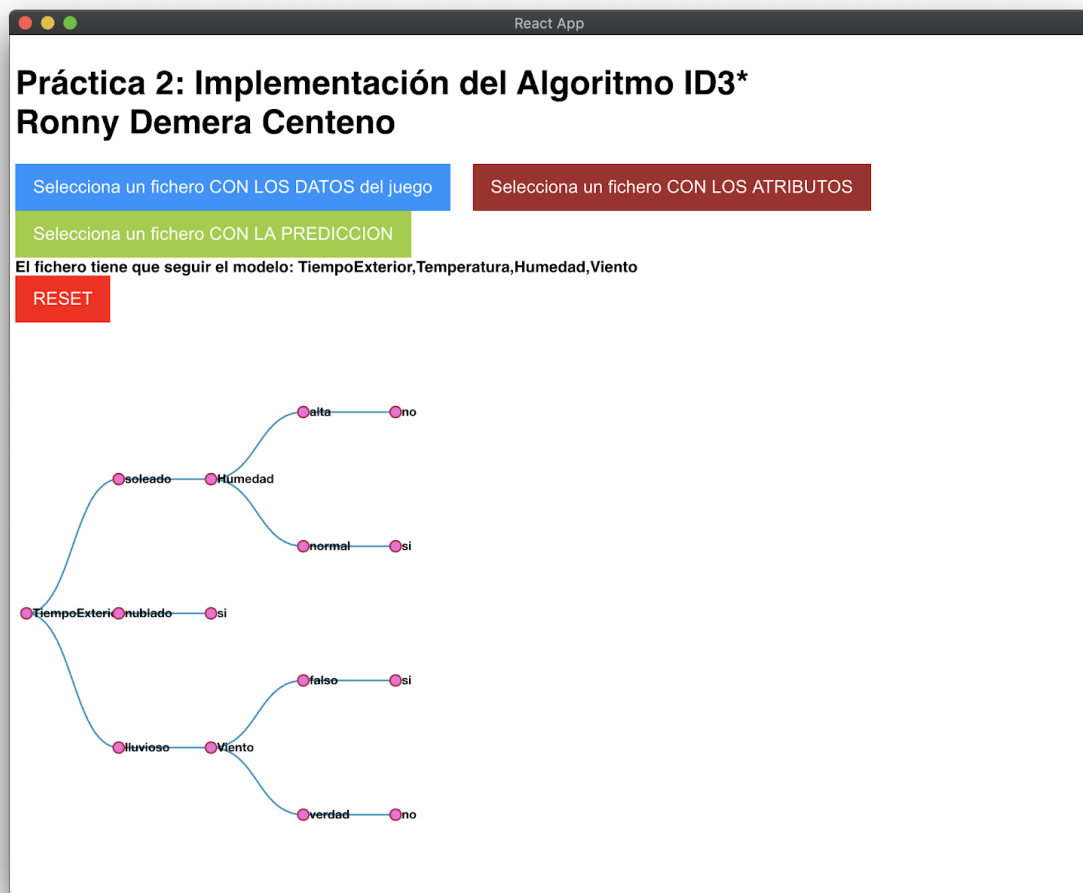
- El primer botón (AZUL) lo pulsaremos para seleccionar el fichero correspondiente con los datos del juego. En el adjunto el archivo que se

asocia a este Botón era el llamado **Juego.txt**. Cuando pulsamos nos aparece una ventana donde tenemos que elegir el archivos que queramos.



- El segundo botón (MARRÓN) , tiene el mismo funcionamiento abrirá una nueva ventana donde tendremos que escoger el archivo con los atributos (**AtributosJuegos.txt**).
- Tercer botón **RESET** (ROJO) con el podremos reiniciar el estado del árbol, útil por si nos hemos equivocado seleccionando algún archivo.
- El último botón lo seleccionaremos cuando ya hayamos seleccionado los dos primeros. **Crear el árbol.** *** Es importante que hayamos seleccionado este botón después de seleccionar los dos primeros ficheros.

Cuando ya tenemos nuestro árbol creado aparecerá un nuevo botón.



Gracias al nuevo botón podremos **comprobar el correcto funcionamiento del algoritmo para los ejemplos de la tabla**. Para ello seleccionamos el nuevo botón, nos vuelve a aparecer la ventana para seleccionar el fichero.txt, ***debajo del botón vemos el formato que tiene que tener el fichero.

Una vez seleccionado debajo del árbol nos aparece el resultado final.

