



# AGRADECIMIENTOS

Querría agradecer a todas las personas que me han ayudado y apoyado durante estos años de carrera, especialmente a mis padres por siempre ofrecerme ayuda.

También querría agradecer a mis amigos, pues sin ellos esto se habría hecho mucho más largo y aburrido, y finalmente a mi tutor de este trabajo, Salvador Alcaraz Carrasco, por ayudarme y darme la idea de realizar este proyecto.



# ABSTRACT

Gamification is a term often used in the academic field in order to describe the process of increasing the engagement of certain activities that otherwise would be more bothersome, so as to make the process of completing said activities more pleasant.

This is the primary motivator in the construction of this application, the gamification of the “capture the flag” game, applied inside a schoolroom.



# RESUMEN

Gamificación es un término que se usa en el ámbito académico para describir el proceso de incrementar la participación de ciertas actividades que de otra manera se harían muy tediosas, para que el proceso de completarlas sea más ameno.

Esta ha sido la principal motivación para construir esta aplicación, la gamificación del juego “capturar la bandera”, aplicado a un aula de enseñanza.



# PALABRAS CLAVE

Gamificación

Front-End

Back-End

Full-Stack

Mobile First

Capturar la bandera

Similitud Coseno



# Índice de contenidos

AGRADECIMIENTOS .....	2
ABSTRACT .....	3
RESUMEN .....	4
PALABRAS CLAVE .....	5
Tabla de ilustraciones: .....	8
1. INTRODUCCIÓN .....	10
1.1 APLICACIÓN “CAPTURAR LA BANDERA” .....	10
1.1.1 FUNCIONAMIENTO DE LA APLICACIÓN .....	10
1.2 TECNOLOGÍAS UTILIZADAS .....	14
1.2.1 FRONT-END .....	14
1.2.2 LOCALSTORAGE Y WEBSOCKET .....	15
1.3 DECISIONES DE DISEÑO .....	15
1.3.1 APLICACIÓN FULL-STACK VS APLICACIÓN FRONT-END .....	15
1.3.2 FRAMEWORK FRONT-END.....	16
1.3.3 MATERIAL DESIGN VS NG ZORRO .....	17
1.3.4 DISEÑO MOBILE-FIRST .....	17
1.3.5 FICHERO DE RETROALIMENTACIÓN SOLO AL ALUMNADO .....	18
1.3.6 SIMILITUD COSENO VS IA .....	18
2 ACTORES.....	19
2.1 PROFESORES .....	19
2.2 ESTUDIANTES.....	28
2.3 ADMINISTRADORES.....	35
3. PUESTA EN MARCHA .....	35
3.1 INSTALAR NODEJS .....	36
3.2 DESCARGAR FICHEROS .....	41
3.3 INSTALAR ANGULAR, ANGULAR MATERIAL Y EJECUTAR APLICACIÓN.....	43
3.3.1 WINDOWS:.....	43
3.3.2 MACOS:.....	43
3.3.3 LINUX. ....	44
4. EVALUACIÓN DE LA PRUEBA.....	47
4.1 EVALUACIÓN DE LAS PREGUNTAS DE DESARROLLO.....	48
4.2 IMPORTANCIA DE CADA APARTADO .....	49
4.2.1 PUNTUACIÓN DIRECTA .....	50
4.2.2 PUNTUACIÓN NORMALIZADA .....	50
4.2.3 PUNTUACIÓN A TRAVÉS DE CURVA DE APRENDIZAJE.....	50

4.2.4 PUNTUACIÓN RASCH.....	50
4.2.5 DECISIÓN FINAL.....	51
5. CONCLUSIONES .....	51
6 ANEXO I .....	53
7 BIBLIOGRAFÍA.....	54



# Tabla de ilustraciones:

Pantalla de Inicio (Figura 1).....	11
Pantalla creación prueba (Figura 2) .....	12
Plantilla preguntas opción múltiple (Figura 3).....	20
Plantilla preguntas desarrollo (Figura 4).....	21
Plantilla error plantilla hoja de cálculo (Figura 5) .....	22
Indicar número preguntas y tiempo prueba (Figura 6).....	23
Plantilla prueba creada (Figura 7) .....	24
Pantalla errores XML (Figura 8).....	25
Pantalla añadir pistas y puntuación preguntas (Figura 9).....	26
Pantalla progreso prueba 1 (Figura 10).....	27
Pantalla progreso prueba 2 (Figura 11).....	28
Pantalla introducir nombre y código prueba (Figura 12) .....	29
Pantalla resolver prueba (Figura 13).....	30
Pantalla resolver prueba, no avance sin respuesta (Figura 14).....	31
Pantalla resolver prueba, mostrar pista (Figura 15) .....	32
Pantalla prueba finalizada y descarga fichero resultados (Figura 16).....	34
Fichero resultados (Figura 17) .....	35
Pantalla descargar NodeJS (Figura 18) .....	36
Pantalla Instalación NodeJS (Figura 19).....	37
Pantalla acuerdo licencia NodeJS (Figura 20) .....	38
Pantalla ruta instalación (Figura 21) .....	39
Pantalla añadir NodeJS a PATH (Figura 22) .....	40
Opciones extra de instalación (Figura 23).....	40
Pantalla instalar (Figura 24) .....	41
Descargar ficheros de GitHub (Figura 25).....	42

Ficheros descargados (Figura 26) .....	42
Acceder al CMD de Windows (Figura 27) .....	43
Comando instalación Angular (Figura 28).....	44
Ejecución proyecto (Figura 29).....	45
Función de conexión WebSocket (Figura 30).....	45
Comando ipconfig (Figura 31).....	46
Comando ifconfig en Linux (Figura 32) .....	46
Comando ifconfig en MacOS (Figura 33).....	47
Código funcionamiento similitud coseno (Figura 34).....	49



# 1. INTRODUCCIÓN

En este capítulo se explicará en qué consiste la aplicación desarrollada, así como el proceso que se ha seguido para elegir las tecnologías utilizadas y las decisiones de diseño que se han tomado.

Nótese que los números entre paréntesis que aparecen a lo largo del presente documento hacen referencia a los elementos de la bibliografía del número en cuestión. Por ejemplo (1) hace referencia al elemento número 1 de la bibliografía.

## 1.1 APLICACIÓN “CAPTURAR LA BANDERA”

La aplicación se trata de una reinterpretación digital del juego “Capturar la bandera”.

Originalmente, el juego consta de dos o más equipos que compiten entre sí para capturar la bandera del rival o una bandera neutral.

En nuestro caso, en vez de tener equipos enfrentados, todos los participantes compiten entre sí para conseguir llegar a la “bandera” con la mejor puntuación posible.

Es así el objetivo de esta aplicación, la *gamificación* de un juego clásico para fomentar el aprendizaje de los estudiantes (1), utilizando como apoyo la competición entre personas de manera sana (2).

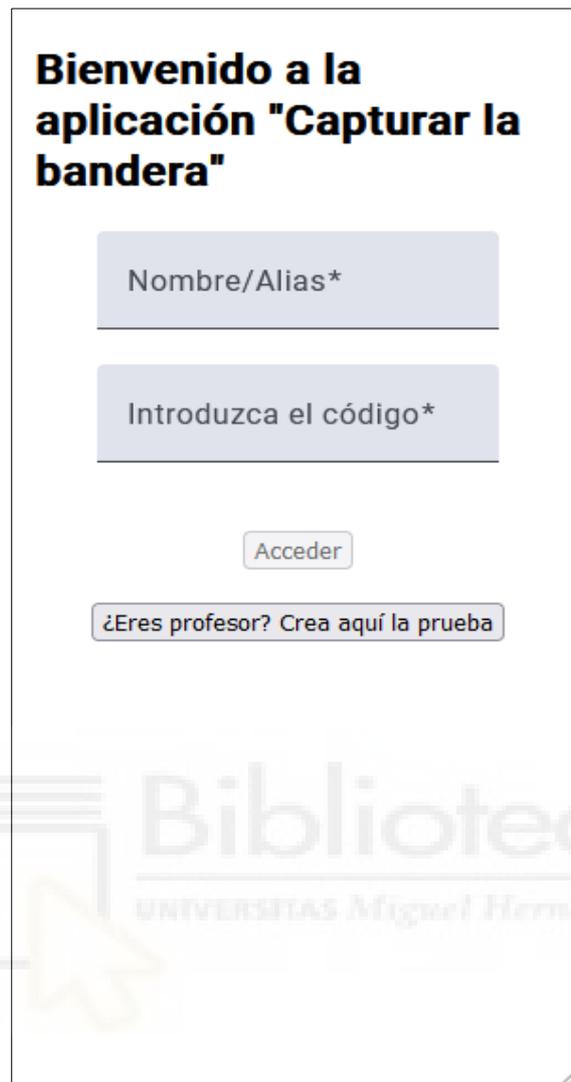
Es importante no abusar en exceso de la competitividad entre estudiantes, pues puede tener efectos negativos sobre ellos, especialmente a nivel psicológico (3)(4).

Cabe destacar que este programa es una herramienta, y como toda herramienta, dependiendo del uso que se le dé, se obtendrán unos resultados más o menos favorables. Se deja a discreción del profesor que quiera utilizarla.

### 1.1.1 FUNCIONAMIENTO DE LA APLICACIÓN

A continuación, se explicará el funcionamiento de la aplicación, desde la creación de la prueba, hasta que todos los estudiantes hayan terminado, o el profesor decida terminarla, pasando por cómo los estudiantes afrontarán dicha prueba.

Un usuario accederá a la aplicación, y se encontrará con la siguiente pantalla:



**Bienvenido a la  
aplicación "Capturar la  
bandera"**

Nombre/Alias\*

Introduzca el código\*

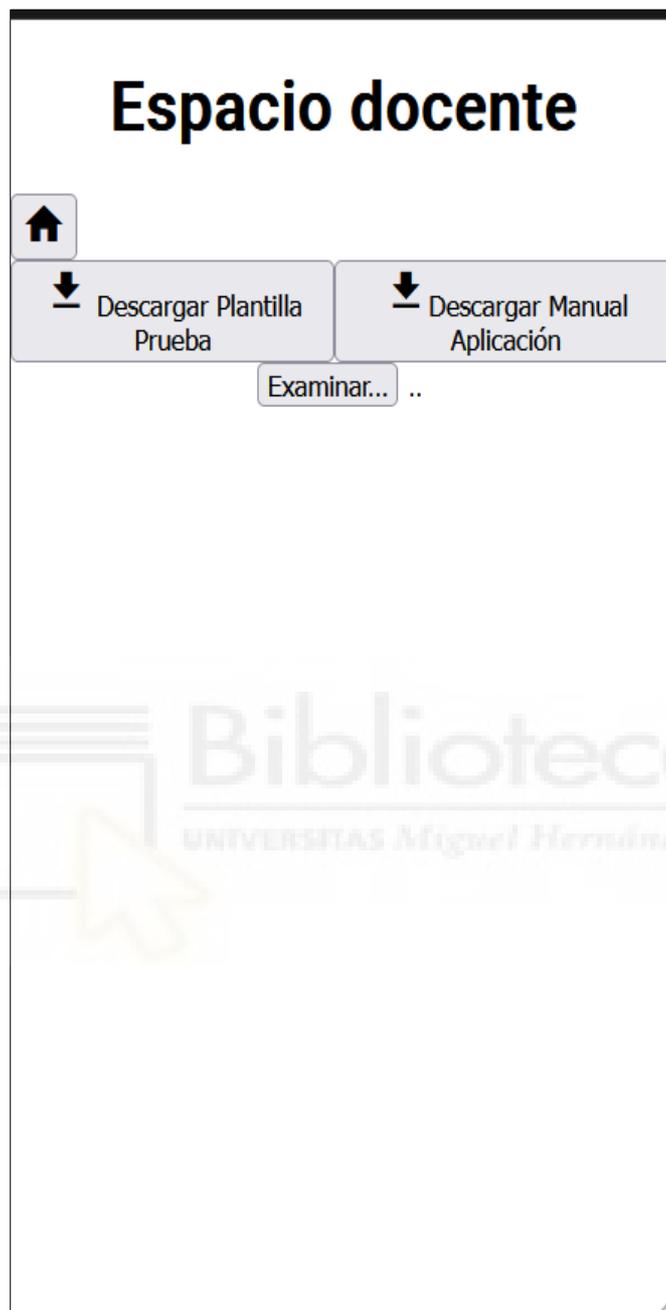
Acceder

¿Eres profesor? Crea aquí la prueba

Biblioteca  
UNIVERSITAS Miguel Hernandez

*Pantalla de Inicio (Figura 1)*

En el caso en que el usuario sea un profesor, hará click en el botón para crear la prueba, lo que le llevará a la siguiente pantalla:



*Pantalla creación prueba (Figura 2)*

En esta pantalla se le presentan los siguientes elementos:

- **Descargar Plantilla Prueba:** Se le proporcionará al profesor una plantilla del fichero de Hoja de Cálculo para que lo rellene con las preguntas que desee.

Esta plantilla consta de dos hojas, una para las preguntas de opción múltiple y otra para las preguntas de desarrollo.

- **Descargar Manual Aplicación:** Manual en el que se explica al profesor cómo funciona la aplicación (cómo debe rellenar la plantilla, cómo se crea una prueba, cómo acceden los estudiantes a ella, cómo obtener los resultados, etc.).
- **Examinar:** Elemento que le permite al profesor seleccionar el fichero a partir del que se creará la prueba.

Una vez el profesor haya seleccionado un archivo y se compruebe que el contenido del mismo sea correcto para la creación de la prueba, aparecerá un elemento que le permitirá al profesor indicar cuántas preguntas quiere incluir en la prueba. Deberá introducir un valor mayor o igual al número de preguntas extraídas del fichero. En caso que elija un valor mayor al número de preguntas, la prueba incluirá todas las preguntas extraídas. En caso contrario, se elegirán aleatoriamente entre las preguntas de elección múltiple y las de desarrollo.

Una vez seleccionado el número de preguntas, se creará la prueba y se generará un código para acceder a ella, el profesor compartirá dicho código con los estudiantes para que estos accedan a la prueba.

Cuando los estudiantes introduzcan el código e indiquen su nombre o alias, accederán a la pantalla para resolver la prueba.

En esta pantalla, se presentarán de una en una las preguntas de las que consta la prueba.

Para avanzar en la prueba, deberán contestar a las preguntas en la que se encuentren. No podrán avanzar a la siguiente pregunta sin haber contestado la pregunta en la que estén.

Una vez terminen, se les presentará en pantalla donde podrá descargar un fichero con una serie de datos relacionados con el intento que han realizado:

- **Número de aciertos:** el número de preguntas que se han contestado de manera correcta.
- **Número de fallos:** el número de preguntas que se han contestado de manera incorrecta

En el caso de las preguntas de desarrollo, se contarán como aciertos si superan un cierto porcentaje de coincidencia con la respuesta establecida por el profesor.

Las preguntas de múltiple elección sólo contarán como aciertos en el caso en que se seleccionen todas las opciones correctas, en caso contrario contarán como fallos.

- **Pistas utilizadas:** cuando el usuario decida recurrir a una pista para obtener más información respecto a la pregunta actual, se le restará una cantidad de puntos determinada, que por defecto es diez puntos.
- **Tiempo del intento:** el tiempo que ha tardado el estudiante en terminar la prueba. Puede ser mayor o inferior al tiempo “límite” que ha puesto el profesor, y esto afectará a la puntuación final de manera positiva o negativa respectivamente.
- **Puntuación final:** puntuación que ha obtenido el estudiante en la prueba. Se calcula mediante los aciertos, los fallos, el tiempo que ha tardado el usuario en finalizar la prueba y si el usuario ha recurrido a pistas o no.
- **Porcentaje de aciertos:** porcentaje de aciertos que ha tenido el usuario. Hay que tener en cuenta que, aunque la puntuación final y el porcentaje de aciertos están relacionados, debido al factor de las pistas, dos usuarios que tengan el mismo porcentaje de aciertos pueden tener puntuaciones muy diferentes.

Se explicará en más detalle el funcionamiento de la aplicación en el apartado de Actores.

## 1.2 TECNOLOGÍAS UTILIZADAS

A continuación, se realizará una explicación sobre las tecnologías utilizadas para crear la aplicación, así como el razonamiento para su elección frente a la de otras opciones disponibles.

### 1.2.1 FRONT-END

Para la parte de front-end de este proyecto, se ha decidido utilizar la combinación de Angular y TypeScript, junto con la librería de componentes Material Design.

Esto permite una mejor integración y conexión entre todas las tecnologías, puesto que están creadas por la misma entidad, en este caso Google, aportando también cierta tranquilidad a futuro, pues son tecnologías que siguen recibiendo constantes mejoras y actualizaciones.

## 1.2.2 LOCALSTORAGE Y WEBSOCKET

Esta aplicación no utiliza un back-end ni una API ni una base de datos para almacenamiento de los datos de las pruebas y los intentos, todo se hace en el front-end.

El almacenamiento se soluciona mediante el uso de LocalStorage para guardar la información relativa a las pruebas y a los intentos de resolución de las mismas.

Para que todos los usuarios puedan acceder a esa información, se ha utilizado WebSocket para establecer un canal bidireccional de comunicación, en el que el profesor que haya creado la prueba funcionará como host, y los alumnos se conectarán a esa sesión.

## 1.3 DECISIONES DE DISEÑO

En este apartado se profundizará en qué decisiones de diseño se han tomado a nivel de aplicación a lo largo de su desarrollo.

### 1.3.1 APLICACIÓN FULL-STACK VS APLICACIÓN FRONT-END

La primera pregunta que se planteó después de tomar la decisión que la aplicación sería una aplicación web fue si esta sería una aplicación sólo front-end o si se trataría de una aplicación full-stack.

Los beneficios de una aplicación sólo front-end son:

- **Desarrollo más rápido y sencillo:** al utilizar simplemente typescript para toda la lógica, y encontrarse todos los datos ya en el front-end, simplificará la comunicación entre las diferentes partes de la aplicación, pues se elimina la necesidad de conectarnos a una API para acceder a los recursos del back-end.

Esto también significa quitar la necesidad de tener una API y un back-end, pues todo el procesamiento se haría en el lado del cliente.

- **Facilidad de despliegue:** al tratarse solamente de un proyecto web, simplifica la puesta en marcha de la aplicación a través de cualquier servicio de *hosting* que se podría elegir (Amazon Web Service, Google Cloud, Microsoft Azure, etc.) sin tener que preocuparnos de posibles problemas, como por ejemplo que el tipo de base de datos utilizado no esté soportado de manera nativa en ese servicio (Como pasa con MongoDB y Amazon Web Services, este último no tiene integración nativa para el primero, teniendo que recurrir a alternativas como DynamoDB o

DocumentDB) o que ciertas librerías de nuestro back-end no sean compatibles con el sistema operativo del servidor de *hosting*.

- **Independencia respecto a Internet:** esta aplicación está pensada para que se ejecute en local y los diferentes usuarios accedan, a través de red de área local, al recurso necesario para resolver la prueba, lo que elimina la necesidad de una conexión a Internet constante.

Por el contrario, aparecen los siguientes inconvenientes:

- **Dependencia de APIs externas:** Se precisaría de utilizar APIs de terceros para obtener datos y/o funcionalidad, lo que limita la flexibilidad y el control que se podría tener sobre la aplicación.
- **Seguridad:** No será posible ocultar toda la lógica que se utiliza, lo que puede llevar a problemas con la seguridad, ya que las vulnerabilidades serán más accesibles para posibles atacantes.
- **Rendimiento y escalabilidad limitados:** Relacionado con la necesidad de utilizar APIs de terceros, si se requiere del uso de muchos servicios diferentes, la experiencia del usuario en nuestra aplicación puede verse mermada.
- **Prescindir de una base de datos:** Este es el principal problema con una aplicación sólo front-end, la persistencia de los datos. Las opciones actuales para la persistencia son *LocalStorage* y *SessionStorage*, las cuales están bastante limitadas a la hora de almacenar datos.

Teniendo en cuenta los puntos positivos y negativos, y el ámbito relativo a la aplicación, se ha tomado la decisión de desarrollar una aplicación sólo front-end, ya que se ha priorizado un fácil despliegue y ejecución de la aplicación sobre funcionalidades extra y persistencia de datos.

### 1.3.2 FRAMEWORK FRONT-END

Una vez decidido que se trataría de una aplicación sólo front-end, el siguiente paso fue elegir el framework que se utilizaría para el desarrollo.

Después de analizar (5) los diferentes frameworks y lo que ofrece cada uno, se tomó la decisión de optar por Angular 18 (6), el cual se ha convertido en la versión por defecto para todos los nuevos proyectos que se realicen con este framework.

Algunas de las mejoras que trae esta nueva versión son las siguientes:

- Material 3 y control de flujo nativo ahora son estables y presentan mejoras.
- Rendimiento general de las aplicaciones.

Otro motivo para haber elegido Angular es, que es el framework con el que el programador tiene más experiencia.

### 1.3.3 MATERIAL DESIGN VS NG ZORRO

En las etapas de decisión de tecnologías, se planteó hacer uso de la librería de componentes NG Zorro, en vez de usar Material Design. Finalmente, se optó por esta última debido a los siguientes puntos:

- **Consistencia visual:** Material Design proporciona un sistema de diseño consistente y estético, reconocido por su uso tanto en aplicaciones de Google como de terceros.
- **Migración a otros entornos:** Material Design tiene soporte nativo para Android, Jetpack Compose, Flutter y Web, lo cual facilitaría el trabajo que llevaría migrar/replicar una aplicación a cualquiera de las opciones que ofrece.
- **Integración con Angular:** Dado que ambas tecnologías están desarrolladas por Google, la integración es perfecta, lo que dota de una experiencia más fluida a la hora de desarrollar.
- **Documentación y facilidad de desarrollo:** Material Design es una de las librerías de componentes más utilizadas, lo que significa que hay una gran cantidad de información (proyectos creados por otras personas, entradas en StackOverflow, discusiones en foros) la cual facilita el desarrollo de la aplicación.

### 1.3.4 DISEÑO MOBILE-FIRST

Desde su introducción en el 2011(7), Google ha estado dándole cada vez más importancia a que las aplicaciones web se desarrollen pensando que el usuario principal va a utilizar, en primera instancia, un teléfono móvil para interactuar con la aplicación (8), llegando a redactar una serie de pautas, o “buenas prácticas” para un correcto desarrollo mobile-first (9).

Recientemente, algunos medios de comunicación han informado que a partir del 5 de julio de 2024 Google iba a implementar medidas para dar prioridad al contenido que tenga buena visibilidad en móvil (10).

Dado que es el actual estándar para el desarrollo de aplicaciones web, así como las noticias más recientes respecto a cambios por parte de Google, se ha decidido desarrollar la aplicación principalmente para su uso a través de teléfono móvil.

Otro factor importante para esta decisión es que, esta aplicación está pensada para que se use en un ámbito educativo, y no siempre va a haber infraestructura suficiente para proporcionar a todos los estudiantes un equipo único. Sin embargo, según datos del Instituto Nacional de Estadística, el 68'7% de las personas (entre 10 y 15 años) tienen un smartphone (11). Este porcentaje aumenta conforme se aumenta la edad de los sujetos. Al diseñar la aplicación para que se use desde un teléfono móvil, se facilita el acceso a la misma para el mayor número de usuarios posible.

### 1.3.5 FICHERO DE RETROALIMENTACIÓN SOLO AL ALUMNADO

Cuando un estudiante termina la prueba, se le facilita la descarga de un fichero con la información de su intento. El profesor no tiene acceso a este fichero a menos que el alumno se lo facilite.

El motivo para esta decisión es “obligar” a que los profesores interactúen con sus estudiantes, ya sea mediante una corrección de la prueba en voz alta, lo que permite a los estudiantes preguntar las dudas que les hayan surgido en el transcurso de la misma, o mediante alguna otra manera de proporcionar una retroalimentación más exacta que la que se proporciona en la aplicación.

Los estudiantes, por regla general, funcionan mejor cuando disponen de una retroalimentación constante (24)

### 1.3.6 SIMILITUD COSENO VS IA

Otra de las características principales de la aplicación es la inclusión de la posibilidad de uso de preguntas de desarrollo a la hora de crear una prueba. Para ello, se encontraron dos opciones en la etapa de diseño, el utilizar la librería `fuzzyset.js`, la cual permite acceder a métodos para el cálculo de la similitud coseno, que permiten averiguar el índice de coincidencia entre dos cadenas, la que ha proporcionado el profesor al crear la prueba

y la que ha escrito el alumno durante la resolución de la misma, o la utilización de una IA, como podría ser ChatGPT, Google Gemini, Deepseek, etc, a la que se realizaría una consulta para comprobar si las cadenas del profesor y del alumno son parecidas, y recogiendo la respuesta que devolviera.

Finalmente, se decidió la utilización de la librería fuzzyset.js y la similitud coseno, pues el utilizar una IA, aunque sea para una parte específica de la aplicación, limitaría los entornos en los que se puede utilizar, no tanto en el ámbito universitario ya que las universidades suelen tener redes propias robustas, pero sí más en el ámbito de centros de educación secundaria, ya que estos dependen de la red de la GVA. Además, el no tener que depender de una IA permite que esta aplicación se pueda usar, aunque no haya conexión a internet, pues sirve con que los equipos tanto del profesor como de los alumnos estén conectados en lan para su funcionamiento completo.

## 2 ACTORES

En este punto se definirán a los diferentes actores, y se explicarán cuáles son sus funciones dentro del contexto de la aplicación.

### 2.1 PROFESORES

Los profesores son los responsables de la creación de las pruebas, así como de su supervisión mientras estas se están realizando.

A continuación, se explicarán todas las acciones que puede llevar a cabo un profesor:

- RELLENAR PLANTILLA PRUEBA:

Para poder crear una prueba, el profesor deberá proporcionar un archivo de preguntas que haya rellenado previamente.

Este archivo tiene el formato de un fichero Excel con una estructura por columnas y dos hojas, que separan las preguntas de opción múltiple de las preguntas de desarrollo.

	A	B	C	D	E	F	G	H	I	J	K
1	Valor	Pregunta	Opción 1	Opción 2	Opción 3	Opción 4	Valor	Pista_1	Pista_2	Pista_3	Respuesta Correcta
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											
24											
25											
26											

Plantilla preguntas opción múltiple (Figura 3)

	A	B	C	D	E	F
1	Valor	Pregunta	Pista_1	Pista_2	Pista_3	Respuesta
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						
26						
27						

*Plantilla preguntas desarrollo (Figura 4)*

A todos los profesores que utilicen la aplicación se les facilitará una guía de uso de la aplicación, en la que se incluyen los pasos a seguir desde cómo rellenar el fichero de preguntas hasta cómo crear las pruebas, así como la plantilla de hoja de cálculo para que la rellenen con sus preguntas.

Estos ficheros se encontrarán en la pantalla de “Crear Prueba”.

- Manual
- Plantilla Hoja de Cálculo
- CREAR PRUEBA:

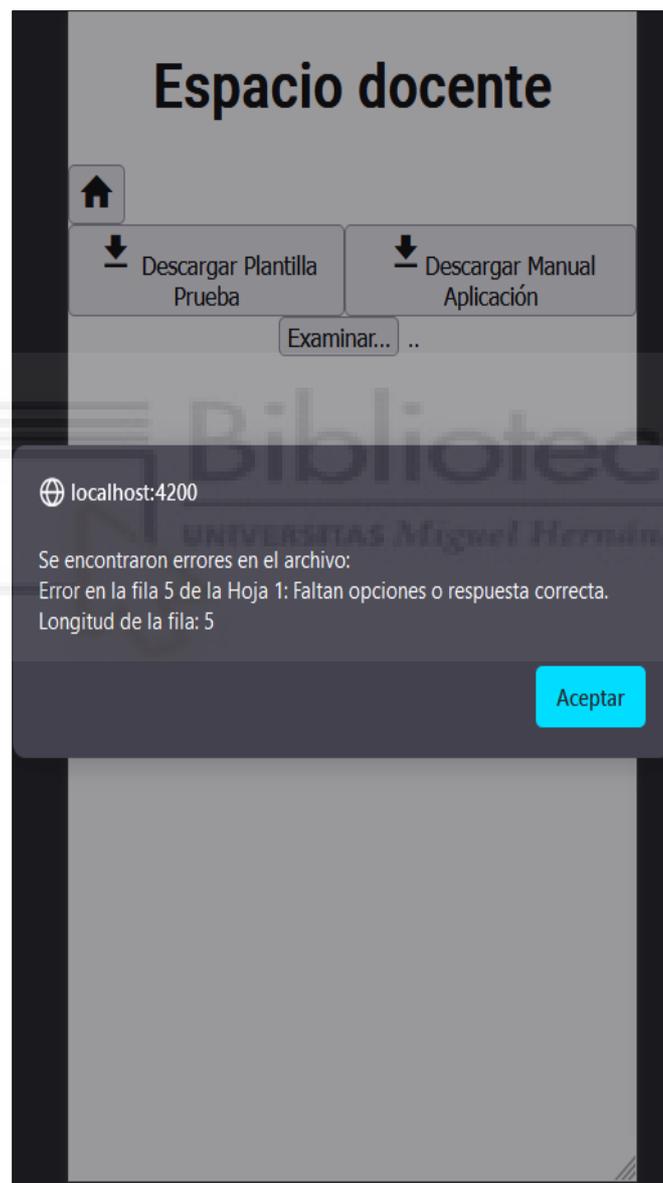
Una vez haya añadido a la plantilla las preguntas que desee, el profesor rellenará un pequeño formulario en el que debe aportar el fichero de las preguntas.

Dependiendo del fichero, el proceso varía ligeramente.

- HOJA DE CÁLCULO:

Una vez subido el fichero, se comprobará que las preguntas del archivo sigan la estructura indicada.

En caso de que se hayan encontrado errores, se notificará de qué tipo son mediante un mensaje por pantalla. Si se tratan de errores dentro del archivo, se indicará en qué fila y columna se han encontrado, así como a qué se debe el fallo, mostrando el valor actual de esa celda y el valor esperado.



*Plantilla error plantilla hoja de cálculo (Figura 5)*

En caso de que no se hayan encontrado errores, se pedirá al profesor que introduzca el número de preguntas del que constará la prueba (mínimo 1). En caso de que seleccione

un número mayor al que hay de preguntas disponibles, se incluirán todas las preguntas en la prueba.

**Espacio docente**

🏠

⬇ Descargar Plantilla Prueba    ⬇ Descargar Manual Aplicación

Examinar...

**Seleccione el número de preguntas y el tiempo estimado(minutos):**

Preguntas

Tiempo(minutos)

Crear Prueba

Biblioteca  
UNIVERSITAS Miguel Hernández

*Indicar número preguntas y tiempo prueba (Figura 6)*

También se introducirá el tiempo que estima el profesor que debe durar la prueba. Este valor se usará en el cálculo de la puntuación final de la prueba.

Una vez introducidos todos los datos, se creará la prueba y el profesor obtendrá un código, que deberá compartir con los estudiantes para que estos puedan acceder a la misma para resolverla.

Examinar...

**Seleccione el número de preguntas y el tiempo estimado (minutos):**

7

4

Crear Prueba

**¡Prueba creada!**

Código de acceso:  
**edxpq6zi**

Comparte este código con otros usuarios para que puedan acceder a la prueba.

Finalizar Prueba

*Plantilla prueba creada (Figura 7)*

- XML:

Una vez subido el fichero, se comprobará que las preguntas del archivo sigan la estructura indicada.

En caso de que se hayan encontrado errores, se notificarán mediante un mensaje por pantalla qué errores son y qué pregunta es la que contiene el error.

Como los ficheros XML son algo más complejos de editar, las preguntas en las que se encuentren errores se descartarán y se procederá a crear la prueba con las preguntas restantes.



*Pantalla errores XML (Figura 8)*

A continuación, el profesor introducirá el número de preguntas y la duración de la prueba, con la diferencia que, una vez introducidos estos datos, se le mostrará por pantalla las diferentes preguntas que se han seleccionado, así como una serie de campos que podrá rellenar.

## Espacio docente

↓ Descargar Plantilla Prueba   ↓ Descargar Manual Aplicación  
Examinar... preguntas-RC\_2799\_GI..0240228-1849(2).xml

**Seleccione el número de preguntas y el tiempo estimado (minutos):**

50  
10   Crear Prueba

Pregunta 1 (Opción Múltiple):

**Texto:** En cuanto a la Calidad de un Servicio, puede ser [.....].

**Opciones:**

- orientado a conexión o sin conexión
- confirmado o no confirmado
- confiable y No confiable
- Todas las otras respuestas son FALSAS

**Respuesta Correcta:** confiable y No confiable

Puntuación: 0   Pista 1:

undefined	Pista 2:
undefined	Pista 3:
undefined	

---

Pregunta 2 (Opción Múltiple):

**Texto:**   
 ¿qué ocurriría si falla una conexión (enlace)?

**Opciones:**

- Impediría que dos nodos pudieran conectarse directamente, pero seguirían conectados a

*Pantalla añadir pistas y puntuación preguntas (Figura 9)*

- **PORCENTAJES Y FINALIZAR PRUEBA:**

El profesor tendrá acceso a una pantalla donde podrá observar el porcentaje de prueba que llevan completada los estudiantes, así como el tiempo restante. Estos datos se irán actualizando cada pocos segundos, para mostrar lo más cercano a tiempo real el avance de la prueba, y estará ordenada descendientemente por porcentaje, es decir, desde el estudiante con mayor porcentaje hasta el que menor porcentaje tenga. También se mostrará la puntuación actual que tengan los participantes de la prueba, la cual depende del número de preguntas que hayan contestado de manera correcta y la cantidad de pistas que hayan utilizado.

En cualquier momento a lo largo de la prueba, el profesor puede decidir hacerla terminar de manera forzosa, bien sea porque ha detectado algún error con las preguntas o porque considere que no puede proporcionar más tiempo para que se resuelva.

La prueba también acabará si el profesor cierra la pestaña en la que la ha creado, pues es esa pestaña la que hace de “servidor” de la prueba. Si se finaliza el proceso de esta manera, no se tendrá acceso a los datos de resolución de los estudiantes.

**Progreso de la Prueba**

**Código de la prueba:** mvuvj0cw

Terminar Prueba

Participante Prueba

Puntuación: 0

---

40%

Participante Enrique

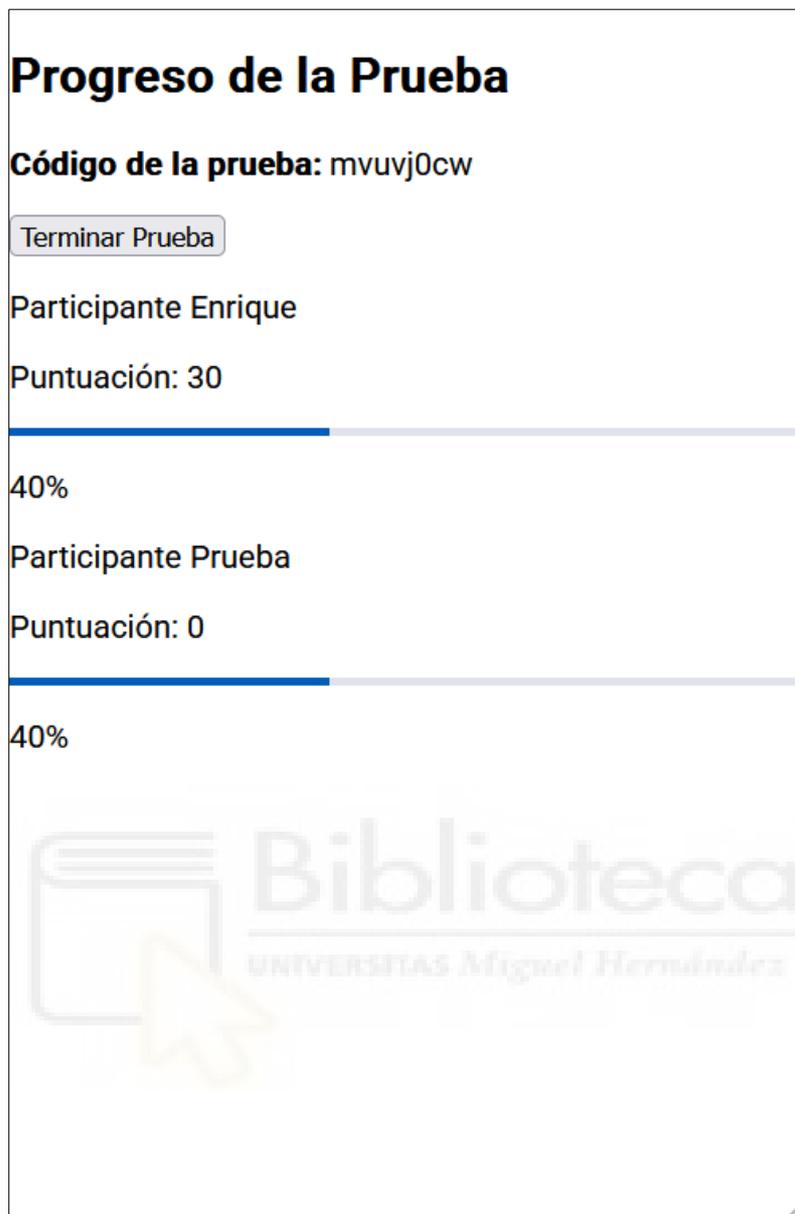
Puntuación: 0

---

0%

The screenshot shows a web interface for monitoring a test. At the top, it displays the title 'Progreso de la Prueba' and the test code 'mvuvj0cw'. Below this is a button labeled 'Terminar Prueba'. The interface then lists participants. The first entry is 'Participante Prueba' with a score of 0 and a progress bar showing 40%. The second entry is 'Participante Enrique' with a score of 0 and a progress bar showing 0%. A watermark for 'Biblioteca UNIVERSITAS Miguel Hernández' is visible in the background.

*Pantalla progreso prueba 1 (Figura 10)*



*Pantalla progreso prueba 2 (Figura 11)*

## 2.2 ESTUDIANTES

Los estudiantes son los usuarios que resolverán la prueba propuesta por el profesor.

- ACCEDER A LA PRUEBA:

Desde la pantalla inicial de la aplicación, el estudiante introducirá el código que le haya proporcionado el profesor para acceder a resolver la prueba que se haya creado.

En esta pantalla también deberá introducir o bien su nombre o un alias, pues esta es la manera en la que se identifica a los estudiantes de cara a la recolección de los datos de la resolución de la prueba.

Hay que mencionar que, aunque está implementada una lista de palabras prohibidas, queda a discreción del profesor cuánta libertad proporciona a los estudiantes para elegir un alias en caso de que no quieran utilizar su nombre.

**Bienvenido a la aplicación  
"Capturar la bandera"**

Nombre/Alias\*  
Prueba

Introduzca el código\*  
edxpq6zi

Acceder

¿Eres profesor? Crea aquí la prueba

*Pantalla introducir nombre y código prueba (Figura 12)*

- RESOLVER PRUEBA:

Durante la prueba, los estudiantes deberán contestar las preguntas que aparezcan para poder avanzar. Si los estudiantes no contestan una pregunta, no podrán pasar a la siguiente. Los estudiantes dispondrán también de una cuenta regresiva que indica el tiempo del que disponen para realizar la prueba con el bono de tiempo. Si el estudiante no consigue finalizar la prueba dentro del límite de tiempo establecido por el profesor, podrá continuar resolviéndola, pero perderá la puntuación asociada con resolverla a tiempo.

**Capturar la bandera**

**Resolver Prueba**

**Tiempo restante: 04:39**

**Progreso: 0%**

**Pregunta 1 de 10**

En cuanto a la Calidad de un Servicio, puede ser [.....]

- orientado a conexión o sin conexión
- confirmado o no confirmado
- confiable y No confiable
- Todas las otras respuestas son FALSAS

Mostrar Pista 1

Mostrar Pista 2

Mostrar Pista 3

Siguiente

*Pantalla resolver prueba (Figura 13)*

**Capturar la bandera**

## Resolver Prueba

**Tiempo restante: 01:58**

**Progreso: 0%**

**Pregunta 1 de 10**

En [.....]

localhost:4200

Selecciona una opción o escribe la respuesta

Todas las otras respuestas son FALSAS

*Pantalla resolver prueba, no avance sin respuesta (Figura 14)*

## Capturar la bandera

### Resolver Prueba

Tiempo restante: 01:55

Progreso: 0%

#### Pregunta 1 de 10

En cuanto a la Calidad de un Servicio, puede ser [.....]

- orientado a conexión o sin conexión
- confirmado o no confirmado
- confiable y No confiable
- Todas las otras respuestas son FALSAS

Mostrar Pista 1

Esta es fácil

Mostrar Pista 2

Mostrar Pista 3

Siguiente

*Pantalla resolver prueba, mostrar pista (Figura 15)*

Si el estudiante decide ver alguna de las pistas disponibles, se le restará la puntuación correspondiente a la pista seleccionada. Por defecto, todas las pistas tienen un coste de diez puntos, pero este valor se puede modificar dentro del programa para ajustarlo a las necesidades del profesor.

Es importante tener en cuenta que la aplicación no guarda sesiones, si el estudiante, por el motivo que sea, sale de la prueba, deberá acceder de nuevo y comenzar a resolverla desde cero.

- **DESCARGAR RESULTADOS PROPIOS PRUEBA:**

Al finalizar la prueba, se presentará a los estudiantes una pantalla donde tendrán un resumen de sus aciertos y fallos, el tiempo que han tardado en resolver la prueba, así como una puntuación final. También se les indicará, cuáles han sido las preguntas que han contestado de manera correcta, y cuáles han contestado de manera incorrecta.

Aparecerá un botón mediante el cual los estudiantes podrán descargar un fichero que incluya el mismo resumen que tienen en pantalla, en caso de que quieran conservarlo, para así ver los resultados de manera más detenida, o hacer un histórico con todos los resultados de las pruebas que vayan resolviendo, etc.



## **Resolver Prueba**

**Tiempo límite: 4 minutos**

**¡Has completado el cuestionario!**

Guardar respuestas



*Pantalla prueba finalizada y descarga fichero resultados (Figura 16)*

```
Resultados de la Prueba
-----
Preguntas y Respuestas:
-----
1. Pregunta: 5ajsndnjasjda
   Tu respuesta: ddd
   Respuesta correcta: aaa
   ¿Correcta?: X

2. Pregunta: 4ajsndnjasjda
   Tu respuesta: ddd
   Respuesta correcta: ddd
   ¿Correcta?: 

3. Pregunta: 3ajsndnjasjda
   Tu respuesta: ddd
   Respuesta correcta: ccc
   ¿Correcta?: X

4. Pregunta: ¿Cómo estas?
   Tu respuesta: sda
   Respuesta correcta: Esta es una pregunta muy ambigua puesto que es subjetiva a la persona que pregunta y responde
   ¿Correcta?: X

5. Pregunta: 1ajsndnjasjda
   Tu respuesta: ddd
   Respuesta correcta: aaa
   ¿Correcta?: X

6. Pregunta: 2ajsndnjasjda
   Tu respuesta: ddd
   Respuesta correcta: bbb
   ¿Correcta?: X

Resumen:
-----
Respuestas correctas: 1/6
Puntaje por respuestas: 14.17%
Bono por tiempo: 15%
Puntaje total: 29.17%
```

*Fichero resultados (Figura 17)*

## 2.3 ADMINISTRADORES

Los administradores del sistema serán los encargados de mantener y asegurar el correcto funcionamiento de la aplicación.

Estas personas serán las que hagan la puesta en marcha de la aplicación, así como su mantenimiento. También podrán añadir al código nuevas funcionalidades o realizar correcciones en el caso de ser estas necesarias.

Dado que la aplicación se trata de un proyecto público, cualquier persona es libre de descargar los ficheros de la aplicación y hacer su propio despliegue.

## 3. PUESTA EN MARCHA

En este apartado se explicará el proceso que debe seguir cualquier persona que quiera descargar el código de la aplicación y ponerla en funcionamiento en su máquina propia.

## 3.1 INSTALAR NODEJS

El primer paso, si bien no puede ser otro, es comprobar que esté NodeJs instalado en la computadora en la que se quiera ejecutar la aplicación. De no ser así, se seguirán los siguientes pasos:

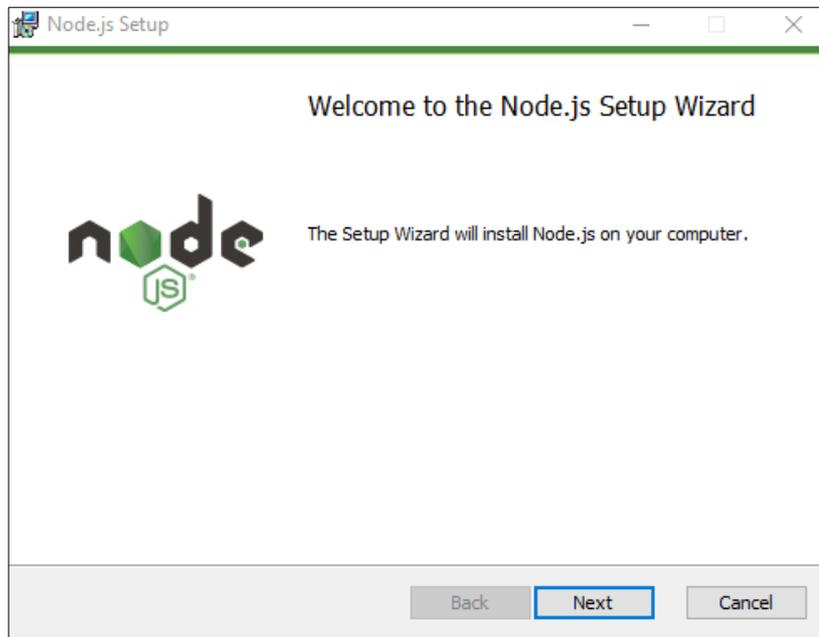
1. Visitar la página de descargas de Node.Js, seleccionar el menú de “Prebuilt Installer” y descargar el ejecutable para el sistema operativo que se esté usando.
2. Hacer doble click en el ejecutable que se acaba de descargar. Es posible que



*Pantalla descargar NodeJS (Figura 18)*

aparezca un aviso cuando se ejecute, en el que se avise de los posibles riesgos que esto conlleva. No obstante, siempre y cuando se haya obtenido el ejecutable de la página oficial, no habrá de qué preocuparse.

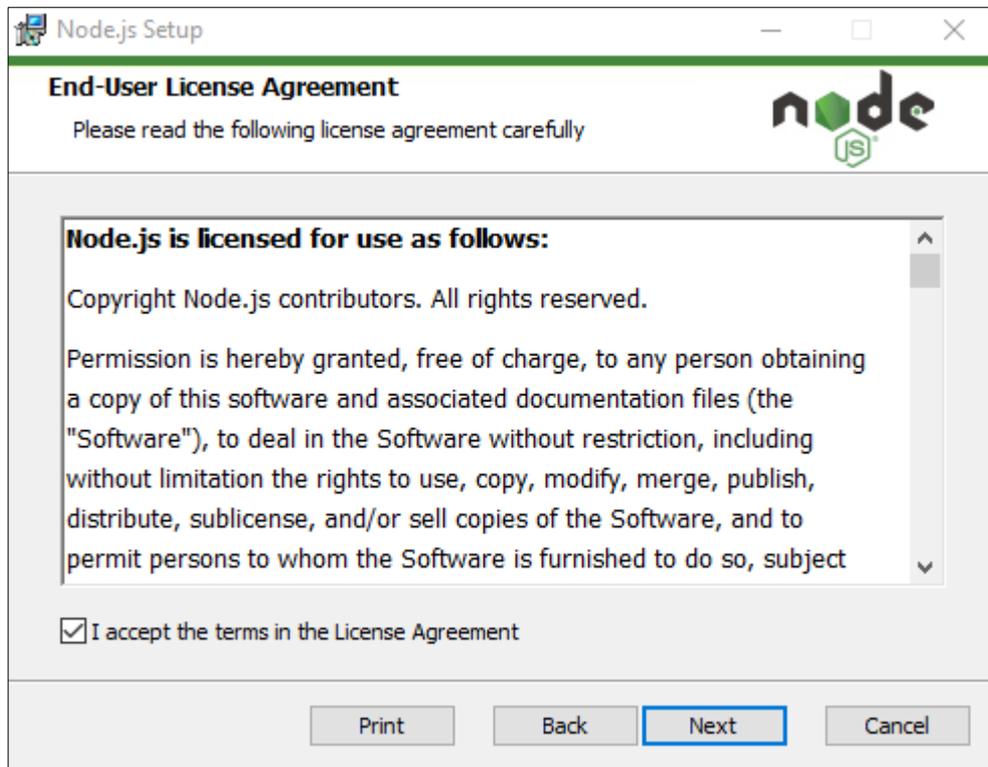
A continuación, aparecerá la siguiente ventana.



*Pantalla Instalación NodeJS (Figura 19)*

Cabe destacar que esta ventana es en Windows 10, puede que la apariencia sea diferente si se usa otro sistema operativo.

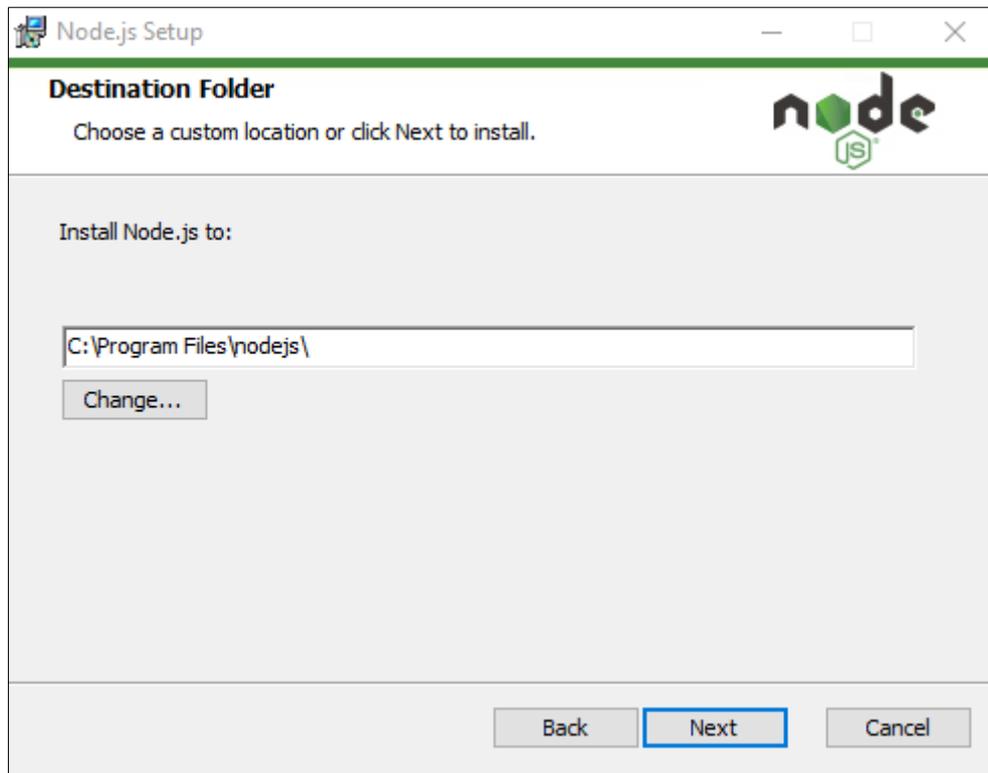
Si se hace click en “Next”, se llegará al acuerdo de licencia con el usuario final (End-User License Agreement)



*Pantalla acuerdo licencia NodeJS (Figura 20)*

Se seleccionará la casilla “I accept the terms of the License Agreement” para continuar con la instalación.

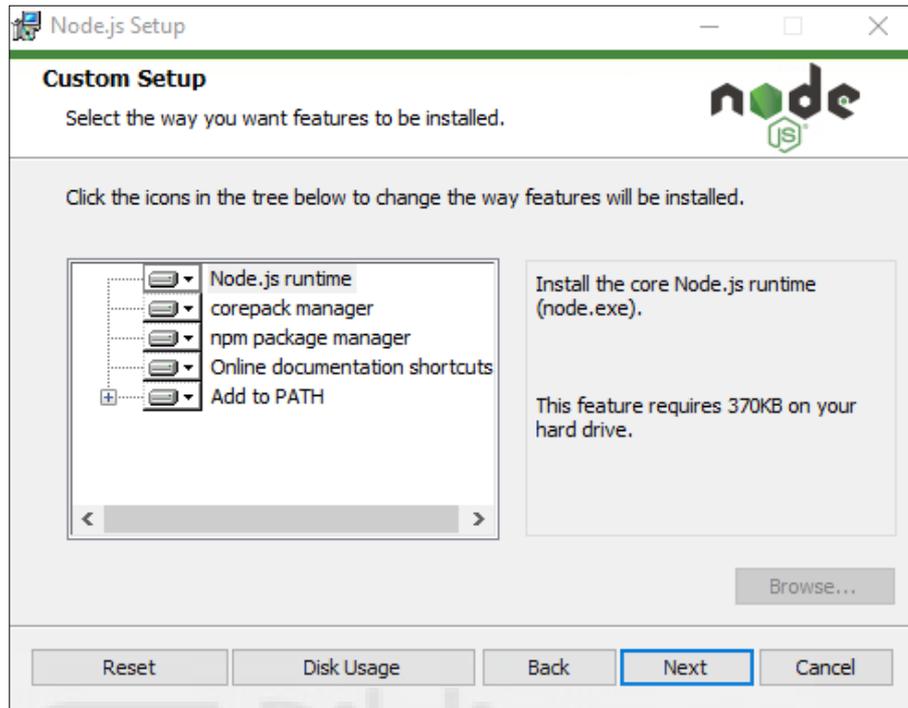
A continuación, se deberá elegir la ruta donde se quiera que instalar NodeJs.



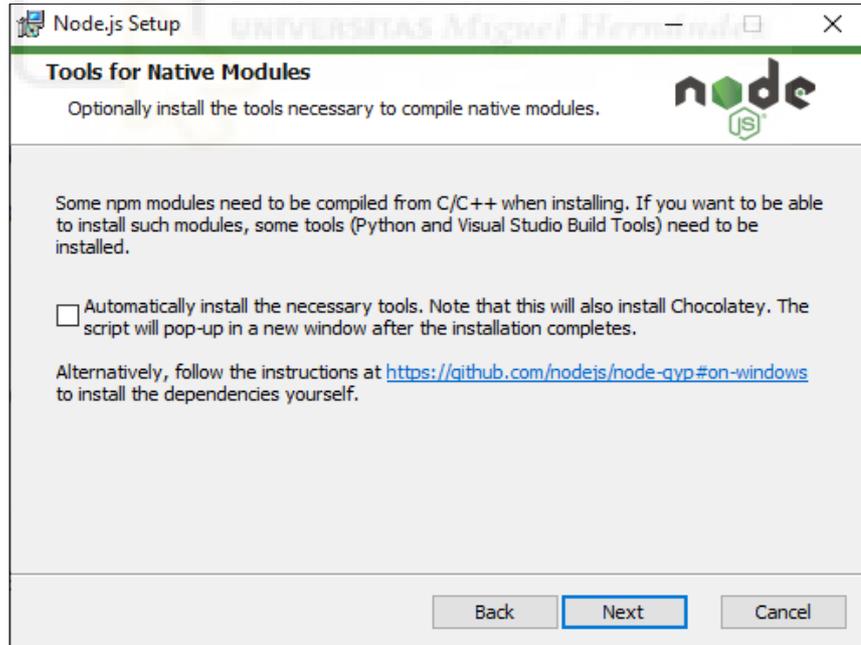
*Pantalla ruta instalación (Figura 21)*

El usuario puede modificar la ruta que aparece por defecto si así lo desea.

Una vez seleccionada la ruta, hacer click en “Next”, aparecerán dos pantallas, en ambas se hará click en “Next”

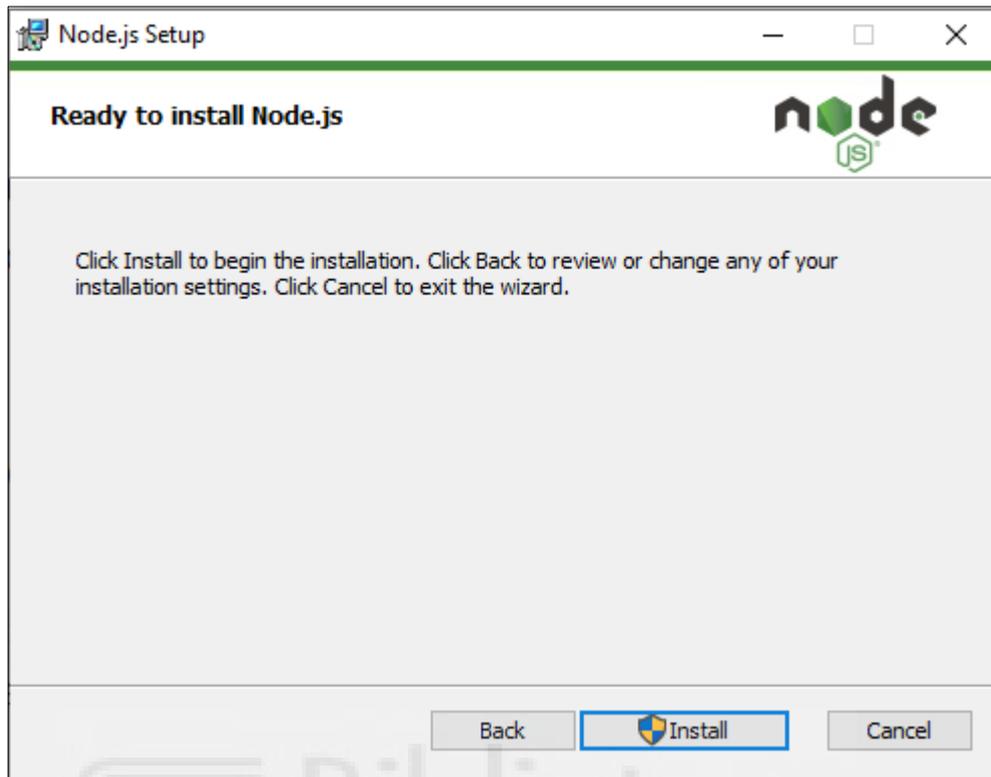


*Pantalla añadir NodeJS a PATH (Figura 22)*



*Opciones extra de instalación (Figura 23)*

El último paso será hacer click en “Install”



*Pantalla instalar (Figura 24)*

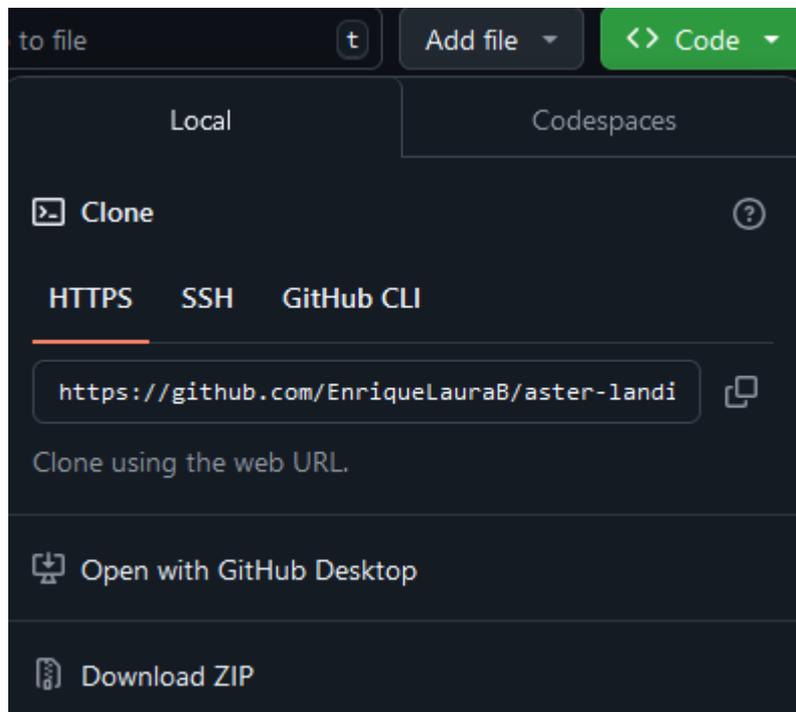
Pasados unos segundos, NodeJs estará instalado en la máquina.

## 3.2 DESCARGAR FICHEROS

El siguiente paso será adquirir los ficheros de la aplicación.

Estos se encuentran en el siguiente repositorio de GitHub.

Para descargar los ficheros, se seleccionará el botón “< >Code”, y la opción “Download ZIP” del menú desplegable



Descargar ficheros de GitHub (Figura 25)

Una vez descargados, lo ideal será que se guarden en una carpeta que tenga accesible el usuario.

En esta imagen se ve aproximadamente lo que debe tener el usuario una vez haya descomprimido el zip (Es una imagen de ejemplo, no todos los archivos/carpetas que aparecen se encontrarán en el zip)

.vscode	29/11/2023 15:09	Carpeta de archivos	
src	29/11/2023 15:09	Carpeta de archivos	
swagger-gen-clients-config	29/11/2023 15:09	Carpeta de archivos	
.browserslistrc	29/11/2023 15:09	Archivo BROWSER...	1 KB
.dockerignore	29/11/2023 15:09	Archivo DOCKERI...	1 KB
.editorconfig	29/11/2023 15:09	Archivo de origen ...	1 KB
.gitignore	29/11/2023 15:09	Documento de te...	1 KB
.gitlab-ci	29/11/2023 15:09	Archivo de origen ...	1 KB
angular	29/11/2023 15:09	Archivo de origen ...	4 KB
dist	29/11/2023 15:09	Archivo WinRAR Z...	480 KB
Dockerfile	29/11/2023 15:09	Archivo	1 KB
karma.conf	29/11/2023 15:09	Archivo JavaScript	2 KB
package	29/11/2023 15:09	Archivo de origen ...	2 KB
package-lock	29/11/2023 15:09	Archivo de origen ...	847 KB
README	29/11/2023 15:09	Archivo de origen ...	2 KB
tsconfig.app	29/11/2023 15:09	Archivo de origen ...	1 KB
tsconfig	29/11/2023 15:09	Archivo de origen ...	1 KB
tsconfig.spec	29/11/2023 15:09	Archivo de origen ...	1 KB

Ficheros descargados (Figura 26)

## 3.3 INSTALAR ANGULAR, ANGULAR MATERIAL Y EJECUTAR APLICACIÓN

Una vez descargados los ficheros, lo siguiente será instalar Angular y Angular Material en nuestro equipo. Para ello se debe acceder a la consola de comandos.

### 3.3.1 WINDOWS:

En la barra de búsqueda, escribir “cmd”, y seleccionar la opción “Símbolo del Sistema”



Acceder al CMD de Windows (Figura 27)

### 3.3.2 MACOS:

1. Haz clic en el icono de Launchpad  en el Dock, escribe Terminal en el campo de búsqueda y, a continuación, haz clic en Terminal.

2. En el Finder , abre la carpeta /Aplicaciones/Utilidades y, a continuación, haz doble clic en Terminal.

### 3.3.3 LINUX.

La aplicación **Konsole** se puede iniciar eligiendo Menú K->Sistema->Konsole (Programa de terminal) desde el sistema de menús del escritorio.

Una vez se haya accedido al terminal, se deberá navegar hasta la carpeta en la que se encuentran los ficheros que se han descargado, haciendo uso de los comandos correspondientes a nuestro sistema operativo.

De manera opcional, si el usuario dispone de un editor de código instalado, como por ejemplo Visual Studio Code, siempre se puede abrir directamente la carpeta donde se encuentran dichos archivos y utilizar la consola integrada del editor.

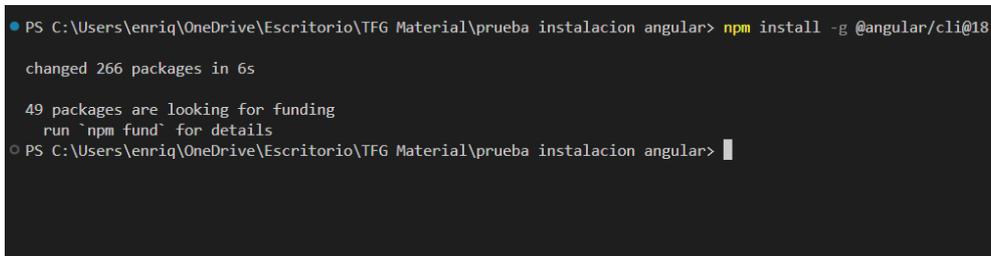
Cuando se haya accedido a la carpeta desde el terminal, comenzará el proceso de instalar Angular.

Esto lo se hará introduciendo la siguiente cadena en el terminal y pulsando “Enter”:

**npm install -g @angular/cli**

Cabe destacar que este comando instalará la versión más reciente de Angular. En caso de que se desee descargar la versión en concreto con la que se desarrolló este proyecto, para evitar posibles problemas de dependencias, se usará el siguiente comando:

**npm install -g @angular/cli@18**



```
PS C:\Users\enriq\OneDrive\Escritorio\TFG Material\prueba instalacion angular> npm install -g @angular/cli@18
changed 266 packages in 6s

49 packages are looking for funding
  run `npm fund` for details
PS C:\Users\enriq\OneDrive\Escritorio\TFG Material\prueba instalacion angular> |
```

*Comando instalación Angular (Figura 28)*

A continuación, se hará lo mismo con Angular Material

**ng add @angular/material**

Es importante tener en cuenta que, para usar este comando, se debe navegar a la carpeta principal del proyecto

Una vez instalado todo, si no aparece ningún error en el código, se podrá ejecutar la aplicación.

Para ello el usuario deberá posicionarse en la carpeta principal del proyecto, si es que no se encontraba ya en ella, e introducirá el siguiente comando:

**ng serve --host=0.0.0.0 --port=4200**

Pasados unos segundos, aparecerá lo siguiente en la consola:



```
PS C:\Users\enriq\OneDrive\Escritorio\Material\tfg> ng serve --host=0.0.0.0 --port=4200

Warning: This is a simple server for use in testing or debugging Angular applications locally. It hasn't been reviewed for security issues.

Binding this server to an open connection can result in compromising your application or computer. Using a different host than the one passed to the "--host" flag might result in websocket connection issues. You might need to use "--disable-host-check" if that's the case.

Initial chunk files | Names | Raw size
polyfills.js | polyfills | 90.20 kB
styles.css | styles | 76.79 kB
main.js | main | 68.53 kB
| Initial total | 235.51 kB

Application bundle generation complete. [2.159 seconds]

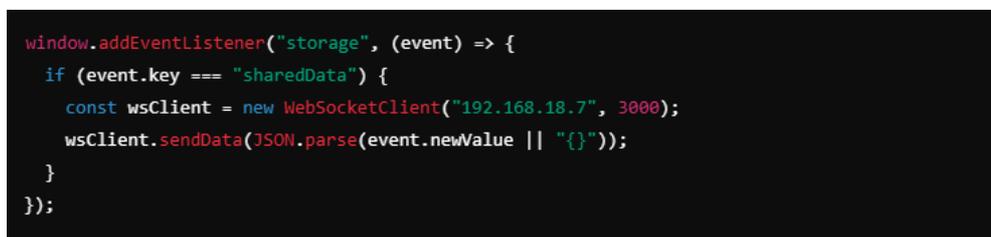
Watch mode enabled. Watching for file changes...
NOTE: Raw file sizes do not reflect development server per-request transformations.
→ Local: http://localhost:4200/
→ Network: http://192.168.18.7:4200/
→ Network: http://192.168.56.1:4200/
→ press h + enter to show help
```

Ejecución proyecto (Figura 29)

Cabe destacar que el comando normal para “lanzar” una aplicación Angular es ng serve, pero debido a que el profesor debe actuar como host, se añaden al comando las variables y valores adicionales.

Si se sigue el enlace que se proporciona a localhost, o se introduce a mano esa url en cualquier navegador, se accederá a la página principal de la aplicación.

Para finalizar, el administrador/profesor deberá establecer en el código del programa la dirección ip del equipo en el que va a trabajar, para que la información almacenada pueda ser accesible por todos los equipos que se conecten a esa sesión.



```
window.addEventListener("storage", (event) => {
  if (event.key === "sharedData") {
    const wsClient = new WebSocketClient("192.168.18.7", 3000);
    wsClient.sendData(JSON.parse(event.newValue || "{}"));
  }
});
```

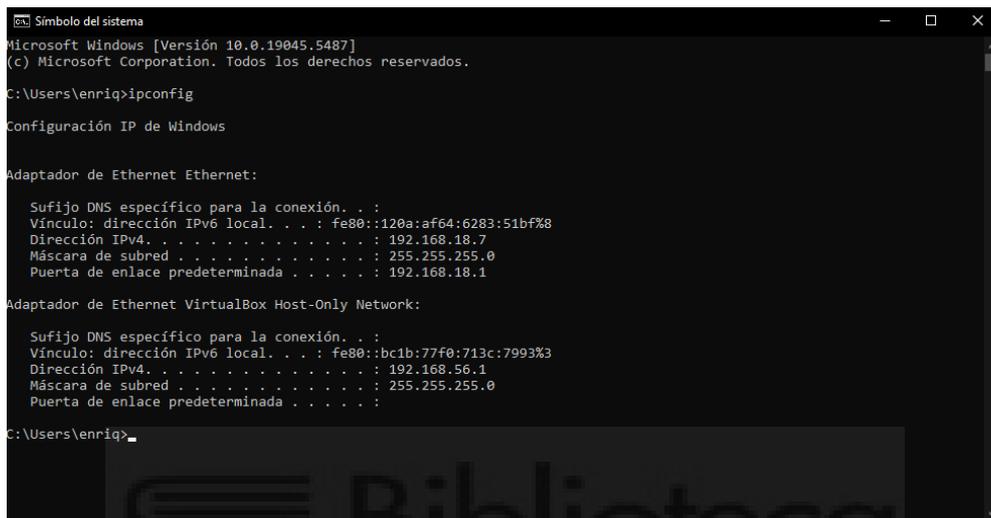
Función de conexión WebSocket (Figura 30)

Para terminar, los usuarios deberán acceder a URL correspondiente

**http://192.168.X.X:4200**

donde las “X” dependen de la dirección IP del equipo del profesor. Para conocer la IP del equipo, se deberá utilizar el comando

### **ipconfig (Windows)**



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.5487]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\enriq>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Sufixo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . . : fe80::120a:af64:6283:51bf%8
    Dirección IPv4. . . . . : 192.168.18.7
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . : 192.168.18.1

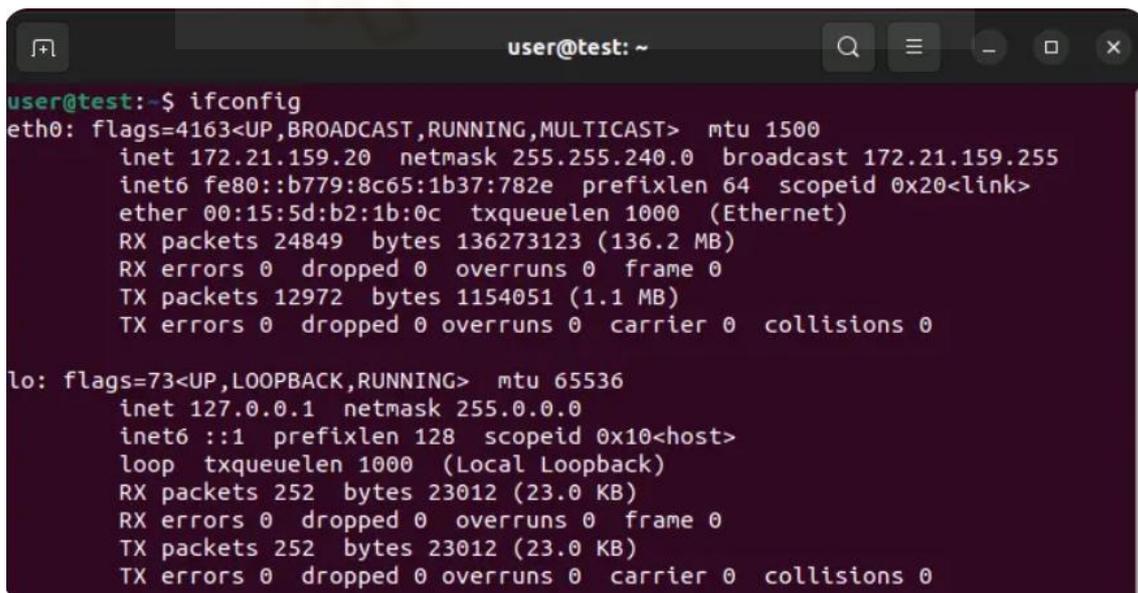
Adaptador de Ethernet VirtualBox Host-Only Network:

    Sufixo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . . : fe80::bc1b:77f0:713c:7993%3
    Dirección IPv4. . . . . : 192.168.56.1
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . :

C:\Users\enriq>
```

Comando ipconfig (Figura 31)

### **ifconfig (Linux/Lliurex y MacOS)**



```
user@test: ~
user@test:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.21.159.20 netmask 255.255.240.0 broadcast 172.21.159.255
    inet6 fe80::b779:8c65:1b37:782e prefixlen 64 scopeid 0x20<link>
    ether 00:15:5d:b2:1b:0c txqueuelen 1000 (Ethernet)
    RX packets 24849 bytes 136273123 (136.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 12972 bytes 1154051 (1.1 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 252 bytes 23012 (23.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 252 bytes 23012 (23.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Comando ifconfig en Linux (Figura 32)

```
Pauls-MacBook-Pro:~ Paul$ ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    options=3<RXCSUM,TXCSUM>
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    ether 28:cf:e9:15:96:4d
    inet6 fe80::2acf:e9ff:fe15:964d%en0 prefixlen 64 scopeid 0x4
    inet 192.168.0.10 netmask 0xfffff00 broadcast 192.168.0.255
    media: autoselect
    status: active
p2p0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 2304
    ether 0a:cf:e9:15:96:4d
    media: autoselect
    status: inactive
vnic0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=3<RXCSUM,TXCSUM>
    ether 00:1c:42:00:00:08
    inet 10.211.55.2 netmask 0xfffff00 broadcast 10.211.55.255
    media: autoselect
    status: active
vnic1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=3<RXCSUM,TXCSUM>
    ether 00:1c:42:00:00:09
    inet 10.37.129.2 netmask 0xfffff00 broadcast 10.37.129.255
```

Comando `ifconfig` en MacOS (Figura 33)

desde la consola del sistema operativo que se esté utilizando, el cual mostrará la información relacionada con su configuración de red.

De esta manera, terminará el proceso de puesta en marcha de la aplicación, y el usuario tiene acceso al código para realizar las modificaciones que desee.

## 4. EVALUACIÓN DE LA PRUEBA

En este apartado se explicarán los criterios de evaluación de la prueba. Dado que no es una prueba solo con preguntas de opción múltiple, sino que también puede incluir preguntas de desarrollo, y mide el tiempo que ha tardado un estudiante en completarla, es importante explicar el razonamiento seguido a la hora de evaluar las pruebas (cuánta importancia se le da a cada apartado, cómo evaluar las preguntas de desarrollo)

## 4.1 EVALUACIÓN DE LAS PREGUNTAS DE DESARROLLO

En este apartado se explicará cómo se evalúan las preguntas de desarrollo, indicando también cómo surgió la idea de incluir este tipo de preguntas y las modificaciones que se han realizado a estos criterios a lo largo del desarrollo de la aplicación.

La idea de añadir preguntas de desarrollo surgió de querer diferenciar esta aplicación de otras aplicaciones similares, como lo podría ser Kahoot!, la cual es una aplicación que permite crear pruebas en las que participan todos los estudiantes de manera simultánea.

Otro de los motivos de incluir este tipo de preguntas fue para proporcionar a los profesores más opciones a la hora de crear las preguntas para una prueba, que no se vean limitados por el formato de las preguntas.

Aunque haya estudios que indiquen que la mejor manera para evaluar de manera objetiva sea realizando pruebas de tipo test, es importante también que los estudiantes desarrollen su capacidad de redacción y síntesis de los conocimientos que poseen. (12)

Una vez establecido los motivos para su inclusión, se comenzará la explicación sobre cómo se evalúan las preguntas de desarrollo.

Para evaluar estas preguntas se utiliza una librería llamada **fuzzysset.js** (13).

Esta librería permite realizar búsquedas difusas en conjuntos de cadenas de texto, aunque en la aplicación solo tiene una cadena objetivo, que es la respuesta proporcionada por el profesor.

La comprobación la realiza a través de la “Similitud Coseno” (14), la cual devolverá un valor entre 0 y 1 (el valor que devuelve la función de la librería es entre 0 y 1, pero el intervalo real de una Similitud Coseno es  $[-1, 1]$ , tratándose de un intervalo cerrado).

```

checkOpenEndedAnswers() {
  this.openEndedQuestions.forEach((question, index) => {
    const correctAnswerSet = FuzzySet([question.correctAnswer.toLowerCase()]);
    const userAnswer = this.userAnswers[index]?.toLowerCase();

    // Comparar la respuesta del usuario con la respuesta correcta usando FuzzySet
    const result = correctAnswerSet.get(userAnswer);

    // Si result no es null y la similitud es mayor a un umbral (ej. 0.7)
    if (result && result[0][0] > 0.7) {
      this.correctAnswersCount++;
    } else {
      // Si es necesario, puedes registrar respuestas incorrectas
      console.log(`Respuesta incorrecta para la pregunta ${index + 1}`);
    }
  });
}

```

*Código funcionamiento similitud coseno (Figura 34)*

Aunque pueda parecer un término algo extraño, la Similitud Coseno se encuentra en diversos aspectos de la vida cotidiana en Internet, como puede ser, por ejemplo, cuando se quiera encontrar nuestra provincia en una página del gobierno, o cuando se busca cualquier tipo de información/objeto en el buscador interno de un portal web.

En este caso, se ha determinado que, si no se llega al menos a un 70% de coincidencia entre la respuesta proporcionada por el estudiante, y la respuesta proporcionada por el profesor, se detectará la respuesta como incorrecta. En caso contrario, es decir un 70% de coincidencia o más, detectará la respuesta como correcta. Se ha elegido este porcentaje ya que es lo suficientemente elevado para evitar pura suerte, pero deja suficiente margen para acomodar diferentes estilos de redacción.

Como se puede apreciar en el código, es relativamente sencillo modificar el porcentaje requerido para registrar un acierto, quedando esto a disposición del profesor o de la persona que esté ejecutando la aplicación.

## 4.2 IMPORTANCIA DE CADA APARTADO

En este apartado se explicará el proceso que se ha seguido para determinar el peso de cada uno de los apartados de los que consta una prueba, siendo estos las preguntas y el tiempo en el que se ha realizado la prueba. Se han estudiado cuatro opciones diferentes a la hora de evaluar las pruebas.

#### 4.2.1 PUNTUACIÓN DIRECTA

El método de puntuación directa es el más directo, valga la redundancia.

Se asigna un valor fijo a cada pregunta, generalmente será  $1/N$  donde  $N$  es el número de preguntas del que consta la prueba, y a la puntuación obtenida de las preguntas, se resta en función del tiempo excedido del límite que haya puesto el profesor, si se da el caso, en caso contrario no se restaría nada.

#### 4.2.2 PUNTUACIÓN NORMALIZADA

Este método consiste en asignar una ponderación a cada parte, por ejemplo 75% a las preguntas y 25% al tiempo empleado.

Estos porcentajes se pueden modificar a voluntad de la persona encargada de la prueba.

#### 4.2.3 PUNTUACIÓN A TRAVÉS DE CURVA DE APRENDIZAJE

Aplicado a este programa, el método consiste en asignar un tiempo determinado a cada pregunta, en vez de tener un tiempo global.

De esta manera, las preguntas que el profesor considere que son más complejas podrán ser más permisivas con el tiempo del que se dispone para contestarlas, mientras que las preguntas que el profesor considere más sencillas tendrán asignado un menor tiempo disponible. (15)

#### 4.2.4 PUNTUACIÓN RASCH

Este método se basa en el Modelo Rasch (16)(17), el cual permite, siempre que el ajuste de los datos sea el idóneo, la medición conjunta de personas e “items” en una misma dimensión o constructo. En este caso, los “items” serán las preguntas de la prueba.

El modelo Rasch se fundamenta en los siguientes supuestos:

- El atributo que se desea medir puede representarse en una única dimensión, que consistirá de manera conjunta de personas e “items”.
- El nivel de la persona en el atributo, y la dificultad del “item”, determinan la probabilidad que la respuesta sea correcta.

Esto viene encapsulado en la siguiente fórmula:

$$\ln \left( \frac{P_{is}}{1 - P_{is}} \right) = (\theta_s - \beta_i)$$

La ecuación indica que el cociente entre la probabilidad de una respuesta correcta y la probabilidad de una respuesta incorrecta a un “item” ( $P_{is} / 1 - P_{is}$ ), es una función de la diferencia en el atributo entre el nivel de la persona ( $\theta_s$ ) y el nivel del “item” ( $\beta_i$ ). Así, cuando una persona responde a un “item” equivalente a su umbral de competencia, tendrá la misma probabilidad de una respuesta correcta y de una respuesta incorrecta ( $P_{is} / 1 - P_{is} = 0,50/0,50$ ). En este caso, el logaritmo natural de  $P_{is} / 1 - P_{is}$ , refleja que la dificultad del “item” es equivalente al nivel de competencia de la persona ( $\theta_s - \beta_i = 0$ ). Si la competencia del sujeto es mayor que la requerida por el “item” ( $\theta_s - \beta_i > 0$ ), la probabilidad de una respuesta correcta será mayor que la de una respuesta incorrecta. Por el contrario, si la competencia del sujeto es menor que la requerida por el “item” ( $\theta_s - \beta_i < 0$ ), la probabilidad de una respuesta correcta será menor que la de una respuesta incorrecta.

Sin embargo, hay estudios que afirman que el modelo Rasch no es infalible, y que depende de cómo se establezca el modelo estadístico para determinar el valor de “dificultad de un item” y “habilidad de una persona” (18).

#### 4.2.5 DECISIÓN FINAL

Finalmente, teniendo en cuenta las personas que van a utilizar y/o modificar esta aplicación, y el objetivo de la misma, se ha decidido utilizar la puntuación normalizada, pues se ha considerado como el método tanto más sencillo de implementar y modificar, como el más sencillo de entender y explicar tanto a profesores como a estudiantes.

Esto no significa que esta opción sea la mejor objetivamente hablando, simplemente se ha considerado como la mejor opción para lo que se quiere conseguir con la aplicación.

Aquellas personas que lo deseen, podrán modificar el código de la función de evaluación de una prueba a su antojo, ya sea bien para modificar la ponderación de cada una de las partes, o bien para implementar una manera de evaluación totalmente diferente.

## 5. CONCLUSIONES

Se ha creado una aplicación que por un lado, permite a los profesores la creación de pruebas más atómicas y rápidas para evaluar, puesto que los resultados se generan de manera automática para su descarga; y por el otro, permite a los estudiantes una manera más “divertida” de realizar una prueba, incluyendo un aspecto de competición para motivarlos tanto durante el transcurso de la prueba como a posterior para obtener mejores

resultados en pruebas futuras, destacar que la aplicación proporciona al alumnado retroalimentación de manera inmediata una vez terminada la prueba (esto último es especialmente útil si se trata con estudiantes que tengan TDA o TDAH(19)).

Se considera que se han cumplido los objetivos establecidos en la introducción.



## 6 ANEXO I

Manual del usuario, creado por el autor del TFG

<https://drive.google.com/uc?export=download&id=1fA-ev9rtgPKsN4X-DrNE0nfSQE-h5FgoUpgdiauxyJY>

Plantilla de hoja de cálculo, creada por el autor del TFG

<https://drive.google.com/uc?export=download&id=1fA-ev9rtgPKsN4X-DrNE0nfSQE-h5FgoUpgdiauxyJY> Hoja de Cálculo



# 7 BIBLIOGRAFÍA

1. Estudio sobre la competitividad como factor motivacional para el aprendizaje:  
<https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2015.01282/full>
2. Estudio de MDPI:  
<https://www.mdpi.com/2075-4663/12/1/37>
3. Estudio Universidad California del Sur:  
[https://web-app.usc.edu/web/rossier/publications/219/Posselt%20Lipson\\_2016\\_JCSD.pdf](https://web-app.usc.edu/web/rossier/publications/219/Posselt%20Lipson_2016_JCSD.pdf)
4. Gamificación en la enseñanza:  
[http://www.scielo.org.co/scielo.php?pid=S0123-12942020000400611&script=sci\\_arttext](http://www.scielo.org.co/scielo.php?pid=S0123-12942020000400611&script=sci_arttext)
5. Front-end Most Popular FrameWorks:  
<https://www.monocubed.com/blog/best-front-end-frameworks/>
6. Angular 18 Dev Notes:  
<https://blog.angular.dev/angular-v18-is-now-available-e79d5ac0affe>
7. Mobile First Index- Documentación Oficial Google:  
<https://developers.google.com/search/docs/crawling-indexing/mobile/mobile-sites-mobile-first-indexing?hl=es-419>
8. Mobile First Index:  
<https://nothingad.com/blog/mobile-first-index-google/>
9. Mobile First - El concepto actual de Diseño Web:  
<https://www.linkedin.com/pulse/mobile-first-el-concepto-actual-en-dise%C3%B1o-web-nicolas-bocacci/>
10. Noticias Priorización Mobile First Webs:  
<https://xatelite.com/google-mobile-first/>

11. INE Porcentaje niños smartphone:

<https://www.ine.es/jaxi/Datos.htm?tpx=50168>

12. Abrir Terminal MacOS:

<https://support.apple.com/es-es/guide/terminal/apd5265185d-f365-44cb-8b09-71a064a42125/mac>

13. Abrir Consola Linux:

<https://help.ubuntu.com/kubuntu/desktopguide/es/terminals.html>

14. Comandos Terminal:

<https://www.xataka.com/basics/comandos-basicos-para-dar-tus-primeros-pasos-consola-windows-cmd>

15. Estudio Preguntas de desarrollo vs tipo test

<https://ice.ua.es/es/jornadas-redes-2016/documentos/tema-3/801419.pdf>

16. Librería Fuzzysset.js:

<https://glench.github.io/fuzzysset.js/>

17. Similitud Coseno:

[https://es.wikipedia.org/wiki/Similitud\\_coseno](https://es.wikipedia.org/wiki/Similitud_coseno)

18. Diseño de pruebas objetivas:

<https://ice.ua.es/es/documentos/recursos/materiales/ev-pruegas-objetivas.pdf>

19. Curva de aprendizaje:

[http://www.scielo.org.co/scielo.php?pid=S2027-83062019000200037&script=sci\\_arttext](http://www.scielo.org.co/scielo.php?pid=S2027-83062019000200037&script=sci_arttext)

20. Modelo Rasch:

[https://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/1476/CHIN\\_CARO\\_OMAR\\_MODELO\\_DE\\_RASCH\\_DICOTOMICO.pdf?sequence=1&isAllowed=y](https://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/1476/CHIN_CARO_OMAR_MODELO_DE_RASCH_DICOTOMICO.pdf?sequence=1&isAllowed=y)

21. Estudio uso Modelo Rasch:

<https://www.psicothema.com/pdf/1029.pdf>

22. Estudio fiabilidad Modelo Rasch

<https://www.redalyc.org/journal/1332/133242591009/html/>

23. Gamificación y TDA

<https://www.mdpi.com/2073-431X/13/2/46>

24. Retroalimentación

<https://www.sciencedirect.com/science/article/abs/pii/S092753711500010X>

