



**Universidad de Valladolid**

**E.T.S. Ingeniería Informática**

**TRABAJO FIN DE GRADO**

**Grado en Ingeniería Informática**

**Experimentación con CAPTCHA sonoros**

Autor:

**Fco. Javier García Maestro**

Tutor:

**Valentín Cardeñoso Payo**



# *Agradecimientos*

*A mi familia y amigos que me han animado y apoyado a lo largo del tiempo de la realización del trabajo.*

*A mi novia que al encontrarse en la misma situación durante este año ha sido capaz de darme fuerza cuando esta se agotaba.*

*A mi tutor Valentín Cardeñoso que me ha guiado y ayudado en la realización de este trabajo fin de grado.*

*A mi empresa por las facilidades que me ha aportado y la ayuda cuando ha sido necesaria.*





# Resumen

---

Los llamados captchas se han convertido en algo habitual en internet, consisten generalmente en una secuencia fija de caracteres distorsionados que deben ser transcritos correctamente por el usuario de la página para poder tener acceso a una determinada zona o realizar determinada acción.

Su finalidad es poder distinguir al usuario humano de una máquina, es decir, previene de accesos automáticos, creación masiva de cuentas de correo, etc... Para un ser humano generalmente no es complicado leer los captchas en cambio para un sistema automático es una tarea muy complicada, por lo que equivale a una garantía de que el sistema lo está utilizando un ser humano.

Para maximizar la accesibilidad debe tenerse en cuenta a los usuarios con problemas de visión, con esa motivación nacen los captchas sonoros, consisten en reproducir al usuario distintos sonidos (generalmente números) que se deben escribir mientras se realiza la escucha o al finalizar está.

La idea del trabajo consiste en realizar una serie de pruebas con distintas configuraciones que deben ser cumplimentadas por los usuarios de la prueba. A su vez se realizan esas pruebas con un sistema reconocedor automático. Una vez recopilados los datos suficientes para poder analizar estas repuestas, llegando a la conclusión de que factores son los que provocan que un captcha sea ininteligible o no.

# Abstract

---

The so-called captchas have become common on the Internet, usually consist of a fixed sequence of distorted characters that must be transcribed correctly by the user of the page to access a certain area or perform a certain action.

Its purpose is to distinguish the human user of a machine, ie prevents automatic access, massive creation of email accounts, etc ... For a human being is generally not difficult to read captchas however for an automatic system is a very complicated task, that means a guarantee that the system is using a human.

To maximize accessibility, must be taken into account the vision impaired users, with that motivation audio captchas are born, consisting of the user play different sounds (usually numbers) to be written while it is being heard or at the end.

The work involves a series of tests with different configurations that must be met by the test users. And once collected enough data to statistically analyze these responses, concluding that factors are causing a captcha unintelligible or not.





# Tabla de contenidos

<b>RESUMEN</b> .....	<b>5</b>
<b>ABSTRACT</b> .....	<b>5</b>
<b>TABLA DE CONTENIDOS</b> .....	<b>7</b>
<b>LISTA DE FIGURAS</b> .....	<b>9</b>
<b>LISTA DE TABLAS</b> .....	<b>10</b>
<b>INTRODUCCIÓN</b> .....	<b>11</b>
1 OBJETIVOS .....	11
2 FASES DEL TRABAJO Y PLANIFICACIÓN.....	13
2.1 <i>Fases del trabajo</i> .....	13
2.1.1 Investigación.....	13
2.1.2 Recopilación del software adecuado.....	13
2.1.3 Realización de una aplicación web .....	13
2.1.4 Publicación de la web y pruebas con usuarios reales.....	14
2.1.5 Pruebas con reconocedor de voz .....	14
2.1.6 Análisis de resultados y realización de memoria.....	14
2.2 <i>Tiempos de trabajo</i> .....	15
3 CONTENIDO DE LA MEMORIA .....	17
4 QUE ES UN CAPTCHA .....	19
4.1 <i>Importancia de los CAPTCHA</i> .....	20
4.2 <i>Algunas formas de romper el sistema</i> .....	21
4.3 <i>Accesibilidad</i> .....	21
4.3.1 Idioma de los CAPTCHA .....	21
4.4 <i>Tipos de CAPTCHA</i> .....	22
4.4.1 CAPTCHA sonoros.....	22
4.4.2 Captchas gráficos.....	23
4.4.3 reCAPTCHA .....	25
<b>DESARROLLO DEL EXPERIMENTO</b> .....	<b>27</b>
5 SOFTWARE Y TECNOLOGÍAS UTILIZADAS .....	27
5.1 <i>Sintetizador de voz</i> .....	27
5.1.1 Tipos de sintetizadores de voz .....	28
5.1.2 Speak.js.....	30
5.2 <i>Aplicación web</i> .....	34
5.2.1 ASP.NET .....	34
5.2.2 Visual Studio 2010.....	35
5.2.3 Reproductor de audio web.....	35
5.2.4 AJAX.....	37
5.3 <i>Base de datos</i> .....	37
5.3.1 SQL SERVER .....	37
5.4 <i>Reconocedor de voz</i> .....	38
5.4.1 API Reconocimiento de voz de Google.....	38
5.4.2 SoX.....	38



5.4.3	Wget.....	39
6	DISEÑO .....	41
6.1	<i>Aplicación web del experimento</i> .....	41
6.1.1	Diseño inicial .....	41
6.1.2	Diseño final.....	43
6.2	<i>Base de datos</i> .....	53
7	EXPERIMENTO Y PRUEBAS .....	57
7.1	<i>Mecánica del experimento</i> .....	57
7.1.1	Envío de propuestas .....	57
7.2	<i>Mecánica de las pruebas</i> .....	58
7.2.1	Pruebas con el reconocedor automático.....	59
8	DESPLIEGUE Y PUESTA EN MARCHA.....	61
8.1	<i>Servidor web</i> .....	62
8.1.1	Somee.com.....	62
8.2	<i>Servicio de Correo</i> .....	62
8.2.1	Yahoo .....	62
8.3	<i>Mensajes de invitación</i> .....	63
8.3.1	Correo electrónico.....	63
8.3.2	Redes sociales .....	64
8.4	<i>Encuesta de opinión</i> .....	65
	<b>RESULTADOS DE LA PRUEBA .....</b>	<b>67</b>
	<i>Resultados de pruebas con usuarios</i> .....	67
	Resultado general de la prueba .....	67
	Número de fallos, aciertos, pasar por captcha individual .....	68
	Resultados agrupados por rango de edad.....	69
	Resultados agrupados por frecuencia de acceso a internet.....	72
	Resultados agrupados por sexo .....	74
	Resultados por número de intentos.....	76
	Resultados agrupados por parámetros .....	77
	<i>Resultados de pruebas con reconocedor automático de Google</i> .....	79
	<i>Resultados de la encuesta</i> .....	81
	<i>Análisis de resultados</i> .....	83
	<b>CONCLUSIONES .....</b>	<b>85</b>
	<b>BIBLIOGRAFÍA Y WEBS CONSULTADAS .....</b>	<b>87</b>
	<i>Bibliografía</i> .....	87
	<i>Webs consultadas</i> .....	87
	<b>ANEXOS .....</b>	<b>91</b>
	APENDICE A ESPEAK.....	91
	<i>Historia / Que es</i> .....	91
	<i>Características</i> .....	91
	<i>Lenguajes disponibles</i> .....	92
	<i>Síntesis por formantes</i> .....	93
	<i>Métodos de síntesis en eSpeak</i> .....	94
	APENDICE B DATOS DE LA ENCUESTA.....	95
	APENDICE C CONTENIDOS DEL CD .....	99



# Lista de Figuras

FIGURA 1 - UNO DE LOS CAPTCHAS MÁS POPULARES EN INTERNET .....	19
FIGURA 2 - TIPOS DE CAPTCHA CARACTERES .....	23
FIGURA 3 - TIPOS DE CAPTCHA TEXTO DISTORSIONADO.....	23
FIGURA 4 - TIPOS DE CAPTCHA FIGURAS INDEPENDIENTES .....	23
FIGURA 5 - TIPOS DE CAPTCHA HAZ CLIC EN LA CELDA CORRECTA .....	24
FIGURA 6 - TIPOS DE CAPTCHA HAZ CLIC EN EL CÍRCULO ABIERTO .....	24
FIGURA 7 - TIPOS DE CAPTCHA COMPLETA EL PUZLE .....	24
FIGURA 8 - LOGO DE RECAPTCHA.....	25
FIGURA 9 - TIPOS DE CAPTCHA RECAPTCHA .....	25
FIGURA 10 - LOGO DE ESPEAK.....	30
FIGURA 11 - LOGO DE EMSCRIPTEN .....	30
FIGURA 12 - COMPATIBILIDAD CON ELEMENTOS AUDIO EN NAVEGADORES .....	32
FIGURA 13 - COMPATIBILIDAD PARA ARRAYS TIPADOS EN NAVEGADORES.....	33
FIGURA 14 - LOGO DE ASP.NET .....	34
FIGURA 15 - LOGO DE VISUAL STUDIO .....	35
FIGURA 16 - FORMATOS DE AUDIO COMPATIBLES EN NAVEGADORES .....	35
FIGURA 17 - LOGO DE AJAX.....	37
FIGURA 18 - LOGO DE SQL SERVER .....	37
FIGURA 19 - LOGO DE GOOGLE .....	38
FIGURA 20 - DISEÑO INICIAL DE PANTALLA CAPTCHA.....	41
FIGURA 21 - FORMULARIO DE ACCESO I .....	43
FIGURA 22 - FORMULARIO DE ACCESO II .....	44
FIGURA 23 - VENTANA DE PRUEBA DE CAPTCHA.....	46
FIGURA 24 - VENTANA DE AGRADECIMIENTO .....	47
FIGURA 25 - ESQUEMA DE TABLAS BASE DE DATOS.....	56
FIGURA 26 - LOGO DE SOMEE.COM .....	62



## Lista de Tablas

---

TABLA 1- TIEMPOS DEDICADOS/ESTIMADOS DE TRABAJO .....	15
TABLA 2- PARÁMETROS DE LOS CAPTCHAS SONOROS.....	50
TABLA 3 - RESULTADOS GENERALES DE LA PRUEBA.....	67
TABLA 4 - RESULTADOS AGRUPADOS POR CAPTCHA.....	68
TABLA 5 - RESULTADOS AGRUPADOS POR CAPTCHA Y POR RANGO EDAD.....	71
TABLA 6 - RESULTADOS AGRUPADOS POR RANGO EDAD .....	71
TABLA 7 - RESULTADOS CAPTCHAS POR AGRUPADOS POR CAPTCHA Y FRECUENCIA USO INTERNET .....	73
TABLA 8 - RESULTADOS AGRUPADOS POR FRECUENCIA USO INTERNET .....	74
TABLA 9 - RESULTADOS AGRUPADOS POR CAPTCHA Y SEXO.....	75
TABLA 10 - RESULTADOS AGRUPADOS POR SEXO .....	75
TABLA 11 - PORCENTAJE DE ERRORES POR NÚMERO DE NÚMEROS.....	77
TABLA 12 - PORCENTAJE DE ERRORES POR ESPACIO ENTRE PALABRAS .....	77
TABLA 13 - PORCENTAJE DE ERRORES POR VELOCIDAD.....	78
TABLA 14 - RESULTADOS COMPRENSIÓN SINTETIZADOR DE VOZ .....	79
TABLA 15 - RESULTADOS RECONOCEDOR DE VOZ (SIN RUIDO).....	80
TABLA 16 - CAPTCHAS MÁS COMPLICADOS .....	84



# Introducción

---

## 1 Objetivos

El objetivo del trabajo consiste en realizar un estudio experimental sobre los Captchas sonoros. Se pretende descubrir la diferencia del nivel de comprensión para el ser humano de distintas pruebas que incluyen sonidos (captchas sonoros) que los usuarios deben resolver. A su vez se pretenden realizar las mismas pruebas frente a un reconocedor de voz automático.

Una vez recopilados todos los resultados se realizará una estimación de la dificultad que ha entrañado cada captcha para un ser humano y compararla con la dificultad que ha tenido para el reconocedor de voz. De este modo podremos concluir que parámetros son los que hacen que un captcha sea comprensible para un humano y no para un sistema automático o al revés.

Para ello se realizará una aplicación web que nos permita realizar el experimento, los captchas que utilizaremos consisten en una superposición de dos sonidos:

- Secuencia de números hablados.
- Ruido de fondo.

Las pruebas consisten en experimentar en la parte de la secuencia de números hablados con distintos factores como el tono, espaciado entre palabras, etc...

Se contará con una base de usuarios que serán los que realicen las pruebas y que nos darán todos los datos que nos permitirán sacar las conclusiones finales. Los resultados de cada prueba se almacenan (éxitos, fallos, reintentos) para obtener conclusiones de la complejidad de la prueba particular.





## 2 Fases del trabajo y planificación

### 2.1 Fases del trabajo

#### 2.1.1 Investigación

En un primer lugar se busca información sobre los captchas sonoros, que son, para que se utilizan, su importancia en la actualidad, como es su modo de funcionamiento, que tipo de ruido incluyen, etc...

Se hace un estudio sobre su uso en páginas web, utilidad real.

Se analiza el problema de la accesibilidad y problemática del idioma (los números del captcha deben ser hablados en tu idioma para poder entenderlos)

#### 2.1.2 Recopilación del software adecuado

Se deben localizar distintas tecnologías o aplicaciones que sean los más adecuados para la realización del trabajo, una idea general de las categorías del software recopilado sería:

- Tecnología para la realización de una aplicación web.
- Sistema gestor de base de datos para persistir la información recopilada.
- Tecnología que permita la reproducción en una aplicación web de secuencias de audio.
- Sintetizador de voz, que pueda ser embebido en un entorno web, para generar las secuencias de números.
- Reconocedor de voz que permita el envío de secuencias de audio.

#### 2.1.3 Realización de una aplicación web

Se diseña una aplicación web para realizar el experimento en la que de una manera sencilla para el usuario se le muestran distintas pruebas con captchas sonoros, estos captchas tienen diferentes características entre uno y otro, en cada captcha se mezclan aleatoriamente ruidos y números siguiendo un patrón para cada captcha.

La web incluirá un sistema de autenticación de usuarios, para evitar acceso masivo o datos incorrectos que estropearían el resultado del experimento.



Se compondrá una encuesta que los usuarios pueden rellenar para recibir una retroalimentación por parte de los usuarios de lo que les ha parecido el experimento con los captchas.

## 2.1.4 Publicación de la web y pruebas con usuarios reales

Se busca un servidor adecuado en el que se pueda publicar la página, y se comprueba que todo funciona correctamente, web, base de datos etc...

Se encuentra un servicio de correo para mandar los correos necesarios de autenticación a los usuarios.

Un enlace a la aplicación web se envía a los usuarios a modo de test que deben rellenar. Los datos de respuesta de esos test se recopilan y analizan.

## 2.1.5 Pruebas con reconocedor de voz

Se debe investigar los reconocedores de voz disponibles que permitan enviar una secuencia de audio y que devuelvan una respuesta. Eso sería lo más adecuado en el caso de no disponer de ello se podría hacer el experimento con un reconocedor de voz normal enviando el sonido por un micrófono.

Una vez elegido el reconocedor de voz se realizan las mismas pruebas que se propusieron a los usuarios humanos con un reconocedor automático de voz, recopilando los resultados y observando el nivel de comprensión del reconocedor.

## 2.1.6 Análisis de resultados y realización de memoria

Al disponer de todos los datos se realizará un análisis con las conclusiones alcanzadas. Disponemos de las respuestas de los usuarios a cada uno de los captchas. Con ellas se pueden realizar estadísticas de fallos y acierto que se compararán con las características de cada uno de los captchas. De este modo conoceremos que parámetro de los usados es el que complica más la comprensión de los captchas.

Por parte del reconocedor automático es igual, con las respuestas recopiladas podemos conocer que parámetros hacen que el captcha sea fácil o difícil de reconocer.

Una vez se disponga de estos datos se realizara la memoria del trabajo.



## 2.2 Tiempos de trabajo

Los tiempos de trabajo dedicados finalmente a cada una de las fases de un proyecto siempre varían respecto a los tiempos previstos. En mi caso además se incluye otro factor como la menor disponibilidad que provoca el estar trabajando.

A continuación se muestra una tabla en la que se detallan las diferentes fases, su duración estimada en un primer momento y su duración final.

Fase	Nombre	Estimación inicial	Dedicación real	Observaciones
1	Investigación	80 horas	120 horas	Se han necesitado más horas ya que han sido necesarios algunos cambios de tecnologías en el transcurso del proyecto.
2	Recopilación del software adecuado	30 horas	50 horas	La búsqueda de software adecuado como su puesta a punto ha sido complicada en particular el sintetizador de voz y se ha llevado bastante tiempo.
3	Realización de una aplicación web	100 horas	200 horas	Se ha ido evolucionando el diseño dando lugar a trabajo añadido de adaptación, de nuevos esquemas de base de datos, software.
4	Publicación de la web	15 horas	50 horas	Se han encontrado problemas con el servidor elegido que han tenido que solucionarse, problemas con servidores de correo de tamaño de los archivos publicados, etc..
5	Pruebas con usuarios reales	1 semana	3 semanas	La prueba ha durado más tiempo dado que es complicado y laborioso conseguir un número aceptable de usuarios que hagan la prueba.
6	Pruebas con reconocedor de voz	30 horas	20 horas	Las pruebas con el reconocedor de Google han sido hechas en menos tiempo que el estimado.
7	Análisis de resultados y realización de memoria	50 horas	80 horas	La estimación era errónea ya que para la realización de la memoria se requiere bastante más tiempo así como para el análisis de resultados.

Tabla 1- Tiempos dedicados/estimados de trabajo





## 3 Contenido de la memoria

La memoria contiene una explicación detallada de las distintas partes que componen el trabajo.

### **Explicación de que es un captcha:**

En la que se pone al lector en el contexto adecuado sobre que son los captchas su uso en la actualidad los distintos tipos etc...

### **Tecnologías y software utilizado:**

Sección en la que se detallan las tecnologías utilizadas para realizar el trabajo y los programas necesarios. Así como el uso que se ha hecho de ellos.

### **Diseño de la aplicación web:**

Se detallan las diferentes etapas del diseño de la aplicación web, los primeros acercamientos y el diseño final.

### **Secciones de la aplicación web:**

Se comentan cada una de las páginas de las que se compone la aplicación web, además se explica el funcionamiento de distintas partes de la aplicación, en la que se explica la autenticación de usuarios, los tipos de captchas etc...

### **Diseño de la base de datos:**

Estructura del diseño final de la base de datos, tablas y significado de los diferentes campos.

### **Experimento y pruebas:**

Se detalla la mecánica del experimento y las opciones disponibles del usuario al resolver las pruebas.

### **Puesta en marcha:**

Se detallan las acciones necesarias y servicios necesarios para la puesta en marcha. Así como los mensajes de invitación a los usuarios.

### **Recopilación de resultados:**

Los datos recopilados se analizan tanto para usuarios como para el reconocedor de voz y se exponen los resultados.



### **Conclusiones:**

Después de la realización de la prueba y la recolección de datos, reflexiones sobre el objetivo del proyecto y su puesta en marcha.

### **Bibliografía y webs consultadas:**

Relación de documentación consultada durante la realización del proyecto.

### **Apendice sobre eSpeak:**

eSpeak es la base del sintetizador de voz usado en la aplicación, en esta sección se explican sus características y funcionamiento.

### **Resultados de la encuesta:**

Se muestran los datos recopilados en la encuesta disponible al final de cada prueba con todas las respuestas de los usuarios.

## 4 Que es un CAPTCHA

Vamos a comentar brevemente lo que es un captcha y su función en la informática actual. Son una prueba que nos permite asegurar hasta un cierto punto que si el captcha ha sido resuelto ha sido por un ser humano.

**C** ompletely  
**A** utomated  
**P** ublic  
**T** uring test to tell  
**C** omputers and  
**H** umans  
**A** part

(Prueba de Turing pública y automática para diferenciar máquinas y humanos)

Se utilizan principalmente en Internet o en entornos informáticos, es muy habitual encontrarse con captchas al navegar por páginas en Internet. Generalmente suelen solicitar al usuario que escriban una serie de caracteres que se ven distorsionados.



Figura 1 - Uno de los captchas más populares en internet

El termino CAPTCHA se comenzó a usar en el año 2000 por Luis von Ahn, Manuel Blum y Nicholas J. Hopper de la Carnegie Mellon University, y John Langford de IBM.

La prueba de Turing estándar es controlada por un humano, pero en este caso el control del captcha es por parte de una máquina por lo que se denomina **Prueba de Turing Inversa**.

Los captchas se han posicionado como una alternativa potente y válida para distinguir entre personas y máquinas. La idea principal es que en el caso de la lectura de caracteres distorsionados o tachados es una tarea sencilla para un ser humano y aún añadiéndole mucha complejidad es posible la resolución de la prueba, en cambio ese tipo de pruebas son muy complicadas de resolver para un sistema automático.



De todas formas un captcha no asegura al 100% que lo que lo ha resuelto sea un humano o una máquina, hay multitud de métodos para conseguir romper la seguridad de los captchas y es una carrera continua porque los sistemas automáticos avanzan muy rápido y los captchas deben hacerse más complejos o innovar a la par que estos sistemas automáticos avanzan y son capaces de resolver las captchas que hace un tiempo no podían.

## 4.1 Importancia de los CAPTCHA

Los captchas son una herramienta muy útil en la actualidad ya que hay millones de webs en internet abiertas a cualquiera que quiera acceder, por lo que es importante distinguir entre los usuarios que están usando la página y asignar los recursos a quien está usando la página realmente.

Su uso combinado con otras medidas de seguridad es muy importante por ejemplo en las siguientes situaciones:

- Creación de cuentas de correo.
- Añadir comentarios a un blog o foro.
- Descarga/Subida de ficheros.
- Participación en concursos, sorteos,
- etc...

Si no existiera esta especie de barrera o filtro podrían ocurrir situaciones como que un sistema automático externo creara miles de cuentas de correo en un tiempo muy corto, estropear los resultados de una encuesta votando masivamente por una opción concreta, limitar o extinguir por completo el ancho de banda de una web con descargas o subidas masivas.

Esta prueba provee de control al administrador de la web, que consigue con los captchas que los recursos, a menudo escasos sean usados por los usuarios auténticos.



## 4.2 Algunas formas de romper el sistema

### OCR (Reconocimiento óptico de caracteres)

Este método solamente es útil si el captcha es medianamente sencillo, por ejemplo no cuenta con diferentes fuentes tipográficas, tamaños, colores, caracteres distorsionados o ruido en el fondo.

Como por ejemplo la aplicación Linux GOCR, que se basa sobre todo en el aprendizaje para distinguir las distintas formas que contiene el captcha.

### Envío del captcha a otras páginas para que lo resuelvan usuarios humanos

El método consiste en que el sistema que se encuentra frente a un captcha que no puede resolver, el sistema tiene capacidad de mostrar ese captcha en alguna web o aplicación que se use a menudo para acceder a recursos como imágenes, etc.. El sistema sólo tiene que esperar a que una persona resuelva el captcha por él y de esta manera romper el captcha.

### Recolección de muestras de sonido

Se crea una librería de sonidos que contenga cada carácter distinto que pueda sonar y con todas sus variantes, después el sistema usa algo similar a un reconocimiento del habla para intentar romper el captcha.

## 4.3 Accesibilidad

Las pruebas que presentan los captchas tienen algunos problemas evidentes de resolución con personas con discapacidades visuales o auditivas.

Los captchas actuales son principalmente visuales, y un gran número de personas, potenciales usuarios de un servicio pueden ser incapaces de acceder debido a problemas de vista u otros problemas.

De ahí nace la idea del captcha sonoro, es un captcha que permite a personas con problemas visuales resolver captchas que de otro modo no serían posibles resolver con los captchas gráficos.

### 4.3.1 Idioma de los CAPTCHA

Un problema importante en la accesibilidad de los captchas sonoros es el idioma, el captcha debe poder detectar el idioma del usuario que está escuchando el captcha, porque el captcha sonoro no es como las letras que son universales y cualquiera puede transcribirlas.



Unos números en inglés pueden no ser comprendidos por un usuario español por ejemplo, por lo que es importante adaptarse y que las secuencias a transcribir sean en el idioma adecuado. Eso por supuesto complica en gran medida el desarrollo del captcha y su uso.

## 4.4 Tipos de CAPTCHA

### 4.4.1 CAPTCHA sonoros

Los captchas sonoros que son la herramienta para llegar al objetivo de este trabajo, tienen ventajas respecto a los captchas gráficos porque permiten ser usados por un grupo de usuarios que hasta su aparición se quedaban fuera de la posibilidad de resolver captchas, estos usuarios son los usuarios con problemas visuales.

Tienen la ventaja que tanto los usuarios sin problemas como los usuarios con problemas de vista son capaces de resolver los captchas sonoros.

Habitualmente el sonido emitido por el captcha consiste en dos partes.

#### 1. Ruido de fondo:

Suele estar a un nivel de volumen medio, lo suficiente para intentar evitar el reconocimiento de voz automático. Puede ser ruido natural o generado por ordenador.

#### 2. Secuencia de números o letras:

Mientras se reproduce el sonido de fondo, una o varias voces van diciendo una secuencia de números o letras, la secuencia puede reproducirse más o menos rápido o con distintos parámetros de distorsión.

Dependiendo del captcha la respuesta puede escribirse a medida que se escucha o debe ser escrita al final. Añadiendo más dificultad a la prueba teniendo que obligar al usuario a memorizar la secuencia.

Volver a reproducir el sonido  
Descargar el sonido en MP3

Escribe lo que oigas

Submit

## 4.4.2 Captchas gráficos

Existen muchos tipos distintos de captchas pero la mayoría se basan en mostrar al usuario letras o números mas o menos distorsionados que este debe teclear y validar.

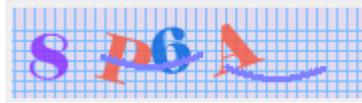


Figura 2 - Tipos de captcha caracteres



Figura 3 - Tipos de captcha texto distorsionado

En otros casos se muestran al usuario figuras o formas que debe identificar y comparar con una lista.

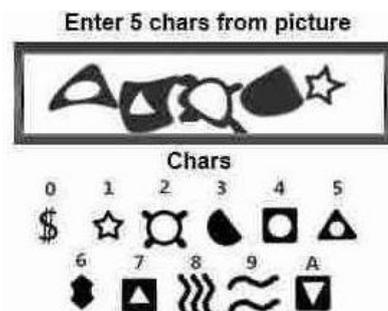


Figura 4 - Tipos de captcha figuras independientes

Otros sólo solicitan al usuario hacer click en el lugar correcto. Por lo que son más cómodos y efectivos.

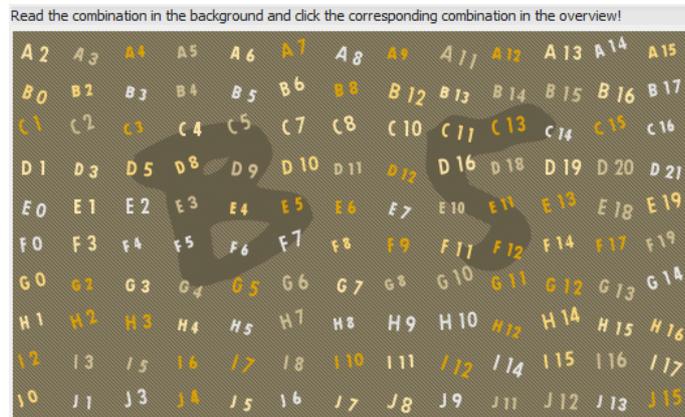


Figura 5 - Tipos de captcha haz clic en la celda correcta

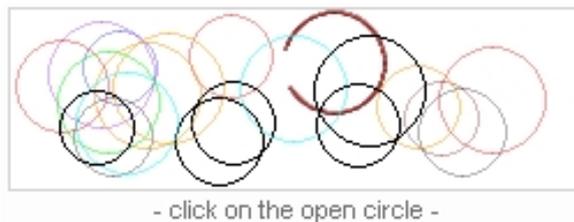


Figura 6 - Tipos de captcha haz clic en el círculo abierto

Por otra parte otros son más complejos y requieren del usuario un esfuerzo mayor como este que te muestra una imagen del puzzle resuelto y debes colocar las piezas en su posición.



Figura 7 - Tipos de captcha completa el puzzle

### 4.4.3 reCAPTCHA



Figura 8 - Logo de reCaptcha

Un ejemplo interesante es reCaptcha que es un servicio de captchas gratuito de Google que se puede usar por cualquiera que se registre y que es muy completo.

Los captchas generados por el componente consisten en dos palabras distorsionadas que se deben escribir. El sistema sabe la respuesta correcta de una de las dos palabras y es la que se utiliza para saber si se ha respondido correctamente, la otra palabra es una palabra escaneada que se añade a la base de datos de libros de Google considerando que si la palabra es correcta, la otra palabra también.



Figura 9 - Tipos de captcha reCAPTCHA

Este tipo de captcha además dispone de una opción de resolverlo por sonido, bastante complicada de resolver dado que no tiene en cuenta el idioma del usuario que resuelve el captcha y solamente dicta palabras en inglés.





# Desarrollo del experimento

---

En esta sección se detallan varios aspectos relativos al desarrollo de las distintas partes que hacen posible la realización del experimento.

Los temas a tratar son los siguientes:

- Tecnologías utilizadas para su realización.
- Diseño de la aplicación web que soportará la prueba.
- Funcionamiento de las distintas pruebas.
- Despliegue y puesta en marcha.

Dado que el objetivo del trabajo y su finalidad es la realización del experimento y la recolección de resultados. No se desarrollan en profundidad temas relativos al desarrollo de la aplicación web, temas como análisis, desarrollo, plan de pruebas, etc... Centrándose en esta memoria en el desarrollo de las distintas áreas que afectan al experimento objeto del trabajo.

## 5 Software y Tecnologías utilizadas

### 5.1 Sintetizador de voz

La síntesis de voz es algo básico en este experimento ya que las pruebas que realizamos con los usuarios deben contener voz hablada que deberá ser transcrita por el usuario. Necesitamos una tecnología que nos permita utilizar algún tipo de voz sintética en la aplicación web.

Se han realizado muchos avances en esta tecnología y hoy en día es posible encontrar sintetizadores de voz que se acercan bastante a las cualidades de la voz humana real. Pudiendo incluso aplicar actitudes o tonos que añaden expresividad casi humana a la voz artificial generada.

Esta síntesis puede llegar a ser muy avanzada pero actualmente no existe sintetizador de voz que tenga la fluidez suficiente como para no ser fácilmente detectado como una voz artificial.



## 5.1.1 Tipos de sintetizadores de voz

Para que la experiencia de uso de la aplicación web sea satisfactoria la aplicación debe responder rápidamente a los órdenes del usuario. El sonido sintetizado debe generarse rápidamente sin necesidad de grandes velocidades de conexión. A la hora de buscar un sintetizador de voz se tendrán en cuenta estas necesidades.

Los distintos tipos de sintetizadores de voz considerados para la prueba son los siguientes:

### **Servidor:**

Realizan una petición a un servidor remoto que es donde se hace la síntesis de la voz y devuelven un archivo de sonido con la voz sintetizada que puede ser reproducida en la aplicación web, un sintetizador de este tipo puede ser la **API TTS de Google**.

### **Ventajas:**

- Se puede realizar una síntesis de voz con más calidad al disponer de los recursos de un servidor para realizarla.
- La página no necesita componentes adicionales a cargar.

### **Desventajas:**

- Depender de un servidor puede hacer que no funcione tu página si el servidor no está en línea
- Se produce un flujo de datos mayor al realizar una petición.

### **Cliente:**

Realizan la síntesis en el propio navegador, sin necesidad de hacer peticiones a servidores remotos, suelen incluir librerías que deben cargarse al cargar la página la primera vez, un sintetizador de este tipo puede ser **Speak.js**.

### **Ventajas:**

- Disponibilidad inmediata y sin depender de elementos externos.
- Mayor control sobre el tiempo de reproducción y el inicio y paro
- Inicio de la reproducción inmediata.



### **Desventajas:**

- Mayor carga de la página al tener que realizar la carga de las librerías de generación de voz.
- Menos calidad de síntesis.

### **Escritorio:**

Se realiza la síntesis en el ordenador donde está alojada la aplicación web, y haciendo uso de determinadas librerías de síntesis de voz se puede generar el archivo de audio y ser reproducido desde la web, un sintetizador de este tipo puede ser **Microsoft .NET Speech Synthesis**.

### **Ventajas:**

- No se depende de ningún agente externo.
- Posible mayor calidad en la síntesis.

### **Desventajas:**

- Mayores tiempos de espera, tiempo de envío de la solicitud al servidor, generación, compresión y descarga del archivo por parte del navegador cliente.

Después de analizar las necesidades y requisitos del experimento se decidió utilizar el sintetizador **Speak.js** que funciona en el cliente/navegador y que pasamos a explicar en detalle en el siguiente punto.

## 5.1.2 Speak.js

Es un sintetizador de voz que está adaptado para poder funcionar mediante el lenguaje Javascript. El proyecto es una adaptación del proyecto de sintetizador de voz de escritorio para Linux y Windows en C++ eSpeak. En el Apéndice A se detalla el funcionamiento y características de la síntesis de voz que realiza eSpeak.

La voz sintetizada por eSpeak usa un método de síntesis por formantes que se refiere a la resonancia vocal característica de la voz humana. Puede no ser tan suave o natural como otros grandes sintetizadores que usan voces humanas grabadas para realizar la síntesis, pero es capaz de ser compatible con varios idiomas en un tamaño muy pequeño.



Figura 10 - Logo de eSpeak

### 5.1.2.1 Emscripten



Figura 11 - Logo de emscripten

El proyecto es de código libre y está realizado gracias al compilador Emscripten, este compilador permite compilar un programa hacia Javascript desde C, C++ y otros lenguajes, de este modo es posible ejecutar este programa en cualquier plataforma donde pueda funcionar Javascript incluida la web.

### 5.1.2.2 Parámetros configurables

Sus características de configuración son las siguientes:

- **Volumen**

Se puede modificar de 0 a 100. De más bajo a más alto.

- **Tono de la voz**

Por defecto 50. Representa lo aguda o grave que será la voz.

- **Velocidad a la que habla la voz sintetizada**

Valor normal 175. Representa las palabras por minuto que pueden ser dichas.

- **Voces en distintos idiomas**

Es configurable la voz en la que habla la voz sintetizada, por defecto en inglés.

- **Espacio entre palabras**

Espacio adicional en unidades de 10ms, por defecto 0. Cada palabra comienza inmediatamente después de la siguiente.

Los captchas que usaremos en el experimento modificarán sus características en gran medida usando distintas configuraciones de estos parámetros.

### 5.1.2.3 Uso

Para poder usarlo deben incluirse en el proyecto las librerías Javascript necesarias y declararse en la cabecera de la página **speakClient.js**

```
<script src="speakClient.js"></script>
```

Para poder reproducir el audio se usa la característica <audio/> de HTML5

La instrucción para reproducir es llamar a la función **speak** seguida de lo que se desea que se pronuncie y las diferentes opciones, si no está definida la opción se usa la configuración por defecto de esa opción.

Un ejemplo de instrucción:

```
speak ( 'hola mundo' , { pitch: 100 } );
```

Se generará el audio "Hola mundo" y se enviará al componente HTML5 audio.

### 5.1.2.4 Configuración de idioma

Por defecto el idioma de speak.js es el inglés, el poder utilizar otro de los idiomas disponibles no es un método sencillo, ya que es necesario compilar el código fuente (que está disponible en la página del proyecto) cambiando en determinados archivos las referencias al idioma inglés por las correspondientes al español.

La compilación requiere del uso de Emscripten que a su vez requiere la instalación de las versiones adecuadas de LLVM with Clang, Node.js y Python.

Una vez realizada la compilación se generará un archivo **speakGenerator.js** de aproximadamente 2 megabytes adaptado al idioma español y que una vez sustituido en el proyecto nos permite hacer pronunciaciones en español utilizando el parámetro **voice**.

```
{ voice: 'es' }
```

### 5.1.2.5 Compatibilidad

Como ya hemos visto el uso de speak.js necesita de la posibilidad de usar etiquetas HTML5 en el navegador del cliente. Afortunadamente la mayoría de navegadores modernos son compatibles con las especificaciones del estándar HTML5. EN la figura siguiente se muestra la compatibilidad actual.

	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser
								2.1	
								2.2	
						3.2		2.3	
						4.0-4.1		3.0	
	8.0	21.0	27.0			4.2-4.3		4.0	
	9.0	22.0	28.0	5.1		5.0-5.1		4.1	7.0
Current	10.0	23.0	29.0	6.0	15.0	6.0-6.1	5.0-7.0	4.2	10.0
Near future	11.0	24.0	30.0	7.0	16.0	7.0			
Farther future		25.0	31.0						

**Figura 12 - Compatibilidad con elementos Audio en navegadores**

La síntesis de la voz que realiza speak.js hace uso de **Arrays tipados**, es una tecnología que permite a Javascript trabajar con datos binarios en bruto (RAW) de una manera mucho más eficiente. Es un requisito en este caso más restrictivo. El mayor problema lo encontramos con las versiones anteriores a la 10 de Microsoft Internet Explorer que no son compatibles. De todas maneras según el estudio consultado ahora mismo es soportado por la totalidad de navegadores a excepción del Opera Mini.



A continuación se muestra una tabla en la que se muestran la compatibilidades de los distintos navegadores respecto a los Arrays tipados.

	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser	Opera Mobile	Chrome for Android	Firefox for Android
24 versions back			4.0									
23 versions back		2.0	5.0									
22 versions back		3.0	6.0									
21 versions back		3.5	7.0									
20 versions back		3.6	8.0									
19 versions back		4.0	9.0									
18 versions back		5.0	10.0									
17 versions back		6.0	11.0									
16 versions back		7.0	12.0									
15 versions back		8.0	13.0									
14 versions back		9.0	14.0									
13 versions back		10.0	15.0									
12 versions back		11.0	16.0									
11 versions back		12.0	17.0		9.0							
10 versions back		13.0	18.0		9.5-9.6							
9 versions back		14.0	19.0		10.0-10.1							
8 versions back		15.0	20.0		10.5							
7 versions back		16.0	21.0		10.6							
6 versions back		17.0	22.0		11.0			2.1		10.0		
5 versions back	5.5	18.0	23.0	3.1	11.1			2.2		11.0		
4 versions back	6.0	19.0	24.0	3.2	11.5	3.2		2.3		11.1		
3 versions back	7.0	20.0	25.0	4.0	11.6	4.0-4.1		3.0		11.5		
2 versions back	8.0	21.0	26.0	5.0	12.0	4.2-4.3		4.0		12.0		
Previous version	9.0	22.0	27.0	5.1	12.1	5.0-5.1		4.1	7.0	12.1		
Current	10.0	23.0	28.0	6.0	15.0	6.0-6.1	5.0-7.0	4.2	10.0	14.0	28.0	23.0
Near future	11.0	24.0	29.0	7.0	16.0	7.0						
Farther future		25.0	30.0									

Figura 13 - Compatibilidad para Arrays Tipados en navegadores

## 5.2 Aplicación web

Para la realización de la aplicación web necesitamos un IDE de desarrollo y una tecnología que nos permita aplicar todas las características que tenemos pensadas para el experimento.

### 5.2.1 ASP.NET



Figura 14 - Logo de ASP.NET

**ASP.NET** es un framework para aplicaciones web desarrollado por Microsoft. Es usado para construir sitios web dinámicos, aplicaciones web y servicios web XML.

Hace uso de la tecnología .NET de Microsoft permitiendo programar en varios lenguajes junto con el IDE Visual Studio. Es sencillo de programar y de publicar en un servidor, y a pesar de ser una tecnología propietaria es ampliamente usada por todo internet y por lo tanto existe gran cantidad de documentación disponible.

Las páginas creadas con ASP.NET tienen extensión .aspx, que contienen etiquetas HTML o XHTML estático, también contienen etiquetas que definen controles que se procesan en el lado del servidor. Además existe para cada página un archivo con la extensión .vb o .cs dependiendo del lenguaje utilizado en la que se programa el código de servidor, se controlan los eventos de los controles, etc...

Para poder usar ASP.NET se debe trabajar con el IDE de Microsoft Visual Studio. En nuestro caso se utiliza la versión 2010.

## 5.2.2 Visual Studio 2010



Figura 15 - Logo de Visual Studio

Es un entorno de desarrollo integrado (IDE) de Microsoft, es compatible con sistemas operativos Windows.

Los lenguajes de programación soportados son Visual C++, Visual C#, Visual J#, y Visual Basic .NET. En este proyecto se usa Visual Basic .NET. Visual Studio permite crear sitios web, servicios web y aplicaciones desde un mismo entorno.

Es por defecto compatible con ASP.NET y sus últimas revisiones, como con las nuevas etiquetas HTML5, con la programación AJAX mediante Callbacks y con la integración con bases de datos como SQL Server.

## 5.2.3 Reproductor de audio web

### 5.2.3.1 HMTL5 <AUDIO>

Etiqueta incluida en el estándar HTML5 que permite reproducir audio dentro de una página sin necesidad de controles externos, dispone de varias formas de controlar la reproducción y algo muy importante para nosotros que es el conocer si la reproducción ha finalizado.

En la siguiente figura se muestran los formatos admitidos en los distintos navegadores.

Navegador	Versión	Formato
Internet Explorer	9.0+	MP3, AAC
Chrome	6.0+	Ogg Vorbis, MP3, WAV†
Firefox	3.6+	Ogg Vorbis, WAV
Safari	5.0+	MP3, AAC, WAV
Opera	10.0+	Ogg Vorbis, WAV

Figura 16 - Formatos de audio compatibles en navegadores



Se optó por el uso del formato .wav para mantener la mayor compatibilidad posible.

En la configuración de los controles se utilizaron los siguientes atributos del control de audio:

- **Loop**

El audio se reproduce y cuando termina vuelve a reproducirse formando un bucle que no termina hasta que se para la reproducción.

- **Preload**

Se realiza una carga en segundo plano del archivo de sonido para no tener que esperar a que se descargue en el momento de la escucha.

### 5.2.3.2 ASP.NET Audio

Es un componente que permite reproducir audio en una página web realizada en ASP.NET. Su uso es sencillo y consiste en un control que debe colocarse en la página, al que se le pueden mandar ordenes mediante Javascript y desde servidor.

Dispone de una API sencilla que permite controlar el fin de la reproducción del fichero y otros parámetros importantes de la reproducción.

Se realizó un prototipo de la aplicación usando este componente, pero más tarde su uso se desestimó al ser un control de pago y cuya versión gratuita incluía pantallas de publicidad.

## 5.2.4 AJAX



Figura 17 - Logo de AJAX

AJAX (Asynchronous JavaScript And XML) (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones con una mayor velocidad de respuesta con el usuario.

El código se ejecuta en el cliente, en el navegador del usuario mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. Se pueden realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página.

## 5.3 Base de datos

### 5.3.1 SQL SERVER



Figura 18 - Logo de Sql Server

SQL Server es sistema para la gestión de base de datos de Microsoft, se complementa a la perfección con ASP.NET y Visual Studio. Y se basa en el modelo de datos relacional.

Gracias a su integración con Visual Studio se pueden realizar conexiones a bases de datos locales o remotas desde la misma aplicación.



## 5.4 Reconocedor de voz

Como parte del experimento debemos hacer pruebas enviando las secuencias de audio que componen cada captcha a un reconocedor automatizado de voz.

### 5.4.1 API Reconocimiento de voz de Google



Figura 19 - Logo de Google

El API de reconocimiento de voz de Google está desarrollado para permitir a un programador web ofrecer a los usuarios un servicio de reconocimiento de voz, para por ejemplo rellenar un formulario. Está diseñada para poder admitir tanto una secuencia breve de voz o secuencias continuas o más largas de voz.

Los resultados del reconocimiento de voz se devuelven al navegador como una lista de hipótesis acompañadas alguna otra información relevante como el tanto por ciento de fiabilidad.

La API se ha añadido al navegador Google Chrome desde febrero de 2013 y está disponible para ser usada.

### 5.4.2 SoX

SoX es una aplicación multiplataforma y por línea de comandos que permite convertir audio entre distintos formatos, es muy configurable y es de código abierto.

Se necesita porque para poder enviar archivos de audio al reconocedor de voz deben estar convertidos al formato FLAC y a una frecuencia de 16000Hz,



### 5.4.3 Wget

Wget es un software libre para Linux y Windows que para enviar o recibir paquetes por HTTP, HTTPS o FTP.

Se utiliza por línea de comandos y es sencillo hacer llamadas mediante scripts.

Para poder hacer la prueba debe tenerse preparado el archivo FLAC que es el que se va a comprobar y ejecutar la siguiente instrucción en la línea de comandos.

Suponiendo que el archivo se llame **12.flac**

```
wget -q -U "Mozilla/5.0" --post-file 12.flac --header="Content-  
Type:          audio/x-flac;          rate=16000" -O -  
"http://www.google.com/speech-api/v1/recognize?lang=es-  
es&client=chromium"> Respuesta12.ret
```

Se realiza una petición al servidor de Google con nuestro archivo y la respuesta de este se almacena en el archivo **Respuesta12.ret**.



## 6 Diseño

### 6.1 Aplicación web del experimento

El diseño de la aplicación ha pasado por diferentes fases. En los distintos diseños se ha intentado que la experiencia de uso fuera sencilla y que los datos recopilados fueran los adecuados para el objetivo del experimento.

#### 6.1.1 Diseño inicial

El diseño consistía en una sola página en la que se mostraban de una vez todos los captchas de la prueba, un captcha individual era como la imagen siguiente.

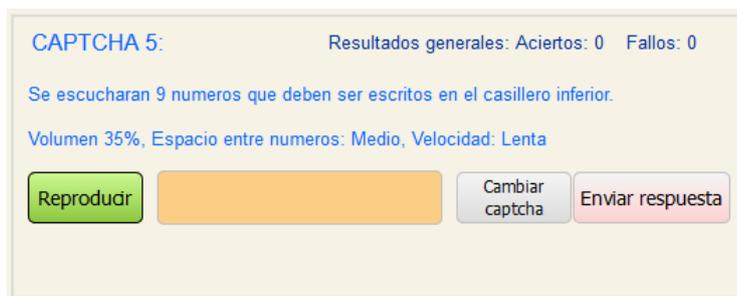


Figura 20 - Diseño inicial de pantalla captcha

Se mostraban varias secciones y comportamientos que fueron eliminados o modificados en la versión final.

#### **Contador global de acierto y fallos:**

Se desestimó porque para el objetivo de la prueba el usuario no necesita saber si se ha fallado mucho o poco en el captcha actual, podría incluso influir en su respuesta.

#### **Descripción de los parámetros del captcha:**

El usuario no debe estar sobre aviso de lo que se va a encontrar al pulsar reproducir, si no le decimos nada el resultado de la prueba será más real.



### **Posibilidad de cambiar el captcha:**

Para una mejor evaluación de los captchas se decidió que los captchas fueran fijos y no se pudiesen cambiar, si se pudiesen cambiar podría darse el caso de que un mismo captcha fuera más sencillo de resolver para un usuario que para otro.

### **Captchas aleatorios:**

Por la razón expuesta anteriormente se optó por mantener todos los captchas con las mismas características para maximizar la igualdad entre las pruebas de distintos usuarios.

### **Autenticación por el nombre:**

La autenticación en un principio era sencilla y solamente incluíamos el nombre como identificador. En la versión final se sustituye por datos más descriptivos del usuario que nos ayudan a que los resultados sean más útiles.

### **Bloqueo de escritura mientras suena el captcha:**

Una serie de captchas tenían como característica el bloqueo de la escritura mientras sonaba el captcha, esto incrementaba su dificultad sobre todo en captchas largos, porque era necesario memorizar, se desestimó porque no era el objetivo del experimento.

Complicar el captcha al usuario humano no nos permite realizar una comparación correcta frente a un reconocedor de voz automático ya que el no poder escribir mientras suena no es un problema para el reconocedor.

Una vez descritos los aspectos que fueron descartados vamos a detallar lo que ha sido el diseño final.

## 6.1.2 Diseño final

Tomando como base el trabajo ya realizado en el diseño inicial se modifica el diseño para adaptarlo de una manera más adecuada al objetivo del experimento.

En el diseño se opta por dirigir al usuario por la prueba, cada captcha va apareciendo una vez se acaba la anterior, explicando al usuario las acciones que debe realizar para que no se pierda.

### 6.1.2.1 Secciones de la web

Analizaremos las distintas ventanas que componen la aplicación comenzamos con el formulario de acceso.

#### 6.1.2.1.1 Formulario de Acceso

El formulario de acceso es la primera página que el usuario puede ver al acceder a la aplicación.

Tiene como funciones principales:

- Identificar al usuario para poder acceder a la prueba.
- Visualizar/Rellenar/Editar los datos de usuario.

Prueba de comprensión de CAPTCHA's sonoros

Acceso a la prueba

Seleccione si es un usuario nuevo o un usuario registrado e introduzca su dirección de correo electrónico en la caja de texto inferior.

Nuevo usuario     Usuario registrado

E-mail:

Acceder

Figura 21 - Formulario de acceso I



Ofrece al usuario de dos opciones:

**1. Nuevo usuario:**

Debe seleccionarse en caso de ser un nuevo usuario que no ha accedido en ningún momento a la aplicación.

**2. Usuario registrado:**

Se usa si es un usuario que vuelve y que ya ha accedido a la aplicación en algún momento. Al seleccionar Usuario Registrado la página ofrece la opción de 'Editar/Ver datos de usuario'.

El usuario una vez seleccionado debe introducir su correo electrónico en la caja de texto para identificarse.

Al seleccionar 'Editar/Ver datos de usuario' se pueden visualizar los datos guardados del usuario. Se puede ver en la imagen siguiente.

Prueba de comprensión de CAPTCHA's sonoros

### Acceso a la prueba

Seleccione si es un usuario nuevo o un usuario registrado e introduzca su dirección de correo electrónico en la caja de texto inferior.

Nuevo usuario  Usuario registrado

E-mail:   Editar/Ver datos de usuario

### Datos de usuario

Por favor rellene la información siguiente, es importante para que posteriormente los datos recogidos sean más útiles.

Profesión:  Edad:

Sexo:  Frecuencia de acceso a internet:

Figura 22 - Formulario de acceso II



El formulario de datos de usuario solicita al usuario la siguiente información.

▪ **Profesión**

Texto libre en el que le usuario puede introducir su profesión.

▪ **Edad**

Lista desplegable en la que se muestran las siguientes opciones.

- Menos de 18.
- 18 a 30.
- 30 a 50.
- Más de 50.

▪ **Sexo**

Sexo del usuario que realiza la prueba.

▪ **Frecuencia de acceso a internet**

Lista desplegable en la que se muestran las siguientes opciones.

- Infrecuente.
- Esporádica.
- Diaria.

Los datos que se solicitan están elegidos para hacernos una idea del tipo de persona que realiza la prueba y más adelante con los resultados poder hacer una distinción de grupos de usuarios que fallan o aciertan más determinadas pruebas.

### 6.1.2.1.2 Ventana de captcha

Esta ventana es la base de la aplicación en ella se realizan las pruebas y el usuario envía sus respuestas.

Consta de una parte superior en la que hay una explicación extensa de lo que se puede hacer en la página y en que consiste la prueba. En total son doce captchas sonoros y desde esta ventana se pasa de unos a otros.

Figura 23 - Ventana de prueba de captcha

Se muestra el número del captcha en el que se encuentra el usuario sobre el total de pruebas para tener un indicador de cuanto queda para el final de la prueba. También se puede ver el número de intentos gastados y disponibles.

Los controles inferiores son.

- **Caja de texto para reproducir el captcha**

Al hacer el usuario clic sobre la caja se comenzará a reproducir el sonido del captcha sonoro una vez finalizado él no se podrá volver a escuchar. El usuario debe escribir lo que cree haber escuchado en la caja de texto.

- **Botón Resolver**

El botón que debe pulsar el usuario para enviar la respuesta al servidor y comprobar si ha acertado o no, solamente se podrá hacer clic en este botón cuando el captcha ya ha sonado.

#### ▪ Botón Pasar

Este botón lleva al usuario a la siguiente prueba, asumiendo que ha abandonado la resolución del captcha anterior.

### 6.1.2.1.3 Ventana de agradecimiento

Una vez finalizada la prueba se redirige al usuario a esta página en la que se le agradece su colaboración y se le propone rellenar un cuestionario con algunas preguntas sobre la prueba.



Figura 24 - Ventana de agradecimiento

Adicionalmente se ofrece al usuario la posibilidad de volver a realizar la prueba con más intentos disponibles para responder los captchas.

### 6.1.2.2 Autenticación de usuarios

En este experimento es muy importante poder identificar al usuario que accede a la aplicación ya que puede que no haga la prueba completa en el momento sino que intente completarla en diferentes momentos. Por lo que la aplicación debe identificar al usuario y permitirle continuar desde donde dejó las pruebas.

Como parte del experimento, cada usuario deberá rellenar en primer lugar un formulario con algunos datos sobre su perfil: profesión, rango de edad, sexo y nivel de uso de internet. Estos serán los únicos datos personales que se obtendrán y en ningún caso se asociarán a otros que permitan identificar unívocamente a la persona en cuestión.



El formulario de datos de usuario se presentará siempre en primer lugar en cada conexión que haga el usuario y podrá modificarse su contenido libremente.

Vamos a utilizar el correo electrónico y la fecha de inicio del experimento como clave única de sesión. El contenido de esta clave combinada se utiliza encriptado para evitar modificaciones y accesos por error.

Las diferentes opciones que se muestran dependiendo si se accede con la clave de sesión incluida en la URL son las siguientes.

### **Sin clave de sesión**

#### ▪ **Usuario nuevo**

Se selecciona si el usuario no ha hecho ninguna prueba anteriormente y desea comenzar a realizar las pruebas. Se envía un correo al usuario para verificar que el correo existe y generar la URL que incluye la clave única de sesión del usuario. Los pasos son los siguientes:

1. La aplicación solicita al usuario su correo electrónico.
2. Al hacer clic en 'acceder' envía un correo electrónico a la dirección indicada.
3. El usuario comprueba su correo y hace clic en la URL incluida en el correo que incluye la clave de sesión.
4. El usuario accede a la página de nuevo, el correo electrónico ha sido verificado y se ha accedido con clave de sesión y al ser un usuario nuevo le muestra el formulario con los datos de usuario.

#### ▪ **Usuario registrado**

El usuario ya ha entrado otras veces y ya existe una clave de sesión asociada a ese correo electrónico.

1. La aplicación solicita al usuario su correo electrónico.
2. El usuario puede decidir si modifica sus datos de usuario o va directamente a la prueba.
3. Si decide modificar sus datos, se le muestra el formulario y se puede modificar.
4. Al hacer clic en 'acceder' se redirige al usuario a la zona de pruebas con su clave de sesión.
5. La aplicación comprueba que existe esa clave de sesión asociada a ese correo y recupera la prueba en el estado que lo dejó el usuario.



### **Con clave de sesión**

1. El sistema comprueba la clave de sesión y muestra los datos almacenados del usuario, en el caso de ser la primera vez muestra los datos por defecto.
2. Al hacer clic en 'acceder' se redirige al usuario a la zona de pruebas con su clave de sesión.
3. La aplicación comprueba que existe esa clave de sesión asociada a ese correo y recupera la prueba en el estado que lo dejó el usuario o comienza una prueba nueva en caso de ser la primera vez.

### 6.1.2.3 Componentes del Captcha Sonoro

La generación del sonido que se puede escuchar en cada uno de los captchas está compuesta por dos partes independientes:

1. Secuencia de números generada por el sintetizador
2. Ruido de fondo que se reproduce en la página.

Las dos componentes sonoras del captcha se reproducen en etiquetas <audio/> y se sincronizan para formar un único sonido. Al comenzar a reproducir el captcha se escuchan los dos sonidos y el sonido de fondo se reproduce en bucle hasta que el audio sintetizado termina.

Los distintos tipos de ruido de fondo son 7, son grabaciones de ruidos reales y por lo tanto no sonidos sintetizados.

### 6.1.2.4 Parámetros a modificar en el captcha

Los parámetros que se modifican en los distintos captchas son:

- **Número de números que componen la secuencia**
  - 4 números.
  - 7 números.
- **Espaciado de tiempo entre los diferentes números hablados**
  - Tiempo muy bajo 1 segundo.
  - Tiempo normal 0 segundos.
- **Velocidad de pronunciación**
  - Lenta 10 palabras por minuto.
  - Normal 175 palabras por minutos.
  - Rápida 275 palabras por minutos.



- **Ruido de fondo**
- **Número de reintentos**

Las distintas configuraciones de parámetros (número de números, velocidad, etc...) no se especifican al usuario para evitar dar pistas sobre lo que se va a encontrar al pulsar reproducir.

Los captchas siempre constarán de la misma configuración de parámetros para todos los usuarios.

Por ejemplo si el captcha nº1 tiene esta configuración, solución 8921, espaciado 1000ms, velocidad normal, para todos los usuarios que entren a la aplicación será igual para evitar que por aleatoriedad el mismo captcha sea más fáciles para unos usuarios y más difíciles para otros.

Las pruebas que mostraremos finalmente serán una combinación de los distintos parámetros. Se pueden ver en la siguiente tabla.

Nº Captcha	Nº de números	Espaciado	Velocidad	Reintentos	
				1ª vez	2ª vez o más
1	4	Normal	Lenta	1	5
2	4	Normal	Normal	1	5
3	4	Normal	Rápida	1	5
4	4	Muy Bajo	Lenta	1	5
5	4	Muy Bajo	Normal	1	5
6	4	Muy Bajo	Rápida	1	5
7	7	Normal	Lenta	1	7
8	7	Normal	Normal	1	7
9	7	Normal	Rápida	1	7
10	7	Muy Bajo	Lenta	1	7
11	7	Muy Bajo	Normal	1	7
12	7	Muy Bajo	Rápida	1	7

**Tabla 2- Parámetros de los captchas sonoros**



## 6.1.2.5 Información recopilada

Las distintas pruebas a las que se van a exponer los usuarios están pensadas para poder sacar toda la información del paso del usuario, la información sería la siguiente.

- **Fallo**

El usuario ha fallado.

- **Éxito**

El usuario ha acertado el captcha.

- **Pasar**

El usuario se ha dado por vencido y ha dado al botón de pasar captcha.

- **Fecha**

Fecha del intento.

- **Número de intentos**

Número del intento actual.

- **Número de reintentos máximos**

Número de reintentos disponibles para ese captcha.

Las opciones que se ofrecen al usuario en detalle son:

- **Escuchar**

Se reproduce el captcha, el usuario sólo puede reproducir el captcha una vez.

- **Resolver el captcha**

- **Éxito:**

- Se pasa al siguiente captcha.

- **Fallo:**

- 1. Si hay intentos disponibles se mantiene el mismo captcha, hasta terminar intentos.
    - 2. Si no hay intentos disponibles se notifica del fallo y se pasa al siguiente captcha.

- **Pasar**

- Se pasa al siguiente captcha, abandonando el captcha actual.





## 6.2 Base de datos

En la base de datos que hemos utilizado hemos creado el esquema que se puede ver en la siguiente página.

El esquema se compone de las siguientes tablas:

### Usuarios

Esta tabla representa la entidad usuario de la aplicación, en ella se guardan los distintos atributos que deseamos guardar de cada usuario registrado.

- **IdUsuario:** Auto numérico identificador único.
- **Correo:** Correo electrónico del usuario.
- **RangoEdad:** Rango de edad del usuario.
- **Sexo:** Sexo del usuario (0 Hombre, 1 Mujer).
- **Profesion:** Profesión de usuario.
- **FrecuenciaUsoInternet:** Nivel de familiarización con internet del usuario.
- **FechaInicioPrueba:** Fecha en el que el usuario se registró.
- **uToken:** Conjunto de caracteres alfanuméricos codificados que identifican la sesión del usuario.

### TiposFrecuenciaUso

Lista de elementos disponibles en las opciones de frecuencia de uso de internet.

- **IdFrecuenciaUso:** Auto numérico identificador único.
- **Nombre:** Nombre de la opción.

### TiposRangosEdad

Lista de elementos disponibles en las opciones de rangos edad.

- **IdRangoEdad:** Auto numérico identificador único.
- **Nombre:** Nombre de la opción.



## Captchas

Esta tabla representa cada uno de los captchas que se muestran en el experimento y contiene todas sus características.

- **IdCaptcha:** Auto numérico identificador único.
- **Texto:** Texto que contiene los números que deberá hablar el sintetizador de VOZ.
- **ArchivoRuido:** Nombre del archivo de voz que se reproduce al mismo tiempo que suenan los números.
- **Espaciado:** Tiempo en milisegundos entre los distintos números.
- **Velocidad:** Velocidad de pronunciación de los distintos números.
- **ReintentosEnSegundaVuelta:** Número de intentos disponibles a partir de la segunda vez que se realiza la prueba.

## EstadoUsuario

Esta tabla se encarga de almacenar el ultimo estado del usuario para cada captcha, de este modo consultando esta tabla se puede conocer en qué situación se encuentra el usuario y a que captcha dirigirle para continuar la prueba.

- **IdEstadoUsuario:** Auto numérico identificador único.
- **IdUsuario:** Identificador del usuario actual.
- **IdCaptcha:** Identificador del captcha actual.
- **NumVuelta:** Número de intento de la prueba, la primera vez se restringen los intentos de los captchas.
- **NumReintentosDisponibles:** Intentos disponibles en el captcha actual.
- **NumReintentosAgotados:** Intentos agotados del captcha actual.
- **CaptchaResuelto:** Indicador de si se ha resuelto o no el captcha.

## RespuestasUsuario

Esta tabla se inserta un nuevo registro cada que se recibe una respuesta del usuario, fallos, aciertos, pasos se registran aquí.

- **IdRespuestaUsuario:** Auto numérico identificador único.
- **IdUsuario:** Identificador del usuario actual.
- **IdCaptcha:** Identificador del captcha actual.
- **CadenaEnviada:** Texto que ha introducido el usuario como respuesta.
- **CadenaCorrecta:** Texto correcto del captcha actual.



- **Intento:** Intento en el que se produce la respuesta.
- **IntentosMaximos:** Intentos máximos permitidos en el captcha actual.
- **Fecha:** Fecha en la que se produce la respuesta.
- **Resultado:** Identificador de los tres posibles resultados, Fallo, Acierto, Paso.

### TiposResultados

Lista de elementos disponibles en las opciones de resultados.

- **IdTipoResultado:** Auto numérico identificador único.
- **Nombre:** Nombre de la opción.

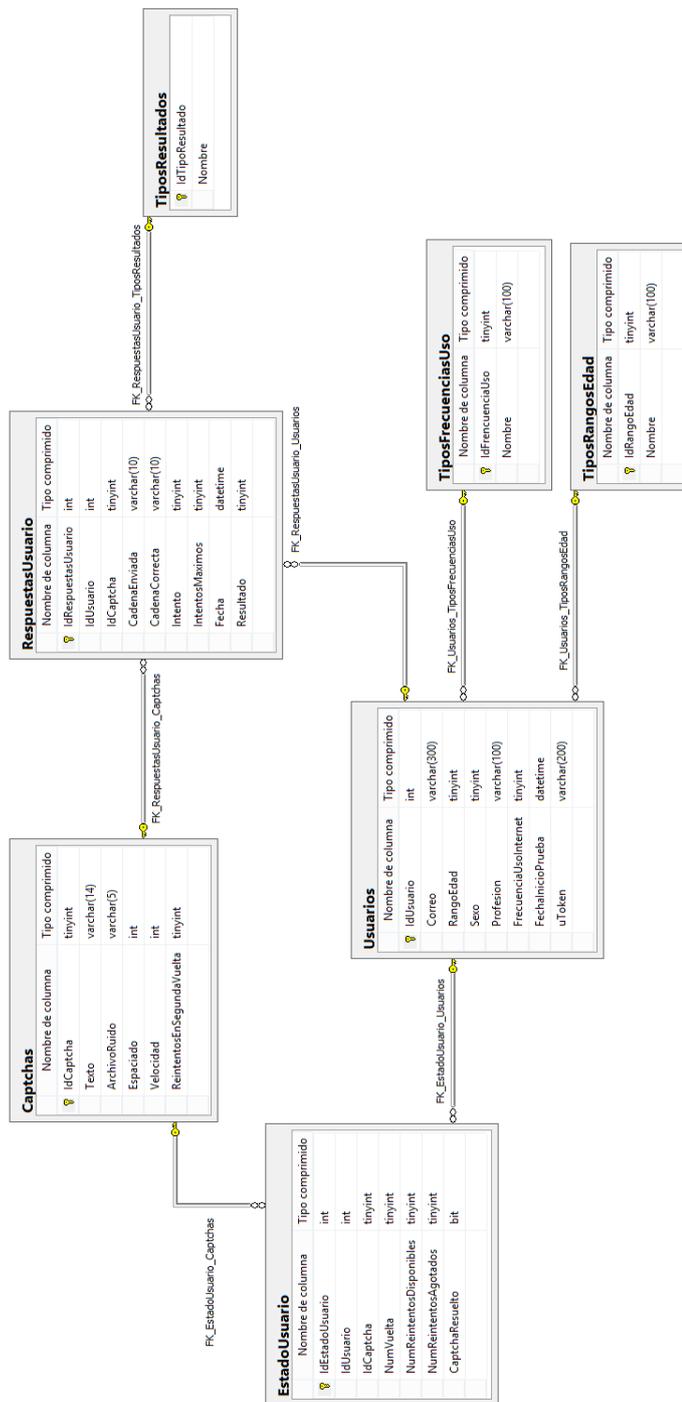


Figura 25 - Esquema de tablas base de datos



# 7 Experimento y pruebas

## 7.1 Mecánica del experimento

Una vez se dispone de la aplicación web se procede a realizar el experimento y a comenzar a recoger resultados.

### 7.1.1 Envío de propuestas

En todo experimento deben existir usuarios que sean los que realicen las pruebas, para eso debemos conseguir la mayor participación posible, haciendo que el mayor número de personas realice las pruebas que tenemos preparadas.

Cada usuario debe disponer de una clave única que lo identificará, esta clave actúa como una variable de sesión y será la que controle los datos tanto de perfil del donante como los resultados de su evaluación. El acceso a la página puede hacerse con esta clave incluida en la URL y de este modo se autentificará automáticamente el usuario.

Disponemos de dos maneras de compartir esta clave única y comunicar la existencia de la prueba e invitar a su realización.

#### 1. Invitaciones por correo electrónico.

Se envían correos de invitación a usuarios (contactos personales del autor, amigos) en los que aparte de una explicación de lo que se va a encontrar si accede a la página y en que consiste la prueba, se incluye una URL con el acceso a la página que puede incluir la clave de sesión.

#### 2. Accesos directos a página de registro.

Se distribuirá por redes sociales la dirección URL de la página de registro, el usuario que desee participar deberá introducir exclusivamente su dirección de correo electrónico, a la que se enviará un mensaje de correo de verificación en el que se le asignará la clave de sesión.

## 7.2 Mecánica de las pruebas

Una vez está el usuario en la zona de pruebas, la mecánica y pasos a realizar son los siguientes.

1. Se muestra al usuario una pantalla que incluye una descripción de las acciones que puede realizar a modo de instrucciones y una zona inferior que muestra el captcha en el que se encuentra respecto al total de 12.

Captcha Sonoro: 2 /12

Intentos disponibles/gastados: 1 / 0

Haga clic en la caja para escuchar --->

Se muestran también los intentos disponibles y gastados, los intentos son las posibilidades de respuesta si llegan a cero no habrá más opción que ir al siguiente captcha.

2. Al hacer clic el usuario dentro de la caja de texto sonará una secuencia de números con una voz distorsionada y un ruido de fondo. Una vez suena la secuencia no se puede volver a reproducir.
3. El usuario debe transcribir en la caja de texto la secuencia que ha reconocido, puede escribir mientras suena el captcha o al final.
4. Una vez ha sonado el captcha puede hacer clic en 'Resolver' o en 'Pasar'.
5. Cada intento resta un intento del contador de intentos del captcha.
6. Al pulsar el botón 'Resolver' se muestra si se ha detectado la secuencia correcta o no y se pasa a la siguiente pantalla, con otro captcha.
7. Al pulsar el botón 'Pasar' se pasa a la siguiente pantalla, con otro captcha. No pudiendo volver al captcha anterior.
8. Una vez finalizadas las 12 pruebas se agradece al usuario su tiempo e interés y se le ofrece rellenar una encuesta sobre la prueba.

Se va a permitir al usuario realizar la prueba completa cuantas veces quiera la primera vez solamente se puede intentar una vez cada captcha pero a partir de la segunda se pueden realizar más intentos de respuesta.



A medida que se complete cada uno de los pasos de reconocimiento o salto de un captcha, quedará registrado en la base de datos de forma que si el usuario abandona temporalmente la misma, cuando regrese seguirá en el paso que le falta.

## 7.2.1 Pruebas con el reconocedor automático

La mecánica es similar la realizada con las pruebas para los usuarios humanos, se extrae el sonido de cada uno de los captchas a archivos separados de audio en el formato adecuado con el programa SoX y lo enviaremos al servidor de reconocimiento de voz de Google mediante el programa Wget que nos devolverá lo que el reconocedor a entendido

Se almacenan los resultados de cada captcha para luego analizarlos.





## 8 Despliegue y puesta en marcha

Con la aplicación web y la base de datos preparada se planea la puesta en marcha y el comienzo de la recopilación de datos.

Las fases de las que se compone son las siguientes:

### **Publicación de la web y de la base de datos en un servidor público.**

La web es accesible para cualquier usuario desde cualquier ordenador o dispositivo con acceso a internet.

### **Difusión de la dirección de la web a distintos objetos de prueba.**

Se envía la dirección a distintas personas para que resuelvan la prueba y recopilar datos, las personas deben ser heterogéneas, para asegurarnos de que la prueba es lo más precisa posible, se deben buscar personas en las que difieran al menos los siguientes parámetros:

- Conocimientos de informática.
- Edad.

### **Espera a recolección de resultados**

Fase en la que se debe esperar a tener la información suficiente para poder analizar los resultados, se deben recolectar el mayor número de resultados posible.

## 8.1 Servidor web

### 8.1.1 Somee.com



Figura 26 - Logo de SOMEE.COM

Se necesitaba un servidor en el que colgar la aplicación web que permitiera la utilización de ASP.NET, SQL Server y ejecución de Javascript.

La página web Somee.com funciona correctamente a la hora de hacer la función de servidor, la interfaz es adecuada y las limitaciones por ser un usuario gratuito son aceptables.

Una pega es que hay que convertir las fechas del huso horario, hay una diferencia de 7 horas entre las fechas almacenadas y su equivalente en España.

## 8.2 Servicio de Correo

### 8.2.1 Yahoo



La aplicación manda correos de confirmación de la dirección de correo electrónico, el servidor utilizado es el servicio de correo de Yahoo, con una cuenta creada en dicho servicio desde la que se mandan los correos.

En un primer momento se intentó con el servicio de Google, pero tenía abundantes restricciones a la hora de mandar determinado número de correos desde la misma cuenta llegando incluso al bloqueo temporal de la misma por posible abuso o spam.



## 8.3 Mensajes de invitación

### 8.3.1 Correo electrónico

El correo de invitación que se envió a los usuarios es el siguiente:

*¡Hola! Soy Javier y estoy haciendo un experimento como parte de mi trabajo de fin de grado.*

*He preparado una pequeña prueba que nos va a permitir conocer si los captchas sonoros son útiles para diferenciar entre humanos y maquinas.*

*Pero necesito tu ayuda. No tardaras más de 5 minutos en realizar el experimento.*

*Consiste en 12 pruebas en las que suenan unos números con una voz distorsionada y se debe escribir lo que se oye.*

*No pasa nada si no entiendes lo que ha sonado, puedes pasar o resolver, cualquier respuesta nos ayudará a recopilar información sobre los captchas sonoros.*

*Para acceder a la prueba debes ir a:*

<http://captchasonorostfg.somee.com/>

*Puedes hacer clic sobre el vínculo o copiarlo en la barra de direcciones.*

*Si les mandas este mismo correo a conocidos tuyos estaría genial, cuanta más gente haga la prueba mejor.*

*¡¡Muchas gracias por tu ayuda!!*



## 8.3.2 Redes sociales

El mensaje difundido por las redes sociales ha sido el siguiente:

*Estoy a punto de presentar mi trabajo fin de grado de la universidad.  
Es una prueba de comprensión de captchas sonoros, se hace desde esta página.*

<http://captchasonorostfg.somee.com/>

*¡Asegúrate de tener los altavoces encendidos!*

*Necesito que lo haga el mayor número de gente posible.  
¡Si puedes compartirlo con tus amigos sería genial!*



## 8.4 Encuesta de opinión

Al finalizar la prueba se ofrece al usuario la opción de responder un cuestionario alojado en Google Drive.

Las preguntas a responder en el cuestionario son las siguientes:

### **¿Le ha parecido interesante la prueba?**

- Si
- No
- No lo se

### **¿Qué parte le ha parecido más complicada a la hora de entender los CAPTCHA's?**

Respuesta en texto.

### **¿Cree que estas pruebas son lo suficientemente complicadas como para no poder resolverse por un sistema automático?**

- Si
- No
- No lo se
- Otra (Respuesta en texto)

### **¿Alguna sugerencia para mejorar el test?**

Respuesta en texto.





# Resultados de la prueba

Una vez pasado el tiempo de pruebas un total de 44 personas han realizado la prueba y llega el momento de analizar los resultados.

Los resultados están vinculados al usuario, por lo que en cada prueba sabremos toda la información de la actividad del usuario.

## Resultados de pruebas con usuarios

Los resultados del experimento después de recibir los datos de los usuarios son los que se detallan y agrupan en los puntos siguientes.

### Resultado general de la prueba

Contabilizando los fallos, aciertos y pasar totales, llegamos a una tasa de acierto en la prueba del 55,56%

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,  
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,  
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar, CAST(SUM(CASE WHEN  
Resultado = 1 THEN 1 ELSE 0 END) * 100 / CAST(COUNT(Resultado)  
AS decimal) AS decimal(10, 2)) AS Error  
FROM RespuestasUsuario
```

Fallos	Aciertos	Pasar	% Acierto
230	335	38	55,56

Tabla 3 - Resultados generales de la prueba



## Número de fallos, aciertos, pasar por captcha individual

Vamos a comprobar el número de respuestas erróneas, acertadas y el número de veces que los usuarios han dado a pasar. Agrupado por cada captcha para hacernos una idea de la dificultad de cada uno.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT IdCaptcha,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,
CAST(SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) * 100 / CAST(COUNT(Resultado)
AS decimal(10, 2)) AS Porcentaje
FROM RespuestasUsuario
GROUP BY IdCaptcha
```

Nº	Texto	Espaciado/Velocidad	Fallos	Aciertos	Pasar	% Acierto
1	7839	Normal/Lenta	7	40	3	80,00
2	0744	Normal/Normal	21	25	6	48,08
3	6102	Normal/Rápida	41	10	6	17,54
4	3667	Muy Bajo/Lenta	3	43	2	89,58
5	8316	Muy Bajo/Normal	9	37	2	77,08
6	5921	Muy Bajo/Rápida	4	42	1	89,36
7	8237401	Normal/Lenta	12	34	2	70,83
8	3822364	Normal/Normal	44	11	3	18,97
9	8074598	Normal/Rápida	44	4	9	7,02
10	9876230	Muy Bajo/Lenta	13	30	0	69,77
11	0876233	Muy Bajo/Normal	19	28	2	57,14
12	5213603	Muy Bajo/Rápida	13	31	2	67,39

Tabla 4 - Resultados agrupados por captcha



Se puede observar que poca gente usa la opción de pasar ya que es un porcentaje muy bajo de todas las respuestas.

Se observa una gran concentración de fallos en los captchas siguientes:

- Captcha 2: Principalmente es por la confusión del número 0 por 5.
- Captcha 3: Principalmente es por la confusión del número 0 por 5.
- Captcha 8: Velocidad y número de números.
- Captcha 9: Velocidad y número de números.

Los fallos en los captchas que incluyen un 0 son significativos, la confusión de 0 por 5 es generalizada.

Respecto a los fallos en los captchas 8 y 9 son esperables ya que son los dos más difíciles, incluyen una secuencia de 7 números, con espaciado normal y velocidad de pronunciación alta. Hacen necesario entender rápidamente la secuencia y memorizarla.

## Resultados agrupados por rango de edad

Vamos a diferenciar los grupos de edad para intentar ver si hay alguna diferencia en el nivel de aciertos, fallos.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT RespuestasUsuario.IdCaptcha, Usuarios.RangoEdad,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar
FROM RespuestasUsuario INNER JOIN Usuarios ON RespuestasUsuario.IdUsuario =
Usuarios.IdUsuario
GROUP BY RespuestasUsuario.IdCaptcha, Usuarios.RangoEdad
ORDER BY RespuestasUsuario.IdCaptcha
```

Nº	Espaciado/Velocidad	Rango edad	Fallos	Aciertos	Pasar
1	Normal/Lenta	0 a 18	0	2	0
		18 a 30	0	10	0
		30 a 50	5	20	2
		> 50	2	8	1
2	Normal/Normal	0 a 18	1	1	0
		18 a 30	2	7	1



		30 a 50	10	15	3
		> 50	8	2	2
<b>3</b>	Normal/Rápida	0 a 18	1	1	0
		18 a 30	9	4	0
		30 a 50	24	3	3
		> 50	7	2	3
<b>4</b>	Muy Bajo/Lenta	0 a 18	0	2	0
		18 a 30	0	10	0
		30 a 50	2	22	2
		> 50	1	9	0
<b>5</b>	Muy Bajo/Normal	0 a 18	1	1	0
		18 a 30	0	10	0
		30 a 50	6	19	1
		> 50	2	7	1
<b>6</b>	Muy Bajo/Rápida	0 a 18	1	1	0
		18 a 30	0	10	0
		30 a 50	0	24	1
		> 50	3	7	0
<b>7</b>	Normal/Lenta	0 a 18	1	1	0
		18 a 30	2	9	0
		30 a 50	7	17	1
		> 50	2	7	1
<b>8</b>	Normal/Normal	0 a 18	2	0	0
		18 a 30	10	3	1
		30 a 50	25	5	2
		> 50	7	3	0
<b>9</b>	Normal/Rápida	0 a 18	2	0	0
		18 a 30	9	0	3
		30 a 50	19	3	5
		> 50	14	1	1
<b>10</b>	Muy Bajo/Lenta	0 a 18	0	1	0
		18 a 30	4	5	0
		30 a 50	6	17	0
		> 50	3	7	0
<b>11</b>	Muy Bajo/Normal	0 a 18	0	1	0
		18 a 30	4	5	1
		30 a 50	11	16	1
		> 50	4	6	0



<b>12</b>	<b>Muy Bajo/Rápida</b>	<b>0 a 18</b>	<b>0</b>	<b>1</b>	<b>0</b>
		<b>18 a 30</b>	<b>5</b>	<b>6</b>	<b>0</b>
		<b>30 a 50</b>	<b>7</b>	<b>16</b>	<b>1</b>
		<b>&gt; 50</b>	<b>1</b>	<b>8</b>	<b>1</b>

**Tabla 5 - Resultados agrupados por captcha y por rango edad**

A continuación se muestran los resultados generales por rango de edad.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT Usuarios.RangoEdad,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 /
CAST(COUNT(RespuestasUsuario_1.IdRespuestasUsuario) AS decimal) AS decimal(10, 2))
AS Error, CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) *
100 / CAST
((SELECT COUNT(IdRespuestasUsuario) AS Expr1
FROM RespuestasUsuario
WHERE (Resultado IN (0, 2))) AS decimal) AS
decimal(10, 2)) AS ErrorTotal,
CAST(COUNT(RespuestasUsuario_1.IdRespuestasUsuario)
* 100 / CAST
((SELECT COUNT(IdRespuestasUsuario) AS Expr1
FROM RespuestasUsuario AS
RespuestasUsuario_2
WHERE (Resultado IN (0, 1, 2))) AS decimal)
AS decimal(10, 2)) AS Participacion
FROM RespuestasUsuario AS RespuestasUsuario_1 INNER JOIN
Usuarios ON RespuestasUsuario_1.IdUsuario = Usuarios.IdUsuario
GROUP BY Usuarios.RangoEdad
```

Rango edad	Fallos	Aciertos	Pasar	% error	% error total	% Participación
0 a 18	9	12	0	42,86	3,36	3,48
18 a 30	45	79	6	39,23	19,03	21,56
30 a 50	122	177	22	44,86	53,73	53,23
> 50	54	67	10	48,85	23,88	21,72

**Tabla 6 - Resultados agrupados por rango edad**



## Resultados agrupados por frecuencia de acceso a internet

Vamos a diferenciar los grupos de edad para intentar ver si hay alguna diferencia en el nivel de aciertos, fallos.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT
RespuestasUsuario.IdCaptcha, Usuarios.FrecuenciaUsoInternet,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos, SUM(CASE WHEN Resultado
= 2 THEN 1 ELSE 0 END) AS Pasar
FROM RespuestasUsuario INNER JOIN Usuarios ON RespuestasUsuario.IdUsuario =
Usuarios.IdUsuario
GROUP BY RespuestasUsuario.IdCaptcha, Usuarios.FrecuenciaUsoInternet
ORDER BY RespuestasUsuario.IdCaptcha
```

Nº	Espaciado/Velocidad	Frecuencia	Fallos	Aciertos	Pasar
1	Normal/Lenta	Infrecuente	1	2	0
		Esporádica	0	6	0
		Diaria	6	32	3
2	Normal/Normal	Infrecuente	2	1	0
		Esporádica	4	3	1
		Diaria	15	21	5
3	Normal/Rápida	Infrecuente	2	1	0
		Esporádica	5	2	1
		Diaria	34	7	5
4	Muy Bajo/Lenta	Infrecuente	0	3	0
		Esporádica	1	5	1
		Diaria	2	35	1
5	Muy Bajo/Normal	Infrecuente	1	1	1
		Esporádica	2	4	0
		Diaria	6	32	1
6	Muy Bajo/Rápida	Infrecuente	2	1	0
		Esporádica	0	6	0
		Diaria	2	35	1
7	Normal/Lenta	Infrecuente	2	1	0
		Esporádica	3	3	1
		Diaria	7	30	1



<b>8</b>	Normal/Normal	Infrecuente	3	0	0
		Esporádica	6	0	1
		Diaria	35	11	2
<b>9</b>	Normal/Rápida	Infrecuente	1	0	2
		Esporádica	4	1	1
		Diaria	39	3	6
<b>10</b>	Muy Bajo/Lenta	Infrecuente	1	1	0
		Esporádica	1	5	0
		Diaria	11	24	0
<b>11</b>	Muy Bajo/Normal	Infrecuente	1	1	0
		Esporádica	2	4	0
		Diaria	16	23	2
<b>12</b>	Muy Bajo/Rápida	Infrecuente	0	1	1
		Esporádica	3	3	1
		Diaria	10	27	0

**Tabla 7 - Resultados Captchas por agrupados por captcha y frecuencia uso internet**

A continuación se muestran los resultados generales por frecuencia de uso de internet.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT Usuarios.FrecuenciaUsoInternet,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 /
CAST(COUNT(RespuestasUsuario_1.IdRespuestasUsuario) AS decimal) AS decimal(10, 2))
AS Error,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 / CAST
((SELECT COUNT(IdRespuestasUsuario) AS Expr1
FROM RespuestasUsuario
WHERE (Resultado IN (0, 2))) AS decimal) AS
decimal(10, 2)) AS ErrorTotal,
CAST(COUNT(RespuestasUsuario_1.IdRespuestasUsuario)
* 100 / CAST
((SELECT COUNT(IdRespuestasUsuario) AS Expr1
FROM RespuestasUsuario AS
RespuestasUsuario_2
WHERE (Resultado IN (0, 1, 2))) AS decimal)
AS decimal(10, 2)) AS Participacion
FROM RespuestasUsuario AS RespuestasUsuario_1 INNER JOIN
Usuarios ON RespuestasUsuario_1.IdUsuario = Usuarios.IdUsuario
GROUP BY Usuarios.FrecuenciaUsoInternet
```



Frecuencia	Fallos	Aciertos	Pasar	% error	% error total	% participación
Infrecuente	16	13	4	60,61	7,46	5,47
Esporádica	31	42	7	47,50	14,18	13,27
Diaria	183	280	27	42,86	78,36	81,26

Tabla 8 - Resultados agrupados por frecuencia uso internet

## Resultados agrupados por sexo

Vamos a diferenciar los grupos de edad para intentar ver si hay alguna diferencia en el nivel de aciertos, fallos.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT
  RespuestasUsuario.IdCaptcha, Usuarios.Sexo,
  SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
  SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos, SUM(CASE WHEN Resultado
  = 2 THEN 1 ELSE 0 END) AS Pasar
FROM RespuestasUsuario INNER JOIN Usuarios ON RespuestasUsuario.IdUsuario =
  Usuarios.IdUsuario
GROUP BY RespuestasUsuario.IdCaptcha, Usuarios.Sexo
ORDER BY RespuestasUsuario.IdCaptcha
```

Nº	Espaciado/Velocidad	Sexo	Fallos	Aciertos	Pasar
1	Normal/Lenta	Hombre	2	22	2
		Mujer	5	18	1
2	Normal/Normal	Hombre	8	14	4
		Mujer	13	11	2
3	Normal/Rápida	Hombre	19	6	3
		Mujer	22	4	3
4	Muy Bajo/Lenta	Hombre	2	21	1
		Mujer	1	22	1
5	Muy Bajo/Normal	Hombre	7	15	2
		Mujer	2	22	0
6	Muy Bajo/Rápida	Hombre	3	20	1
		Mujer	1	22	0
7	Normal/Lenta	Hombre	7	16	1
		Mujer	5	18	1
8	Normal/Normal	Hombre	19	7	1



		Mujer	25	4	2
<b>9</b>	Normal/Rápida	Hombre	27	0	4
		Mujer	17	4	5
<b>10</b>	Muy Bajo/Lenta	Hombre	4	17	0
		Mujer	9	13	0
<b>11</b>	Muy Bajo/Normal	Hombre	6	15	1
		Mujer	13	13	1
<b>12</b>	Muy Bajo/Rápida	Hombre	6	16	1
		Mujer	7	15	1

**Tabla 9 - Resultados agrupados por captcha y sexo**

A continuación se muestran los resultados generales por sexo.

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT Usuarios.Sexo,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 /
CAST(COUNT(RespuestasUsuario_1.IdRespuestasUsuario) AS decimal) AS decimal(10, 2))
AS Error,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 / CAST
((SELECT COUNT(IdRespuestasUsuario) AS Expr1
FROM RespuestasUsuario
WHERE (Resultado IN (0, 2))) AS decimal) AS
decimal(10, 2)) AS ErrorTotal,
CAST(COUNT(RespuestasUsuario_1.IdRespuestasUsuario)
* 100 / CAST
((SELECT COUNT(IdRespuestasUsuario) AS Expr1
FROM RespuestasUsuario AS
RespuestasUsuario_2
WHERE (Resultado IN (0, 1, 2))) AS decimal)
AS decimal(10, 2)) AS Participacion
FROM RespuestasUsuario AS RespuestasUsuario_1 INNER JOIN
Usuarios ON RespuestasUsuario_1.IdUsuario = Usuarios.IdUsuario
GROUP BY Usuarios.Sexo
```

Sexo	Fallos	Aciertos	Pasar	% error	% error total	% participación
Hombre	110	169	21	43,67	48,88	49,75
Mujer	120	166	17	45,21	51,12	50,25

**Tabla 10 - Resultados agrupados por sexo**



## Resultados por número de intentos

No hay una gran cantidad de gente que realizará la prueba una segunda vez para reintentar los captchas, aquí están los resultados.

```
SELECT NumReintentosDisponibles, NumReintentosAgotados, NumVuelta, IdCaptcha,  
IdUsuario, CaptchaResuelto  
FROM EstadoUsuario  
WHERE (NumReintentosDisponibles = 0) AND (NumReintentosAgotados > 1)
```

Captcha no resueltos por algún usuario aun agotando todos los intentos disponibles.

Nº	Texto	Espaciado/Velocidad
3	6102	Normal/Rápida
8	3822364	Normal/Normal
9	8074598	Normal/Rápida



## Resultados agrupados por parámetros

Vamos a comprobar que parámetro de configuración del captcha es el más relevante a la hora de provocar un error.

### Numero de números

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT DATALENGTH(Captchas.Texto) AS Expr1,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 /
CAST(COUNT(Resultado) AS decimal) AS decimal(10, 2)) AS Error
FROM RespuestasUsuario INNER JOIN
Captchas ON RespuestasUsuario.IdCaptcha =
Captchas.IdCaptcha
GROUP BY DATALENGTH(Captchas.Texto)
```

Longitud	Fallos	Aciertos	Pasar	% error
4	85	197	20	34,77
7	145	138	18	54,15

**Tabla 11 - Porcentaje de errores por número de números**

### Espacio entre palabras

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT Captchas.Espaciado AS Expr1,
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 /
CAST(COUNT(Resultado) AS decimal) AS decimal(10, 2)) AS Error
FROM RespuestasUsuario INNER JOIN
Captchas ON RespuestasUsuario.IdCaptcha =
Captchas.IdCaptcha
GROUP BY Captchas.Espaciado
```

Espaciado	Fallos	Aciertos	Pasar	% error
Muy bajo	61	211	9	24,91
Normal	169	124	29	61,49

**Tabla 12 - Porcentaje de errores por espacio entre palabras**



## Velocidad

La consulta ejecutada para sacar este listado es la siguiente:

```
SELECT Captchas.Velocidad AS Expr1,  
SUM(CASE WHEN Resultado = 0 THEN 1 ELSE 0 END) AS Fallos,  
SUM(CASE WHEN Resultado = 1 THEN 1 ELSE 0 END) AS Aciertos,  
SUM(CASE WHEN Resultado = 2 THEN 1 ELSE 0 END) AS Pasar,  
CAST(SUM(CASE WHEN Resultado = 2 OR Resultado = 0 THEN 1 ELSE 0 END) * 100 /  
CAST(COUNT(Resultado) AS decimal) AS decimal(10, 2)) AS Error  
FROM RespuestasUsuario INNER JOIN  
Captchas ON RespuestasUsuario.IdCaptcha =  
Captchas.IdCaptcha  
GROUP BY Captchas.Velocidad
```

Velocidad	Fallos	Aciertos	Pasar	% error
Lenta	35	147	7	22,22
Normal	93	101	13	51,21
Rápida	102	87	18	57,97

**Tabla 13 - Porcentaje de errores por velocidad**



## Resultados de pruebas con reconocedor automático de Google

Las pruebas con el reconocedor han sido bastante inesperadas.  
Ha sido muy poca la comprensión de los números hablados en los captchas.

A continuación mostramos una tabla en la que se muestran los resultados.

Nº	Texto	Espaciado/Velocidad	Respuesta (% confianza)	Correcto
1	7839	Normal/Lenta	'8' (0,51%)	<b>NO</b>
2	0744	Normal/Normal	NADA	<b>NO</b>
3	6102	Normal/Rápida	NADA	<b>NO</b>
4	3667	Muy Bajo/Lenta	'go go guau guau' (0,67%)	<b>NO</b>
5	8316	Muy Bajo/Normal	NADA	<b>NO</b>
6	5921	Muy Bajo/Rápida	NADA	<b>NO</b>
7	8237401	Normal/Lenta	NADA	<b>NO</b>
8	3822364	Normal/Normal	NADA	<b>NO</b>
9	8074598	Normal/Rápida	NADA	<b>NO</b>
10	9876230	Muy Bajo/Lenta	'go go go go go go go guau' (0,52%)	<b>NO</b>
11	0876233	Muy Bajo/Normal	NADA	<b>NO</b>
12	5213603	Muy Bajo/Rápida	NADA	<b>NO</b>

Tabla 14 - Resultados comprensión sintetizador de voz



Ante la falta de reconocimiento se ha hecho la misma prueba con los captchas sin ruido de fondo. Los sonidos limpios de los números generados por el sintetizador.

Nº	Texto	Espaciado/Velocidad	Respuesta (% confianza)	Correcto
1	7839	Normal/Lenta	'7 ó 8 web' (0,33%)	NO
2	0744	Normal/Normal	'cuatro cuatro' (0,19%)	NO
3	6102	Normal/Rápida	NADA	NO
4	3667	Muy Bajo/Lenta	'he is he is 7' (0,46%)	NO
5	8316	Muy Bajo/Normal	'1' (0,39%)	NO
6	5921	Muy Bajo/Rápida	NADA	NO
7	8237401	Normal/Lenta	'8 los siete 4 en 1' (0,42%)	NO
8	3822364	Normal/Normal	'8 por 4' (0,28%)	NO
9	8074598	Normal/Rápida	NADA	NO
10	9876230	Muy Bajo/Lenta	NADA	NO
11	0876233	Muy Bajo/Normal	'8' (0,50%)	NO
12	5213603	Muy Bajo/Rápida	NADA	NO

Tabla 15 - Resultados reconecedor de voz (SIN RUIDO)



## Resultados de la encuesta

Las preguntas a responder en el cuestionario son las siguientes:

### ¿Le ha parecido interesante la prueba?

Todos los usuarios han respondido que **SI** les ha parecido interesante la prueba.

### ¿Qué parte le ha parecido más complicada a la hora de entender los CAPTCHA's?

Hay multitud de respuestas, pero las más repetidas son:

- La rapidez o velocidad al decir los números.
- La pronunciación del número 0 que ha sido bastante confundida por 5.

### ¿Cree que estas pruebas son lo suficientemente complicadas como para no poder resolverse por un sistema automático?

Hay disparidad de respuestas en esta pregunta, pero la balanza se inclina hacia los que piensan que un **reconocedor de voz automático SI será capaz de entender** la secuencia de números.

9 respuestas fueron SI contra 7 que fueron NO.

### ¿Alguna sugerencia para mejorar el test?

En general las respuestas han sugerido lo siguiente:

- Permitir reproducir el captcha más de una vez.
- Tener más reintentos para la respuesta en la primera vez que se hace la prueba.

Todas las respuestas escritas por los usuarios pueden consultarse en los anexos de esta memoria.





## Análisis de resultados

A la vista de los resultados podemos sacar las siguientes conclusiones.

### Respecto a las características de los usuarios

La diferenciación por **sexo** no es suficientemente determinante, tanto hombre como mujeres tienen los mismos problemas al resolver la prueba, aunque las mujeres tienen un índice de error ligeramente más alto. 45% contra 43%.

La diferenciación por **frecuencia de uso de internet**, los usuarios que usan internet con mayor frecuencia o frecuencia esporádica tienen un menor porcentaje de fallos en comparación con los usuarios con la menor frecuencia de uso y por lo tanto familiarización con el entorno.

La diferenciación por **rango de edad** nos indica que no existen grandes diferencias entre la comprensión entre diferentes edades.

1. En el rango de 30 a 50 años presenta un porcentaje de fallos mayor que otras opciones (48%).
2. En los rangos menores de 18 y mayores de 50 se han obtenido resultados similares con una tasa media de fallos.
3. La menor tasa de fallos se encuentra en el rango de 18 a 30 (39%).

### Respecto a la dificultad de la prueba y sus distintos parámetros

#### ▪ **Espacio entre palabras**

Es el principal parámetro por el cual se producen fallos, el porcentaje en la configuración 'Normal' es del 62%.

#### ▪ **Velocidad**

Es un parámetro determinante en el entendimiento los captchas las configuraciones normal y rápida tienen unos porcentajes de error de 51% y 58% respectivamente.

#### ▪ **Nº de números**

Unido a las configuraciones de velocidad o espaciado puede complicar el captcha y ser un parámetro a tener en cuenta, pero no es decisivo. Obliga a retener en la memoria la lista de números.



Las tasas más altas de errores se encuentran en los captchas que tienen los siguientes parámetros y en ese orden:

1. Espaciado más corto (0 ms).
2. Velocidad (más rápida de pronunciación de los números).
3. Mayor número de números (7).

Datos que coinciden con los captchas con mayor porcentaje de error.

Nº	Texto	Espaciado/Velocidad
3	6102	Normal/Rápida
8	3822364	Normal/Normal
9	8074598	Normal/Rápida

**Tabla 16 - Captchas más complicados**

### **Respecto a los resultados del reconocedor automático**

Podemos concluir que el entendimiento de la voz sintetizada ha sido muy bajo.

Tanto en las prueba de los captchas con ruido de fondo como en la prueba sin ruido de fondo, prácticamente no ha sido capaz de reconocer algún número.

Se puede aventurar pensar que el reconocedor automático puede estar diseñado para entender mejor un flujo de voz real que una voz robótica sintetizada.



# Conclusiones

---

Después de haber realizado la prueba y analizado los resultados podemos sacar como conclusión que la tecnología actual de reconocimiento de voz está bastante lejos de poder resolver captchas sonoros con garantías.

Los captchas sonoros son algo más complicados de resolver que los captchas gráficos pero a la vez son bastante más seguros, en nuestras pruebas el reconocedor de voz no fue capaz de reconocer ninguno de los captchas con ruido de fondo y a los que les quitamos el ruido de fondo no reconoció ninguno completo.

Por lo tanto es un sistema seguro y que resuelven la mayoría de usuarios, se han visto gran cantidad de errores en las pruebas con usuarios, en un porcentaje significativo pero hay que observar que no podían escuchar el captcha más de una vez, los parámetros que provocaban en mayor medida los fallos son la combinación de una menor distancia entre la pronunciación de los números y la velocidad de pronunciación.

Estoy satisfecho de haber podido llevar a cabo este trabajo, he podido experimentar con tecnologías que eran desconocidas para mí y he aprendido muchas cosas interesantes, a pesar de los problemas que me he ido encontrando y de la falta de tiempo debido a mi trabajo estoy contento con el resultado final.





# Bibliografía y webs consultadas

---

## Bibliografía

Los libros consultados para la realización del trabajo son:

- [1] George Shepherd, *ASP.NET 4.0 (Paso a Paso)*. Anaya Multimedia, 2010, 752 p.
- [2] Deitel, Paul J., *Java: Cómo programar*. Pearson Educación, 2008, 1338 p. 26 cm.
- [3] Rebecca M Riordan, *Head First Ajax*, O'Reilly, 2008, 527 p.

## Webs consultadas

Las páginas webs consultadas para la realización del trabajo son:

- [1] <http://webinsight.cs.washington.edu/projects/audiocaptchas/> Información sobre captchas sonoros. (última visita 01/10/2012)
- [2] <http://googleblog.blogspot.com/2006/11/audio-captchas-when-visual-images-are.html> Artículo en el blog del Google sobre los captchas sonoros. (última visita 01/10/2012)
- [3] <http://es.wikipedia.org/wiki/Captcha> Página con información sobre captchas en Wikipedia. (última visita 07/01/2012)
- [4] <http://cdn.ly.tl/publications/decaptcha-audio.pdf> Artículos sobre los captchas sonoros y su utilidad. (última visita 04/10/2012)
- [5] <http://blog.rodriгорега.es/tag/ocr/> Blog que habla sobre las posibilidades de romper captchas gráficos mediante OCR. (última visita 14/07/2013)
- [6] <http://computer.howstuffworks.com/captcha4.htm> Página que habla sobre las posibilidades de romper la seguridad de los captchas. (última visita 15/07/2013)



- [7] <http://jocr.sourceforge.net/> Programa usado para resolver por OCR los captchas gráficos. (última visita 15/07/2013)
- [8] <http://es.wikipedia.org/wiki/AJAX> Página en Wikipedia sobre la tecnología AJAX. (última visita 14/05/2013)
- [9] <http://es.wikipedia.org/wiki/ASP.NET> Página en Wikipedia sobre ASP.NET. (última visita 12/08/2013)
- [10] <http://www.schillmania.com/projects/soundmanager2/> Componente que permite la reproducción de audio en una web mediante Javascript. (última visita 01/02/2012)
- [11] <http://www.aspnetaudio.com/> Componente de pago que permite la reproducción de audio en una web para ASP.NET. (última visita 01/02/2012)
- [12] <http://developer.chrome.com/extensions/tts.html> Página de documentación de texto a voz de Google Chrome. (última visita 22/03/2013)
- [13] <http://lists.w3.org/Archives/Public/public-xg-htlmspeech/2011Feb/att-0022/htmltts-draft.html> Especificación del API TTS (Texto a voz) de Google. (última visita 04/05/2013)
- [14] [http://translate.google.com/translate\\_tts?q=TTS&tl=es](http://translate.google.com/translate_tts?q=TTS&tl=es) Ejemplo de síntesis de voz con el API de Google. (última visita 04/05/2013)
- [15] [http://wiki.openmoko.org/wiki/Google\\_Voice\\_Recognition](http://wiki.openmoko.org/wiki/Google_Voice_Recognition). Página que explica la manera de realizar reconocimiento de voz mediante el API de Google (última visita 12/08/2013)
- [16] <http://www.gnu.org/software/wget/> Página del proyecto del programa de envío de paquetes Wget. (última visita 12/08/2013)
- [17] <https://dvcs.w3.org/hg/speech-api/raw-file/tip/speechapi.html> Página en la que se detalla el API de reconocimiento de voz de Google. (última visita 12/08/2013)
- [18] <http://speechutil.com/> Página web que utiliza un sintetizador de voz para convertir texto a voz. (última visita 21/01/2012)
- [19] <http://www.festvox.org/voicedemos.html> Sintetizador de voz web. (última visita 03/08/2013)



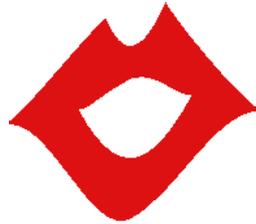
- [20] <http://www.google.com/recaptcha> Página oficial del sistema de captchas creado por Google. (última visita 23/03/2013)
- [21] <https://github.com/kripken/speak.js> Página del proyecto de speak.js. (última visita 15/07/2013)
- [22] <https://github.com/kripken/emscripten> Página del proyecto del compilador usado en speak.js emscripten. (última visita 10/08/2013)
- [23] <http://www.masswerk.at/mespeak/> Demostración web de modularly enhanced speak.js basado en speak.js. (última visita 15/01/2013)
- [24] <http://en.wikipedia.org/wiki/ESpeak> Página sobre eSpeak en Wikipedia. (última visita 25/08/2013)
- [25] <http://en.wikipedia.org/wiki/Formant> Página en Wikipedia sobre formantes. (última visita 25/08/2013)
- [26] [http://en.wikipedia.org/wiki/Speech\\_synthesis#Formant\\_synthesis](http://en.wikipedia.org/wiki/Speech_synthesis#Formant_synthesis) Página en Wikipedia sobre tipos de síntesis de voz. (última visita 25/08/2013)
- [27] [http://liceu.uab.es/~joaquim/speech\\_technology/tecnol\\_parla/synthesis/formant\\_synthesis/sintesis\\_formantes.html](http://liceu.uab.es/~joaquim/speech_technology/tecnol_parla/synthesis/formant_synthesis/sintesis_formantes.html) Página que trata sobre la síntesis por formantes (última visita 25/08/2013)
- [28] <http://espeak.sourceforge.net/docindex.html> Página oficial de documentación de eSpeak. (última visita 25/08/2013)
- [29] <http://sox.sourceforge.net/> Página oficial de SoX, convertidor de audio. (última visita 10/08/2013)
- [30] <http://caniuse.com/typedarrays> Estudio sobre compatibilidad de arrays tipados en navegadores. (última visita 20/08/2013)
- [31] <http://caniuse.com/#feat=audio> Estudio sobre compatibilidad de audio HTML5 en navegadores. (última visita 10/08/2013)
- [32] <http://html5doctor.com/html5-audio-the-state-of-play/> Página que analiza los diferentes formatos de audio admitidos por navegadores en HTML5. (última visita 17/05/2013)
- [33] <https://www.keycaptcha.com/> Captchas gráficos que funcionan a modo de puzle. (última visita 10/05/2013)



# Anexos

---

## Apendice A eSpeak



### Historia / Que es

Es un sintetizador de voz de código abierto escrito en C compatible con distintos idiomas que funciona en Linux y Windows. Anteriormente era conocido como Speak y fue codificado originalmente para Acorn/RISC\_OS en 1995 por Jonathan Duddington.

La voz sintetizada por eSpeak usa dos métodos de síntesis, la síntesis Klatt o el método original, ambos usan la síntesis por formantes que se refiere a la resonancia vocal característica de la voz humana. Puede no ser tan suave o natural como otros grandes sintetizadores que usan voces humanas grabadas para realizar la síntesis, pero es capaz de ser compatible con varios idiomas en un tamaño muy pequeño.

Diferentes formas de uso:

- Por línea de comandos.
- Una librería compartida para usar por otros programas
- Compatible con SAPI5 Windows, por lo que se usa en lectores y otros programas que soporten la interfaz SAPI5.
- eSpeak ha sido portado a varias plataformas: Android, Mac OSX, Solaris etc...
- Por su pequeño tamaño y funcionalidad se incluye en las distribuciones de Ubuntu y otras distribuciones Linux.

### Características

- Incluye diferentes voces a las que se las pueden modificar parámetros, como tono de voz, entonación etc..
- Se puede exportar la salida a un archivo WAV.
- SSML (Speech Synthesis Markup Language) es soportado (no completamente) y HTML.



- Tamaño compacto el programa y los datos incluidos múltiples lenguajes hacen un total de 2 MBytes.
- Puede traducir texto en fonemas, por lo que puede ser adaptado como un frontend para otros motores de síntesis de voz.
- Gran cantidad de idiomas incluidos y potencial para incluir nuevos.
- Herramientas de desarrollo disponibles para producir y ajustar datos fonéticos.

## Lenguajes disponibles

Africano  
Albano  
Aragonés  
Armenio  
Búlgaro  
Cantones  
Catalán  
Croata  
Checo  
Danés  
Holandés  
Inglés  
Esperanto  
Estonio  
Farsi  
Finlandés  
Francés  
Georgiano  
Alemán  
Griego  
Hindú  
Húngaro  
Islandés  
Indonesio  
Irlandés  
Italiano  
Kannada  
Kurdo  
Lituano  
Macedonio  
Malayo  
Mandarín  
Nepalés  
Noruego  
Polaco  
Portugués  
Punjabi



Rumano  
Ruso  
Serbio  
Eslovaco  
Español  
Swahili  
Sueco  
Tamil  
Turco  
Vietnamita

## Síntesis por formantes

eSpeak utiliza la síntesis por formantes como base de la generación de voz. La síntesis por formantes no usa muestras de voz humana. En su lugar su crea la salida de voz sintetizada mediante síntesis aditiva y un modelo acústico.

La síntesis aditiva es una técnica de síntesis de sonido que crea timbre mediante la adición de ondas sinusoidales unidas.

Los parámetros tales como la frecuencia fundamental, la vocalización o los niveles de ruido se varían en el tiempo para crear una forma de onda de voz sintética. Este método se llama a menudo síntesis basada en reglas.

Los sistemas que usan síntesis por formantes suelen generar una voz robótica, artificial, que no es posible confundir que la voz humana real. Pero que la voz sea lo más cercana a la de una persona real no es siempre el objetivo a llegar, La síntesis realizada con este método es generalmente muy inteligible incluso a velocidades muy altas. Una ventaja adicional es que al no tener que contar con una base de datos de muestras de sonido los programas que usan este método suelen ser pequeños en tamaño.

Como se tiene un control total sobre todos los aspectos de la salida de voz, se pueden realizar una gran variedad de modificaciones, por ejemplo en la entonación no solamente poder realizar una entonación de pregunta sino llegar a ser capaz de transmitir distintas emociones con modificaciones en la entonación de la voz.



## Métodos de síntesis en eSpeak

eSpeak usa dos métodos de síntesis distintos basados en la síntesis por formantes.

### Síntesis eSpeak

Este método crea sonidos de voz hablada como vocales o consonantes sonoras, mediante la suma de ondas sinusoidales para hacer los picos de los formantes. Consonantes sordas como /s/ son incluidas reproduciendo sonidos grabados. Las consonantes linguales como /z/ son incluidas mediante una mezcla voz sintetizada y sonido grabado.

### Síntesis Klatt

El sintetizador Klatt utiliza en su mayoría los mismos datos de formantes que la síntesis eSpeak. Crea sonidos de voz hablada comenzando con una forma de onda que es rica en armónicos (simulación de la vibración de las cuerdas vocales) y a continuación, la aplica filtros digitales con el fin de producir los sonidos del habla.



## Apendice B Datos de la encuesta

Las respuestas recogidas en la encuesta de Google Drive y que los usuarios rellenaban al finalizar la prueba son las siguientes.

Fecha	¿Le ha parecido interesante la prueba?	¿Qué parte le ha parecido más complicada a la hora de entender los CAPTCHA's?	¿Alguna sugerencia para mejorar el test?	¿Cree que estas pruebas son lo suficientemente complicadas como para no poder resolverse por un sistema automático?
8/18/2013 19:41:22	Si	Cuando el sintetizador de voz va muy rápido. Se le entiende mal el 0, parece que lo dice un francés.	Los captchas habitualmente permiten ejecutarse más de una vez, por la dificultad que algunos conllevan.  Algunos de estos captchas sonoros son bastante complejos y creo sería bueno poder reproducirlos más de una vez ya que ha habido un par de ellos que por velocidad me he dejado guiar y he jugado un poco a la lotería. Si se pudiera repetir estaríamos seguros de que es porque el humano lo entiende y no porque ha tenido un golpe de suerte	Si
8/24/2013 13:16:26	Si	En cuatro de los intentos los números eran dichos demasiado rápido.	Cambiar el sonido de fondo por conversaciones con la misma voz que la de los números.	Si
8/24/2013 18:33:57	Si	cuando los numeros les dicen muy deprisa	no	No
8/24/2013 18:55:45	Si	diferenciar entre el 5 y el 0	.	No
8/24/2013 19:15:37	Si	la rapidez con la q el hombre tartamudo me ha dictado los numeros.	no	Si
8/24/2013 20:04:39	Si	la velocidad	no	No lo sé
8/25/2013 22:47:09	Si	En algunos casos la velocidad es extrema!	Controlar la velocidad de reproducción para que sean comprensibles. Usar letras ademas de los números?	No lo sé
8/26/2013 9:55:54	Si	Cuando la locución va muy deprisa	No.	No
8/26/2013 9:56:57	Si	Los que iban más rápido	Me ha sorprendido que se pueda repetir el test al terminarlo, aunque supongo que también será curioso ver cuántas oportunidades necesita una persona para resolver los captchas más rápidos.	No lo sé



8/26/2013 11:00:46	Si	Los CAPTCHA's en que los números se dictaban rápidamente, más que nada por el acento no-natural de la voz.	Cuando he fallado, no se podía volver a reintentar... los botones de "Resolver" y "Pasar" no funcionaban correctamente.	No
8/26/2013 12:05:15	Si	Hay que estar pendiente, en algunos captchas va tremendamente rápido, es en los que he fallado, pero aun así es entendible.	Hum, no sabría decir, ya que no se como de preciso es un sistema automático comparado con una persona real, supongo que lo será bastante...	algunas son muy lentas aunque haya distorsión
8/26/2013 14:20:54	Si	Cuando dicta los números a mayor velocidad.	No	No lo sé
8/26/2013 14:21:43	Si	la repetición rápida de números y el distinguir entre el 3 y el 6	la utilización de distintos tonos y voces distintas	No lo sé
8/26/2013 15:28:45	Si	al principio..	Quizas mejor explicado al principio..	No
8/26/2013 15:49:42	Si	al principio	me parece una idea original y que permite resolver mejor este problema que los signos graficos visuales que suelen llamar a error con frecuencia	Si
8/26/2013 17:17:45	Si	En algunos, la velocidad a la que se dicen los números.	No se me ocurre ninguna.	No
8/26/2013 18:04:23	Si	la voz	...no	No lo sé
8/26/2013 18:11:43	Si	Las veces que sonaban más rápido	Marcar de alguna manera cuando han terminado de oirse los números que no sea remarcar el botón de verificación	Si
8/26/2013 18:19:49	Si	el sonido	.	No
8/26/2013 18:36:01	Si	El ruido de fondo no me molestaba demasiado. Lo peor creo que era aquellos en los que los números se decían muy rápido y seguidos, lo que, unido a la distorsión, hacía muy difícil identificarlos.	Ninguna.	No lo sé
8/26/2013 18:39:45	Si	La rapidez a la hora de decir los números, y la pronunciación.	No estaría de más una breve explicación de lo que se quiere conseguir con el experimento, a la hora de finalizar el test.	Si
8/31/2013 17:12:45	Si	la pronunciacion del personaje que dicta los numeros	sin sugerencia	No lo sé
9/1/2013 20:13:49	Si	No se indica el formato en que deben meterse los numeros.  ej/Numero / numero / numero Estaría bien decir que los numeros deben	Desconcierta un poco que en las instrucciones te digan puedes volver a intentar resolver un captcha si tienes intentos disponibles y durante la primera prueba nunca tienes mas de uno. he visto al final que se puede repetir la prueba al final con mas	Si



		<p>intriducirse uno seguido del otro sin espacios. Puede falsear los resultados de la prueba</p> <p>Había ciertos capchas que se hacian muy complicados por la rapidez. No daba tiempo a anotar los numeros, había que memorizarlos y eran secuencias algo largas</p>	<p>intentos pero pensaba que durante la primera prueba por lo que dicen las instrucciones en alguna captcha tendría mas de una oportunidad y no.</p>	
9/3/2013 9:32:32	Si	<p>La mezcla de ruidos diferentes a la vez que se dicen los números.</p>	<p>El primer número que ha hecho falta apuntarlo no sabía que hacia falta y de memoria no he podido completarlo, aunque creo que no era difícil entenderlo.</p>	Si
9/3/2013 11:37:10	Si	<p>los que van mas rapidos</p>	<p>que no vayan tan rapido</p>	Si
9/3/2013 20:30:58	Si	<p>Los que van muy rápido.</p>	<p>No tengo conocimientos sobre el tema.</p>	No lo sé





## Apendice C Contenidos del CD

En el CD que se adjunta con esta memoria se incluyen tanto los programas utilizados como el código fuente.

Se pueden encontrar el siguiente contenido.

- **Memoria en PDF**

Este documento almacenado en formato PDF.

- **Presentación en PowerPoint**

La presentación que se va utilizar para la lectura del trabajo.

- **Carpeta web**

Incluye el proyecto de la página web que incluye su código fuente.

- **Carpeta Speak.js**

Se incluyen los archivos del proyecto del sintetizador de voz Speak.js.

- **Carpeta aplicaciones**

Incluyen las aplicaciones Wget, SoX.

- **Carpeta captchas**

- Los archivos de sonido de cada captcha que suenan en la aplicación exportados a wav y flac, con y sin sonido de fondo.
- Los archivos de resultados individuales del reconocimiento de voz del reconocedor de Google.
- Los sonidos de fondo usados en los captchas sonoros.

- **Carpeta base de datos**

Se incluye una copia de la base de datos actual con todas las tablas y datos usados para realizar las estadísticas.