



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA  
Escuela de Ingeniería Informática



**SED**

# **STROOP EFFECT DETECTOR PROTOTYPE**

PROTOTIPO SOFTWARE DE UN DETECTOR DEL EFECTO STROOP PARA DAÑOS  
CEREBRALES EN EL LÓBULO ORBITOFRONTAL

AUTOR: Xerach Acorán Hernández Quesada  
TUTOR: Beatriz Correas Suárez  
DEPARTAMENTO DE Informática y Sistemas

# Índice de contenido

1 AGRADECIMIENTOS.....	3
2 RESUMEN DEL TRABAJO DE FIN DE GRADO.....	5
3 ESTADO ACTUAL Y OBJETIVOS.....	7
4 SÍNDROME ORBITOFRONTAL: Toma de decisiones.....	9
4.1 INTRODUCCIÓN.....	9
4.2 LA CORTEZA FRONTAL Y SU ANATOMÍA: DESCRIPCIÓN Y CIRCUITOS.....	10
Diagrama Circuito Fronto-Subcortical.....	12
4.3 REGIÓN ORBITOFRONTAL.....	13
4.3.1 ANATOMÍA:.....	13
4.3.2 FUNCIONES:.....	13
4.3.3 SÍNDROME ORBITOFRONTAL.....	14
4.3.4 EXPLORACIÓN NEUROLÓGICA Y NEUROPSICOLÓGICA DE LA PATOLOGÍA FRONTAL.....	15
5 TEST DE STROOP.....	17
5.1 Versiones.....	18
6 COMPETENCIAS A CUBRIR.....	21
6.1 CII01.....	21
6.2. CII02.....	22
6.3. CII04.....	23
6.3.1. Pliego de especificaciones técnicas que han de regir en la utilización de una aplicación informática destinada a la investigación médica.....	23
6.3.1.1. Ámbito geográfico.....	23
6.3.1.2. Requisitos de fiabilidad.....	23
6.3.1.3. Entorno pre-explotación.....	23
6.4. CII18.....	24
6.4.1. Normativa y regulación de la informática en el ámbito internacional.....	25
6.4.2. Estados Unidos.....	25
6.4.3. Alemania.....	26
6.4.4. Austria.....	26
6.4.5. Gran Bretaña.....	26
6.4.6. Holanda.....	27
6.4.7. Francia.....	27
6.4.8. Chile.....	28
6.4.9. Normativa y regulación de la informática en el ámbito europeo.....	28
6.4.10. Normativa y regulación de la informática en el ámbito nacional.....	29
6.5 TFG01.....	34
7 APORTACIONES.....	35
8 NORMATIVA Y LEGISLACIÓN.....	37
8.1. Software libre.....	37
8.2. Licencias de software.....	38
8.3. Leyes sobre seguridad.....	39
9 SECTOR DE LA ACTIVIDAD.....	41
10 ANALISIS.....	43
10.1 DIAGRAMA DE ESTADOS.....	43
10.2 DIAGRAMA DE CASOS DE USO.....	45

10.3 COSTE Y ESFUERZO.....	47
10.3.1 CLASIFICACIÓN DE TRANSACCIONES Y ARCHIVOS.....	47
10.3.2 CÁLCULO DEL FACTOR DE COMPLEJIDAD.....	48
11 REQUISITOS SOFTWARE.....	51
12 REQUISITOS HARDWARE.....	53
13 DESARROLLO.....	55
13.1 METODOLOGÍA.....	55
13.2 VERSION ALPHA.....	55
13.3 VERSION BETA 1.0.....	56
13.4 VERSION BETA 2.0 .....	57
14 MANUAL DE USUARIO.....	59
15 REFERENCIAS BIBLIOGRÁFICAS.....	71
16 BIBLIOGRAFÍA.....	73
17 ANEXO.....	75
17.1 CODIGO FUENTE.....	75
17.1.1 Visor.vb.....	75
17.1.2 VisorDesigner.vb.....	75
17.1.3 SettingsDesigner.vb.....	76
17.1.4 ResourceDesigner.vb.....	78
17.1.5 Control.vb.....	80
17.1.6 ControlDesigner.vb.....	91
17.1.7 AssemblyInfo.vb.....	103
17.1.8 Application_Designer.vb.....	103

# 1 AGRADECIMIENTOS

Me gustaría empezar agradeciendo este proyecto a mis padres, por los esfuerzos realizados para que su hijo estudiara la carrera que quiso, a mi novia Fanny por apoyarme y fortalecerme en momentos de flaqueza y desmotivación que en esta titulación han sido más de los deseados y esperados. A mi tutora de proyecto Beatriz Correas, por facilitarme la dura tarea de realizar un proyecto y acogerme cuando otros no querían/podían. A mis profesores de la adaptación al grado Agustín Sánchez, Javier Sánchez, José Juan Hernandez, Miguel Alemán y Nelson Monzón, de todos ellos he aprendido muchas cosas que seguro podré aplicar en esta nueva etapa que surge en mi vida. Al programa de investigación de Desarrollo Tecnológico Social de la Universidad de Las Palmas de Gran Canaria que ha permitido adentrarme en el mundo de la informática como recurso social y me ha abierto un campo muy divertido, interesante y emocionante al que poder dedicar mi futuro aplicando mi formación. A mis compañeros y amigos de Universidad Aitor, Cristian, Cynthia, David, Deyán, Francisco y Román con los cuales he pasado la mayor parte de este curso, y a mis amigos íntimos Hector, Pablo y Rakesh que siempre se ofrecen para cualquier imprevisto sin pedir nada a cambio incluso, en ocasiones, perdiendo algo suyo para darme ese empujón que es necesario para algunos de nosotros en momentos puntuales de la vida.



## 2 RESUMEN DEL TRABAJO DE FIN DE GRADO

1. El "Detector de Efectos Stroop" (SED – Stroop Effect Detector), es una herramienta informática de asistencia, desarrollada a través del programa de investigación de Desarrollo Tecnológico Social de la Universidad de Las Palmas de Gran Canaria, que ayuda a profesionales del sector neuropsicológico a identificar problemas en la corteza orbitofrontal de un individuo, usándose para ello la técnica ideada por Schenker en 1998.

Como base metodológica, se han utilizado los conocimientos adquiridos en las diferentes materias de la adaptación al grado en Ingeniería Informática como Gestión del Software, Arquitectura del Software y Desarrollo de Interfaces de Usuario así como conocimiento adquirido con anterioridad en asignaturas de Programación e Ingeniería del Software I y II. Como para realizar este proyecto sólo el conocimiento informático no era suficiente, he realizado una labor de investigación acerca del problema, teniendo que recopilar información de otros documentos científicos que abordan el tema, consultas a profesionales del sector como son el Doctor Don Ayoze Nauzet González Hernández, neurólogo del hospital Doctor Negrín de Las Palmas de Gran Canaria y el psicólogo Don José Manuel Rodríguez Pellejero que habló de este problema en clase del máster de Formación del Profesorado y que actualmente estoy cursando.

Este trabajo presenta el test de Stroop con las dos versiones de Schenker: RCN ("Reading Color Names") y NCW ("Naming Colored Words"). Como norma general, ambas pruebas presentan ante los sujetos estudios palabras (nombres de colores) escritas con la tinta de colores diferentes. De esta forma, el RCN consiste en leer la palabra escrita omitiendo la tonalidad de su fuente e intentando que no nos influya. Por el contrario, el NCW requiere enunciar el nombre del color de la tinta con la que está escrita la palabra sin que nos influya que ésta última sea el nombre de un color.

La aplicación ayuda a minimizar riesgo de error que puede cometer el profesional facilitándole la respuesta en una pantalla diferente a la del sujeto al que esté evaluando. También permite, de manera amigable, personalizar la prueba o alterarla durante su ejecución, adaptando el test a las necesidades de cada paciente así como reducir el tiempo medio empleado en este tipo de evaluaciones.

### 3 ESTADO ACTUAL Y OBJETIVOS

Esta herramienta se usa, actualmente, en su modo tradicional (papel y lápiz). Dichas pruebas son lentas de evaluar (duran unos 5 minutos por prueba) principalmente debido al detenimiento que el profesional debe dedicar a la respuesta del sujeto. La versión actual, formato papel, consta de 3 láminas que contienen 5 columnas de 20 elementos separadas entre sí por unos 3 cm que deben ser administradas en el siguiente orden:

1. Lectura de Palabras (P). Cada uno de los elementos de la página uno, es el nombre de los tres colores empleados en el test repetidos de manera aleatoria e impresos en tinta negra. La persona debe leer durante 45 segundos los nombres de los colores “rojo”, “verde” y “azul”, impresos en negro. El evaluador anota el número de aciertos.
2. Denominación de Colores (C). La página dos, está formada por cinco columnas de símbolos tipo “XXX” coloreados de manera aleatoria con los tres colores empleados en el test. Se pide a la persona, durante 45 segundos, que denomine los colores impresos en cada fila de “X”, y se puntúa los aciertos. El evaluador anota el número de aciertos.
3. Por último, la condición de interferencia, Palabras-Colores (PC). En la página tres, aparece de nuevo el nombre de los tres colores empleados en el test pero impresos en tinta coloreada, de manera aleatoria y sin coincidencia entre el nombre del color y el color de la tinta en que está impreso. El sujeto, durante 45 segundos, debe responder el color de la tinta con la que está impresa la palabra ignorando el significado. Se puntúa el número de aciertos.



4. Finalmente, el evaluador aplica las fórmulas siguientes:

$$\text{➤ } PC' = (P * C) / (P + C)$$

$$\text{➤ } \text{Interferencia} = PC - PC'$$

Si la puntuación de interferencia es positiva, el individuo ha inhibido adecuadamente la respuesta automática y, si es negativa, presumiblemente ha inhibido peor de lo que hubiera sido esperable, aunque se considera que la población general oscila entre -10 y 10.

El objetivo de este proyecto es el de informatizar el proceso de Stroop para una mejor aplicación, haciéndolo más fiable (minimizando el error humano) y adaptable para cada paciente en cuestión de segundos y de manera intuitiva para el evaluador.

## **4 SÍNDROME ORBITOFRONTAL: Toma de decisiones.**

### **4.1 INTRODUCCIÓN**

La capacidad ejecutiva es una de las funciones cognitivas más difíciles de definir y, por tanto, el desarrollo de técnicas que la evalúen directamente es un proceso complejo. Desde el punto de vista cognitivo, el término "función ejecutiva" se refiere a aquellos procesos por los que un individuo mejora su actuación en tareas complejas, con varios componentes. Es decir, se trata de la capacidad de pensar en una "meta concreta" y ser capaz de organizar los medios para su consecución, de prever sus consecuencias y valorar las posibilidades de éxito, de concentrarse en los puntos clave, de reflexionar sobre si se está realizando el plan según lo acordado y de modificarlo si no es así.

De esta función, esencial para llevar a cabo las conductas complejas que caracterizan el comportamiento del ser humano, se encarga una región del cerebro muy concreta: la corteza prefrontal. Su alteración da lugar al síndrome disejecutivo, caracterizado por algunos o la totalidad de los siguientes trastornos:

- a) Incapacidad para iniciar, detener y modificar una conducta en respuesta a un estímulo que cambia.
- b) Incapacidad para llevar a cabo una serie de actos consecutivos que permitan resolver un problema.
- c) Incapacidad para organizar un plan de acción .
- d) incapacidad para inhibir respuestas inapropiadas y perseveración o repetición anormal de una conducta.

## **4.2 LA CORTEZA FRONTAL Y SU ANATOMÍA: DESCRIPCIÓN Y CIRCUITOS**

Los lóbulos frontales ocupan 1/3 de la corteza cerebral del ser humano. Constituyen una región de la corteza cerebral cuya función sigue sin estar definida completamente. Luria (1966-1979) la describió como la porción más compleja y de desarrollo más reciente del sistema nervioso central [1].

En los años 70, autores clásicos [2] todavía hablaban de los misteriosos lóbulos frontales. En el año 2000 aparecieron revisiones dando ese título de "misteriosas" a determinadas regiones de la corteza frontal [3]. Parece evidente que, de alguna forma, esa "diferencia anatómica", ha permitido al ser humano sobrevivir y dominar sobre otras especies, lo que supone una capacidad idónea de adaptación al medio [4], a la vez que una independencia del mismo a la hora de llevar a cabo determinadas conductas [5]. Esta capacidad para llevar a cabo conductas complejas, que reside en una parte del lóbulo frontal, la corteza prefrontal. La corteza prefrontal está conectada con la práctica totalidad de las áreas corticales, subcorticales y límbicas, lo que le proporciona información de toda clase de eventos externos e internos, pudiendo conectar ambos entre sí. Ello sitúa a la corteza prefrontal en posición de poder valorar la respuesta más adecuada aun estímulo concreto, dependiendo de las condiciones externas y de la situación del individuo.

Los límites anatómicos de los lóbulos frontales son:

1. La circunvolución central, que los separa del lóbulo parietal.
2. La cisura de Silvio, que los separa del lóbulo temporal.
3. El cuerpo calloso, que los separa de estructuras subcorticales [6].

Al mismo tiempo, la corteza del lóbulo frontal se divide en 3 regiones, en relación con el tipo histológico de las mismas:

- Corteza motora-premotora, que comprende:
  1. Circunvolución precentral o área motora primaria.

2. Corteza premotora.
  3. Área suplementaria motora.
  4. Área de Broca, de producción de lenguaje .
  5. Área del control visual voluntario.
- Corteza prefrontal, que comprende:
    1. Corteza dorsolateral .
    2. Corteza orbitaria. (Areas 11, 12 y 47 de Brodmann).
    3. Corteza mesial.
  - Corteza paralímbica: que comprende la región anterior del giro en cíngulo.

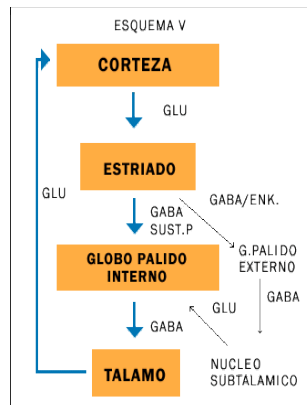
La corteza prefrontal y paralímbica y sus regiones, dorsolateral, orbitaria y mesial, están conectadas con diversas estructuras subcorticales, formando circuitos frontosubcorticales, lo que explica que los síntomas que provoca una lesión frontal sean distintos según la región o circuito afectado.

Hay 5 circuitos frontosubcorticales reconocidos [7]:

1. Un circuito "motor", que se origina en el área suplementaria motora.
2. Un circuito "oculomotor", que parte del área 8.
3. Tres más que parten de las distintas regiones de la corteza prefrontal (dorsolateral, orbitaria y del giro en cíngulo anterior).

Los circuitos tienen una estructura básica [ver [diagrama](#)]: conectan el lóbulo frontal con el núcleo estriado, el pálido, la sustancia negra, el tálamo y de nuevo la corteza frontal. Todos los circuitos comparten estructuras en común, pero se mantienen separados anatómicamente. En estos circuitos intervienen varios neurotransmisores.

### Diagrama Circuito Fronto-Subcortical



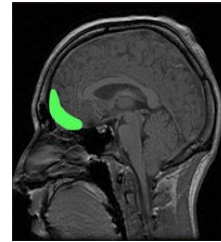
La zona en que se centra SED ("Stroop Effect Detector") es la orbitofrontal, siendo de ahora en adelante, el tema central de este documento.

### **4.3 REGIÓN ORBITOFRONTAL**

La corteza orbitofrontal (“COF” de ahora en adelante) es una región del lóbulo frontal del cerebro relacionada con el procesamiento cognitivo de la toma de decisiones. Incluye las áreas del corticales de asociación de Brodmann 10, 11 y 47 [8]. Algunos autores lo consideran parte del sistema límbico.



área 10 de Brodmann



corteza orbifrontal vista pefil.

#### **4.3.1 ANATOMÍA:**

COF se define, anatómicamente, como la parte de la corteza prefrontal que recibe proyecciones desde los núcleos mediales magnocelulares del tálamo mediodorsal y se encuentra entre las órbitas en que se ubican los ojos.

#### **4.3.2 FUNCIONES:**

La COF se encuentra entre las regiones menos comprometidas del cerebro humano, pero se ha propuesto que se encuentra en funciones de integración sensorial, en la representación del valor afectivo de los reforzadores, en la toma de decisiones y en la formación de expectativas. Estas propuestas se apoyan en estudios de neuroimagen realizados sobre participantes sanos y en investigaciones neuropsicológicas en pacientes con daño cerebral en regiones concretas de la COF. La universidad de Leipzig (Sajonia, Estado Libre) ha realizado investigaciones donde muestran que la COF humana se activa durante la realización de juicios de coherencia intuitiva.

### 4.3.3 SÍNDROME ORBITOFRONTAL

El síndrome orbitofrontal se produciría por una lesión a cualquier nivel del circuito orbitofrontal pero especialmente por lesión de las áreas basales 10, 11 y/o 47 de Brodmann. Una causa frecuente de lesión en esta región es la ruptura de un aneurisma de la arteria comunicante anterior.

La corteza orbitofrontal corresponde a la representación neocortical del sistema límbico [2] y tiene que ver con la adecuación en tiempo, espacio e intensidad, de la conducta en respuesta a un estímulo externo. Las lesiones en este área parecen desconectar un sistema de vigilancia frontal del sistema límbico y, como resultado, se produce una desinhibición y labilidad emocionales, es decir, un cambio de personalidad con las siguientes características:

1. Los pacientes pueden aparecer desinhibidos, sin tener en cuenta las normas sociales, con un mal control de impulsos (agresividad sin motivo, bulimia...), incapaces de inhibir respuestas incorrectas, son generalmente, reiterativos.
2. Pueden presentar el síndrome de dependencia del medio ambiente de Lhermitte [9].
3. “Moria”. El paciente se divierte con cosas que a nadie le hace gracia [4].
4. Euforía o Manía (maníaco).

SED es una herramienta que se centra en la ayuda de conseguir un diagnóstico por parte del profesional encontrando posibles alteraciones o deficiencias en personas incapaces de inhibir respuestas incorrectas.

#### 4.3.4 EXPLORACIÓN NEUROLÓGICA Y NEUROPSICOLÓGICA DE LA PATOLOGÍA FRONTAL

Un problema, a la hora de utilizar los tests que miden la función ejecutiva, es cuáles y cuántos se deben emplear. Unas pruebas hacen mayor énfasis en la capacidad organizativa, o de planear y ordenar, mientras que otros lo hacen en la memoria y otros en la capacidad de inhibir respuestas inadecuadas.

Individualmente, las pruebas frontales no son excesivamente específicos y no sólo se alteran con lesiones frontales, sino con otras patologías como en el caso de las personas con trastorno de memoria asociado a la edad [10]. Por ello, se recomienda realizar al menos tres pruebas de función ejecutiva que midan diversos aspectos del síndrome.

Los distintos tipos de tests de función ejecutiva son:

1. Neuropsicológicos.
2. Síndrome orbitofrontal. Capacidad de inhibición y perseveración.
  1. Secuencias alternantes visuales.
  2. *Stroop Test I y II: El paciente ha de decir los colores (rojo, azul o verde) de unos puntos que aparecen impresos en una hoja de papel. Posteriormente debe decir el color con el que están impresos los nombres de colores en una hoja; nunca una palabra está impresa en el mismo color que representa, es decir, la palabra azul está impresa en rojo, la palabra rojo en negro, etc. Se trata de comprobar cómo el paciente inhibe la tendencia que todos tenemos a leer lo que pone, para decir el color en que está impresa la palabra.*
3. Paradigma go-no go.

SED reproduce el Stroop Test I y II ampliando el número de combinaciones posibles y facilitando la tarea de corrección, minimizando los errores por parte del evaluador.





## 5 TEST DE STROOP

El test de Stroop, originariamente, estaba interesado en conocer los efectos de la interferencia perceptual, principalmente en conocer que efectos tenía esta interferencia sobre el comportamiento. Más tarde se descubrió que era un test sensible para discriminar personas con daño cerebral, siendo capaz, incluso de discriminar la localización de este daño, en el hemisferio derecho y/o izquierdo así como en el parte anterior y/o posterior. También permite localizar disfunciones de tipo subcortical. En este contexto profundizaremos en su descripción y conoceremos algunas de las versiones que han sido desarrolladas.

El test de colores de Stroop (Stroop, 1935) evalúa la capacidad de cambio de una estrategia inhibiendo la respuesta habitual y ofreciendo una nueva respuesta ante nuevas exigencias estímulares (García y Muñoz, 2000). Los resultados obtenidos con esta técnica desde un primer momento y en sucesivos estudios (McClain, 1983) han demostrado que la latencia de respuesta del sujeto, cuando la palabra es incompatible con el color de la tinta, aumenta significativamente con relación a cuando la palabra es congruente o neutra. Así pues, el efecto Stroop viene explicado por la incongruencia o incompatibilidad existente entre el color en que está impreso la palabra y el significado de dichas palabras, siendo en ocasiones conceptualizado (González, Sáiz y Mateos, 1988) como una forma de *priming negativo*<sup>1</sup>.

Desde que en 1886, Cattell comprobase que el tiempo que se tarda en leer palabras es mucho menor que el necesario para reconocer simples colores, y la posterior formulación de la tarea de interferencia por Stroop, este campo de investigación ha sido enormemente prolífico. Arana, Cabaco y Sanfeliú (1997) presentan una contextualización histórica sobre las raíces del fenómeno, así como un posterior análisis del paralelismo de la tarea Stroop y el abordaje del mecanismo de atención en sus cuatro grandes etapas, siendo a partir de mediados de siglo cuando la tarea de interferencia Stroop adquiere gran relevancia para los investigadores, pasando en los últimos años a desarrollar una inusitada importancia en el terreno aplicado, fundamentalmente clínico, tanto desde un plano diagnóstico como predictivo en los ámbitos más diversos.

McLeod (1991) desarrolla una revisión integradora de medio siglo de investigaciones del

---

1 Priming negativo. Literalmente cebo negativo, hace referencia a trampas u obstáculos que penalizan al sujeto.

paradigma Stroop. En la literatura experimental de dos décadas atrás, encontramos una serie de estudios que apelaron a efectos inhibitorios de atención, y que se apoyaban en el empleo de la tarea desarrollada por Stroop en 1935. El efecto típico encontrado con esta tarea es un retraso al nombrar el color de tinta en que está escrita una palabra que es el nombre de un color diferente (por ejemplo, palabra verde escrita en rojo). En principio, podría suponerse para llevar a cabo óptimamente esta tarea, el sujeto debería ignorar el nombre de la palabra de color de la cual debe nombrar el color de la tinta en que esta impresa; es decir, usando el ejemplo anterior debería responder “rojo”. Uno de los trabajos pioneros y más importantes en este campo fue el de Dalrymple y Budayr (1966), quienes encontraron que nombrar colores en una lista de elementos tipo Stroop era especialmente lento cuando el color de cada palabra coincidía con el nombre del color del elementos siguiente. La originalidad del Stroop radica en el hecho de que la palabra coloreada suscita una respuesta verbal automática que requiere muchas de las mismas funciones neuropsicológicas que son necesarias para nombrar los colores. Además, la velocidad de ambas reacciones (leer palabras y nombrar colores) es tal, que la respuesta de leer palabras ocupa los canales neuropsicológicas que, al mismo tiempo, la respuesta de nombrar colores necesita para poder ser procesada.

Algunos estudios neuropsicológicos han demostrado que la interferencia del Stroop se produce como consecuencia de interferir los procesos verbal y no verbal en la fase de respuesta, y no en la confusión del sujeto (Golden, 1978). El sujeto realiza la tarea o bien ejecutando dos respuestas secuencialmente (lectura de palabras seguida del nombramiento del color) o bien suprimiendo mediante control voluntario la respuesta automática de lectura de palabras.

## **5.1 Versiones**

En la práctica clínica neuropsicológica, habitualmente se utiliza el test de Stroop en la versión de papel y lápiz, de Golden (1978). En el ámbito de la investigación neuropsicológica se han introducido múltiples variantes, denominadas genéricamente paradigmas Stroop. La principal variante ha sido introducida en las versiones computarizadas.

Una variante, especialmente pensada para pacientes con diversas psicopatologías es el

Stroop emocional. En esta prueba las palabras son amenazantes o peligrosas, no neutras como en la versión clásica del Stroop. Ha sido utilizado con diversas finalidades (ver, Pérez y Fox, 2003). Anderson y Haldrup (2003) lo aplicaron para estudiar la atención parcial de pacientes con dolor crónico. Los resultados muestran una debilidad del efecto de interferencia del Stroop con tiempos de reacción lentos para palabras desagradables en el grupo de pacientes, pero no difieren significativamente de los sujetos del grupo de control. Ambos grupos fueron más lentos en exhortar palabras y en exhibir la clásica interferencia Stroop del efecto de las palabras de color. Gallardo, Baños, Belloch y Ruipérez (1999), lo aplicaron en pacientes del espectro depresivo, obteniendo que el sesgo atencional sólo se manifiesta en pacientes del subgrupo de depresión mayor, por lo que no parece constituir un factor de vulnerabilidad cognitiva para la depresión. Martínez y Marín (1997) estudiaron el procesamiento de información emocional en relación con la alexitimia, con el objeto de valorar la hipótesis que sostiene que la alexitimia está determinada por un déficit cognitivo en la capacidad para procesar estímulos efectivos. Quero, Baños y Botella (2000) en su estudio sobre la existencia de sesgos atencionales en el trastorno de pánico con agorafobia, obtienen como estos pacientes tienden a procesar selectivamente la información a nivel estratégico y no automático.

Una de las variantes más relevantes ha sido la de usar entradas sensoriales a la modalidad visual. La más comúnmente utilizada es aquella en la que el sujeto tiene que denominar los dígitos presentados visualmente mientras oye otros dígitos presentados de forma auditiva (García, 1991).

La variante creada en este proyecto, es la de Shenker (1998), quien comparó dos modalidades de Stroop: El Stroop que consiste en ignorar las palabras y leer el color y el Stroop que consiste en leer la palabra e ignorar el color.



## 6 COMPETENCIAS A CUBRIR

La realización de este trabajo de fin de grado cubre adecuadamente las competencias que se enumeran a continuación:

### 6.1 CII01

Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación vigente.

1. Se ha satisfecho la capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos debido a que se ha tenido que esquematizar e implementar una aplicación de escritorio capaz de automatizar los tests RCN ("Reading Color Names) y NCW ("Naming Colored Words"), incluyendo sus correspondientes pruebas de calidad, seguridad y fiabilidad, así como vigilando, respetando y cumpliendo las normativas vigentes relacionadas con el uso previsto.
2. Sobre el SED ("Stroop Effect Detector") podemos concluir que, tras un profundo estudio del panorama legislativo actual, a pesar de constituir una novedad a nivel informático, cumple rigurosamente con las leyes actuales. Como muestra documental, en la presente memoria puede consultarse los capítulos "Estado actual y objetivos", "Síndrome Orbitofrontal: toma de decisiones", "Requisitos software" y "Requisitos hardware" para comprobar la completitud de la competencia. En los citados apartados, podemos encontrar la justificación que ha motivado el diseño y desarrollo de esta aplicación de escritorio. Aquí se encuentran exhaustivamente definidas las fases de captura de requisitos, análisis y diseño de los casos de uso necesarios para satisfacer las necesidades de la aplicación, y por otro lado se justifica la utilización del hardware más adecuado para la implementación de la misma, teniendo en cuenta parámetros de fiabilidad, calidad y seguridad. Entiéndase por tanto que, fruto de la combinación de un hardware y un software diligentemente escogido, el resultado de este trabajo de fin de grado resulta en una aplicación robusta, estable y que cumple con la ética y la legislación propia de este tipo de producto.

## **6.2. CII02**

Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social. Esta capacidad ha quedado satisfecha al llevar a cabo la implementación de la aplicación de escritorio de manera correcta y la posibilidad de continuar actualizándola como administrador en el futuro.

1. En la elaboración de un proyecto como éste, la planificación es importante para lograr llegar en los tiempos establecidos. El logro de la presentación de este proyecto en fecha confirma que la competencia que se solicita ha quedado saldada.
2. Cabe destacar que, mientras la valoración del impacto económico se torna difusa dado que en un principio y a corto plazo la aplicación de escritorio no es susceptible de entrar en una fase de explotación que produzca capital monetario, sí que proporciona poderosas herramientas de economizado de tiempo para los investigadores que la utilicen a la hora de realizar el RCN o el RCW, reduciendo los cálculos de las conclusiones del test de una media de 10 minutos por sujeto de estudio a apenas unos cuantos segundos.
3. Cabe destacar, además, que es la primera vez documentada que se implementa un detector de efecto Stroop.
4. Por otro lado, el impacto en el campo facultativo y social sí que se antoja mucho más sencillo de evaluar, dado que el trastorno orbitofrontal constituye una incapacidad difícil de detectar y con múltiples manifestaciones asociales en nuestro día a día. La aceleración en su identificación contribuirá a tratar el defecto de una manera más adecuada y temprana.

### **6.3. CII04**

Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes. Procedemos a cumplir con este objetivo introduciendo a continuación en el presente documento los apartados necesarios:

#### **6.3.1. Pliego de especificaciones técnicas que han de regir en la utilización de una aplicación informática destinada a la investigación médica.**

Para la correcta utilización de esta aplicación las condiciones técnicas requeridas engloban:

##### **6.3.1.1. *Ámbito geográfico***

El ámbito geográfico alcanza el nivel global. Lingüísticamente hablando, la aplicación se ha ideado para castellanoparlantes, pero es posible adaptarla a cualquier otro idioma cambiando los sustantivos que hacen referencia a los colores.

##### **6.3.1.2. *Requisitos de fiabilidad***

El desarrollo de la aplicación debe acometerse de tal forma que su resultado garantice:

- Alta fiabilidad.
- Comportamiento estable.

##### **6.3.1.3. *Entorno pre-explotación***

De ser necesarias tareas correctivas una vez implementada y distribuida la aplicación (con el fin de facilitar el desarrollo), la realización de pruebas no debe afectar al sistema en producción, por lo que será preciso que cualquier actividad a realizar sobre la



funcionalidad se efectúe previamente en un entorno de pre-explotación similar al de producción. En este caso, se utilizará una computadora con sus propios elementos software y hardware antes de proporcionarle distribución. Será preciso verificar y testear el correcto funcionamiento en este entorno como paso previo al despliegue en producción.

#### **6.4. CII18**

Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.

Analizando la legislación vigente en diversos países podemos concluir que las normas jurídicas aplicables están dirigidas a proteger la utilización abusiva y/o fraudulenta de la información reunida y procesada mediante el uso de equipos informáticos.

En alguna de estas normas se ha previsto formar órganos especializados que protegen los derechos de los ciudadanos susceptibles de sufrir dicho daño.

Desde hace aproximadamente diez años, la mayoría de los países europeos han hecho todo lo posible para incluir dentro de la ley la conducta punible penalmente, como el acceso ilegal a sistemas de cómputo o el mantenimiento ilegal de tales accesos, la difusión de virus o la interceptación de mensajes informáticos.

En la mayoría de las naciones occidentales existen normas similares a los países europeos. Todos estos enfoques están inspirados por la misma preocupación de contar con comunicaciones electrónicas, transacciones e intercambios tan confidenciales y seguros como sea posible.

Dar una concreción sobre delitos informáticos no constituye una labor sencilla, y

esto en razón de que su misma denominación alude a una situación muy especial, ya que para hablar de "delitos" en el sentido de acciones tipificadas o contempladas en textos jurídico-penales, se requiere que la expresión "delitos informáticos" este consignada en los códigos penales, lo cual en nuestro país, al igual que en muchos otros, no ha sido objeto de tipificación aún. Sin embargo, muchos especialistas en derecho informático emplean esta alusión a efectos de una mejor conceptualización.

#### **6.4.1. Normativa y regulación de la informática en el ámbito internacional .**

En el contexto internacional, son pocos los países que cuentan con una legislación apropiada. Entre ellos, destacan, Estados Unidos, Alemania, Austria, Gran Bretaña, Holanda, Francia, España, Argentina y Chile.

Por esta razón a continuación se mencionan algunos aspectos relacionados con la ley en los diferentes países, así como con los delitos informáticos que persigue.

#### **6.4.2. Estados Unidos**

Este país adoptó en 1994 el Acta Federal de Abuso Computacional que modificó al Acta de Fraude y Abuso Computacional de 1986.

Con la finalidad de eliminar los argumentos hipertécnicos acerca de qué es y que no es un virus, un gusano, un caballo de Troya y en que difieren de los virus, la nueva acta proscribire la transmisión de un programa, información, códigos o comandos que causan daños a la computadora, a los sistemas informáticos, a las redes, información, datos o programas. La nueva ley es un acierto y un gran avance porque está directamente en contra de los actos de transmisión de virus. Asimismo, en materia de estafas electrónicas, defraudaciones y otros actos dolosos relacionados con los dispositivos de acceso a sistemas informáticos, la legislación estadounidense sanciona con pena de prisión y multa, a la persona que defraude a otro mediante la utilización de una computadora o red informática.

En el mes de Julio del año 2000, el Senado y la Cámara de Representantes de este

país, tras un año largo de deliberaciones, establece el Acta de Firmas Electrónicas en el Comercio Global y Nacional. La ley sobre la firma digital responde a la necesidad de dar validez a documentos informáticos, mensajes electrónicos y contratos establecidos mediante Internet, entre empresas y entre empresas y consumidores.

### **6.4.3. Alemania**

Este país sancionó en 1986 la Ley contra la Criminalidad Económica, que contempla los siguientes delitos:

1. Espionaje de datos.
2. Estafa informática.
3. Alteración de datos.
4. Sabotaje informático.

### **6.4.4. Austria**

La Ley de reforma del Código Penal, sancionada el 22 de Diciembre de 1987, sanciona a aquellos que con dolo causen un perjuicio patrimonial a un tercero influyendo en el resultado de una elaboración de datos automática a través de la confección del programa, por la introducción, cancelación o alteración de datos o por actuar sobre el curso del procesamiento de datos. Además contempla sanciones para quienes comenten este hecho utilizando su profesión de especialistas en sistemas.

### **6.4.5. Gran Bretaña**

Debido a un caso de hacking en 1991, comenzó a regir en este país la Computer Misuse Act (Ley de Abusos Informáticos). Mediante esta ley el intento, exitoso o no, de alterar datos informáticos es penado con hasta cinco años de prisión o multas. Esta legislación tiene un apartado que especifica la modificación de datos sin autorización.

#### **6.4.6. Holanda**

En la fecha del 10 de Marzo de 1993 entró en vigencia la Ley de Delitos Informáticos. En la misma se penaliza los siguientes delitos:

1. El hacking.
2. El preacking (utilización de servicios de telecomunicaciones evitando el pago total o parcial de dicho servicio).
3. La ingeniería social (arte de convencer a la gente de entregar información que en circunstancias normales no entregaría).
4. La distribución de virus.

#### **6.4.7. Francia**

En enero de 1988, este país dictó la Ley relativa al fraude informático, en la que se consideran aspectos como:

1. Intromisión fraudulenta que suprima o modifique datos.
2. Conducta intencional en la violación de derechos a terceros que haya impedido o alterado el funcionamiento de un sistema de procesamiento automatizado de datos.
3. Conducta intencional en la violación de derechos a terceros, en forma directa o indirecta, en la introducción de datos en un sistema de procesamiento automatizado o la supresión o modificación de los datos que éste contiene, o sus modos de procesamiento o de transmisión.
4. Supresión o modificación de datos contenidos en el sistema, o bien en la alteración del funcionamiento del sistema (sabotaje).

### **6.4.8. Chile**

Chile fue el primer país latinoamericano en sancionar una Ley contra delitos informáticos, la cual entró en vigencia el 7 de junio de 1993. Esta ley se refiere a los siguientes delitos:

1. La destrucción o inutilización de los datos contenidos dentro de una computadora es castigada con penas de prisión. Asimismo, dentro de esas consideraciones se encuentran los virus.
2. Conducta maliciosa tendente a la destrucción o inutilización de un sistema de tratamiento de información o de sus partes componentes o que dicha conducta impida, obstaculice o modifique su funcionamiento.
3. Conducta maliciosa que altere, dañe o destruya los datos contenidos en un sistema de tratamiento de información.

### **6.4.9. Normativa y regulación de la informática en el ámbito europeo**

El principal esfuerzo europeo por regular el tema de los delitos informáticos dio como resultado el "Convenio sobre la Ciberdelincuencia", de 21 de noviembre de 2001. Este documento fue firmado por los representantes de cada país miembro del Consejo de Europa, aunque su eficacia depende de su posterior refrendo por los órganos nacionales de cada país firmante. El citado "Convenio sobre la Ciberdelincuencia" permitió la definición exacta y la concreción de los delitos informáticos y algunos elementos relacionados con éstos, tales como "sistemas informáticos", "datos informáticos", o "proveedor de servicios". Estos delitos informáticos fueron clasificados en cuatro grupos:

1. Delitos contra la confidencialidad, la integridad y la disponibilidad de los datos y sistemas informáticos.
  - Acceso ilícito a sistemas informáticos.
  - Interceptación ilícita de datos informáticos.

- Interferencia en el sistema mediante la introducción, transmisión, provocación de daños, borrado, alteración o supresión e estos.
  - Abuso de dispositivos que faciliten la comisión de delitos.
2. Delitos informáticos
- Falsificación informática que produzca la alteración, borrado o supresión de datos informáticos que ocasionen datos no auténticos.
  - Fraudes informáticos.
3. Delitos relacionados con el contenido.
- Delitos relacionados con la pornografía infantil.
4. Delitos relacionados con infracciones de la propiedad intelectual y derechos afines.
- Conviene destacar que en el “Convenio sobre la Ciberdelincuencia” se encomienda a cada Parte que tome las medidas necesarias para tipificar como delito en su derecho interno cada uno de los apartados descritos en cada categoría.

En la Disposición 14221 del BOE núm. 226 de 2010, encontramos el Instrumento de Ratificación del Convenio sobre la Ciberdelincuencia, hecho en Budapest el 23 de noviembre de 2001.

#### **6.4.10. Normativa y regulación de la informática en el ámbito nacional**

Ley Orgánica de Protección de datos de carácter personal: régimen sancionador aplicable (BOE no298 de 14/XII/99 que publicó la Ley Org. 15/1999 de 13 de Dic.)

Objeto: Proteger y garantizar las libertades públicas y derechos fundamentales de las personas, especialmente su HONOR e INTIMIDAD personal y familiar.

Aspectos de interés: Serán responsables: “Los responsables de los ficheros o de los tratamientos” y “los encargados de los tratamientos”

#### Tipos de Infracciones:

1. Leves (art.44.2): multas de 100.000 a 10M pts. Ej. Rectificar datos o no comunicarlos a la Agencia de Protección Datos.
2. Graves (art.43): multas de 10M a 50M pts: Ej. No mantener sistemas de seguridad, obstrucción a inspección, uso en provecho propio.
3. Muy Graves (art.45): multas de 50M a 100M ("Conductas reprochables"). Ej. Vulnerar a propósito el secretismo, etc.
4. Ley 7/1998 de 13 de Abril que regula las condiciones generales de contratación.
5. R.D. 1906/1999 de 17/XII que regula la contratación telefónica.
6. R.D. Ley 14/1999 de 17/XII sobre Firma Electrónica (BOE No224 de 18/XII).
7. Firma electrónica: Dispositivo electrónico que permite la identificación del signatario de las operaciones realizadas por Internet.
8. Identifica: El firmante (autenticación) y evita el retracto (no repudio).
9. Código Penal:
  - Ley Orgánica 10/1995 de 23/XI. Tipifica delitos y faltas por el uso de la informática, concretamente contra la Intimidad, Patrimonio, Socio-económicos y Propiedad Intelectual.
  - Título X: "Delitos contra la intimidad, derecho a la propia imagen y la inviolabilidad del Domicilio".
10. Apoderarse de papeles, e-mails, mensajes, otros.
11. Cracks: delitos.
12. Obtener datos de terceros.

#### Recomendaciones de la APD:

1. Información en la recogida de datos:
  - Cuando suministre datos personales a cualquier organización (proveedores de acceso, proveedores de contenido, vendedores a través de comercio electrónico, etc.) sea consciente de a quién se los facilita y con qué finalidad.

- Procure averiguar la política de sus proveedores y administradores de listas y directorios en lo que se refiere a venta, intercambio o alquiler de los datos que les suministra. Solicite que sus datos personales no vayan unidos a su identificación de acceso a Internet.

## 2. Finalidad para la que se recogen los datos:

- Desconfíe si los datos que le solicitan son excesivos para la finalidad con la que se recogen o innecesarios para el servicio que se le presta.
- Tenga en cuenta que cuando introduce su dirección de correo electrónico en un directorio, lista de distribución o grupo de noticias, dicha dirección puede ser recogida por terceros para ser utilizada con una finalidad diferente, como por ejemplo, remitirle publicidad no deseada.
- Cuando navegue por Internet, sea consciente de que los servidores Web que visita pueden registrar tanto las páginas a las que accede como la frecuencia y los temas o materias por las que busca, aunque no le informen de ello. Asimismo, su pertenencia a determinados grupos de noticias y listas de distribución puede contribuir a la elaboración de perfiles más o menos detallados sobre su persona. En el caso de que no desee dejar constancia de sus actividades en la red, utilice los mecanismos para preservar el anonimato que se describen en el cuerpo de este documento.

## 3. Seguridad en el intercambio de datos:

- Utilice, siempre que sea posible, las últimas versiones de los programas navegadores, ya que cada vez suelen incorporar mejores medidas de seguridad. Considere la posibilidad de activar en dichos programas las opciones que alerten sobre los intercambios de datos no deseados y no rellene aquellos datos que no desee hacer públicos (por ejemplo, dirección de correo electrónico, nombre, apellidos, etc.).
- No realice transacciones comerciales electrónicas a través de proveedores con sistemas inseguros o no fiables. Consulte el manual de su navegador para



averiguar cómo informa de que se ha establecido una conexión con un servidor seguro.

- Recuerde que existen sistemas de dinero electrónico que preservan el anonimato de sus compras en Internet.
  - Utilice los mecanismos de seguridad que tenga a su alcance para proteger sus datos de accesos no deseados. El medio más fiable para conseguirlo es el cifrado de los mismos.
  - Salvo que se utilicen mecanismos de integridad, autenticación y certificación (firma digital, notarios electrónicos, etc.) no confíe ciegamente en que la persona u organización que le remite un mensaje es quien dice ser y en que el contenido del mismo no se ha modificado, aunque esto sea en la mayoría de las ocasiones.
4. Siempre que se le soliciten datos personales que no esté obligado legalmente a suministrar, sopesa los beneficios que va a recibir de la organización que los recoge frente a los posibles riesgos de utilización irregular de los mismos.
  5. Ante cualquier duda sobre la legalidad de la utilización de sus datos de carácter personal, póngase en contacto con la Agencia de Protección de Datos.
  6. La protección jurídica de programas de ordenador. Piratería informática.
    - El Real Decreto Legislativo 1/1996, por el que se aprueba el Texto Refundido sobre Propiedad Intelectual, la protección jurídica de los programas de ordenador, antes regulada por la Ley de Protección Jurídica de Programas de Ordenador y por la Ley de Propiedad Intelectual, crea un marco jurídico en contra de la piratería informática.
    - El Texto Refundido desarrolla una serie de medidas para combatir la piratería informática, como la posibilidad de que los fabricantes de programas de ordenador soliciten a la justicia española la realización de un registro sorpresa en empresas en las que existan sospechas fundadas o evidencias de delito. España es uno de los países en los que se puede acudir a esta medida cautelar. De esta manera se erradica la posibilidad de que los presuntos infractores puedan destruir las pruebas existentes, lo cual, indudablemente ocurrirá si se les notifica por adelantado la realización de un registro.

### **¿En qué casos se infringe la Ley?**

- Al copiar o distribuir un programa de ordenador o la documentación que le acompaña, incluidas aplicaciones, datos, códigos y manuales, sin permiso expreso o licencia del propietario de los derechos de explotación.
- Al utilizar un programa sin la correspondiente licencia o autorización del fabricante, con independencia de que se utilice en un solo ordenador o en varios de forma simultánea.
- Al utilizar programas de ordenador en un número de copias superior al autorizado por el fabricante en sus contratos o licencias de uso.
- En empresas y demás organizaciones, al fomentar, consciente o inconscientemente, permitir, obligar o presionar a los empleados a realizar o distribuir copias no autorizadas del programa.
- Al efectuar copias no autorizadas porque alguien lo requiere u obliga a ello. Al ceder o prestar el programa de forma que pueda ser copiado o al copiarlo mientras está en su posesión en calidad de cedido o prestado.
- Al crear, importar, poseer o negociar con artículos destinados a burlar o neutralizar cualquier medio técnico aplicado para proteger el programa de ordenador.

### **Medidas Judiciales.**

Si finalmente existe evidencia de delito, las medidas judiciales que pueden adoptarse son:

1. Solicitar al Juez un registro sorpresa de las instalaciones del presunto infractor, tanto por la vía civil, como por la penal.
2. Solicitar al Juez la adopción urgente de medidas cautelares de protección.
3. Exigir indemnizaciones acordes con los daños materiales y morales causados.
4. El cierre del centro de actividad del infractor.
5. El secuestro de todos aquellos medios destinados a suprimir los dispositivos técnicos que protegen un programa desarrollado y comercializado por un fabricante de programas.

## **6.5 TFG01**

**Ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas de la Ingeniería en Informática de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.**

Esta competencia se logra al concluir el presente trabajo y al culminar el desarrollo de aplicación a la que hace referencia.

La captura de requisitos, realizar análisis, diseño de interfaces, capacidad de adaptarse a un nuevo lenguaje y entorno de desarrollo nunca visto, pertenece evidentemente a ámbitos y competencias de las tecnologías específicas de la Ingeniería Informática.

## 7 APORTACIONES

Es necesario comprender cómo evaluar a un sujeto que realice la prueba así como en qué consiste y qué aporta cada prueba a realizar.

Este trabajo de fin de grado aporta una herramienta digital neuropsicológica usable para la detección de disfunciones cerebrales que afecta a la atención selectiva y a la distractibilidad en general. Pudiendo, incluso, y bajo la supervisión de un experto, ayudar a localizar un daño en la corteza orbitaria del cerebro humano, concretamente en la corteza orbitofrontal para un caso específico de problemas en la toma de decisiones como son los sujetos incapaces de inhibir respuestas incorrectas. Con esta herramienta, se pretende mejorar y/o facilitar el diagnóstico de este daño cerebral.

En la actualidad no existe una herramienta informatizada que cubra esta necesidad, siendo lo más parecido algunos vídeos (ej: <http://www.youtube.com/watch?v=rbVmHuWksOY>) donde van apareciendo nombres de colores pintados de otro color y evaluando los errores tú mismo. Con la informatización de esta herramienta a través de la aplicación SED, conseguimos minimizar los errores ya que la información viene dividida en 2 pantallas: una para el evaluador y otra para el evaluado, dando en una de ellas lo que debería responder el sujeto que realiza la prueba y la otra donde se ejecuta la prueba en sí. Este método, reduce, claramente, la posibilidad de error por parte del experto, ya que no sólo no se evalúa el mismo sujeto que realiza la prueba sino que el profesional que valora sólo tiene que comparar lo que responde el evaluado con la palabra que aparece en su pantalla. Además, SED permite realizar 2 tipos de pruebas diferentes así como configurar los parámetros de cada test, lo que permite una adaptabilidad total a cada tipo de paciente, necesario en el test de Stroop.

Los profesionales del sector, se encontrarán con una herramienta sencilla, intuitiva y útil que agilizará y mejorará las pruebas para este tipo de diagnósticos en todos los rangos de edad y tipos de personas.

Aunque SED no fue pensada, inicialmente, para detectar el trastorno por déficit de atención con hiperactividad (TDAH), autismo, trastornos obsesivo-compulsivo o síndrome de Tourette, bien es cierto que se puede aplicar para su detección como se hace comúnmente con el test de Stroop tradicional.

Resumiendo, SED es una herramienta asistida por computador que, en las manos apropiadas, servirá para detectar personas con daño cerebral, siendo capaz, incluso de distinguir la localización de este daño.

## 8 NORMATIVA Y LEGISLACIÓN

Se incluye a continuación la legislación vigente que afecta a este trabajo de fin de grado:

### 8.1. *Software libre*

El software libre (en inglés "free software", aunque esta denominación también se confunde a veces con "gratis" por la ambigüedad del término en el idioma inglés) es la denominación del software que respeta la libertad de los usuarios sobre un producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado y redistribuido libremente. Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar el software y distribuirlo modificado.

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito" (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en ocasiones el código fuente; no obstante, este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

Tampoco debe confundirse software libre con "software de dominio público". Éste último es aquel software que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, porque pertenece a todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este software sería aquel cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado, tras un plazo contado desde la muerte de este, habitualmente 70 años. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es del dominio público.

Las aplicaciones de software libre utilizadas son:

- Libreoffice Wirter: Versión 4.0.22 de procesador de texto.
- VirtualBox version 4.3.1: Bajo la licencia GPL 2.

- FireFox: Su código fuente es software libre, publicado bajo una triple licencia GNU GPL, GNU LGPL o Licencia Pública de Mozilla.
- Linux Ubuntu versión 13.04

## **8.2. Licencias de software**

Una licencia de software otorga al usuario derecho legal a utilizar un software. Por cada programa de software de Microsoft que se utiliza, se otorga una licencia al usuario y ésta se documenta en el Contrato de Licencia de Usuario Final (CLUF). Un usuario de software, necesita una licencia. El acuerdo de licencia da al usuario el derecho de utilizar el software. El software está protegido por la ley de derechos de autor, que establece que el producto no se puede copiar sin autorización del dueño de derechos de autor. Hay maneras diferentes de adquirir una licencia de Software Microsoft:

- Producto Empaquetado (Caja): Licencia, CD-Rom y documentación en un paquete.
- Original Equipment Manufacturer (OEM): licencia para software preinstalado en un PC nuevo.
- Licencia por Volumen.

Para el desarrollo del software, en este trabajo de fin de grado se ha utilizado Windows 7, Word del paquete de Microsoft Office y Visual Studio 2010 bajo una licencia del tipo Licencia Original Equipment Manufacturer (OEM):

- Licencia para software preinstalado en un PC nuevo.
- Windows 7 Profesional Virtual: Windows 7 es la versión más reciente de Microsoft Windows, línea de sistemas operativos producida por Microsoft Corporation.
- Visual Studio 2010: Visual Studio 2010 de Microsoft Windows es la antepenúltima versión a día de hoy que circula de esta plataforma de programación. Usa el entorno .Net y ofrece múltiples herramientas para la creación de aplicaciones basadas en Visual Basic.Net, C#, F# y ASP.Net entre otras.

### **8.3. Leyes sobre seguridad**

En lo que se refiere a seguridad de la aplicación resultante de la implementación de este trabajo fin de grado, cabe realizar las siguientes consideraciones:

1. Respecto a la posibilidad de hacer circular información negativa, como virus o gusanos, los equipos de trabajo en los cuales se van a instalar la herramienta disponen de un sistema antivirus. Por otro lado, el software no se instala ni escribe información alguna en disco, minimizando la probabilidad de infección a través de este software.
2. Por último, en dichos equipos de trabajo se recomienda la utilización de las últimas versiones de los Frameworks de Microsoft, ya que cada vez suelen incorporar mejores medidas de seguridad. Por lo tanto no se ve la necesidad de utilizar mecanismos de integridad, autenticación y certificación (firma digital, notarios electrónicos, etc.) puesto que no se gestiona ni solicita dato alguno de carácter personal y es imposible incumplir el artículo 21 de la ley de protección de datos.





## **9 SECTOR DE LA ACTIVIDAD**

El producto está claramente enfocado al sector tecnológico-social, en esencia al sector sanitario.

Es tecnológico porque el manejo y desarrollo de la aplicación es puramente informática y destinada a ordenadores donde corra Windows.

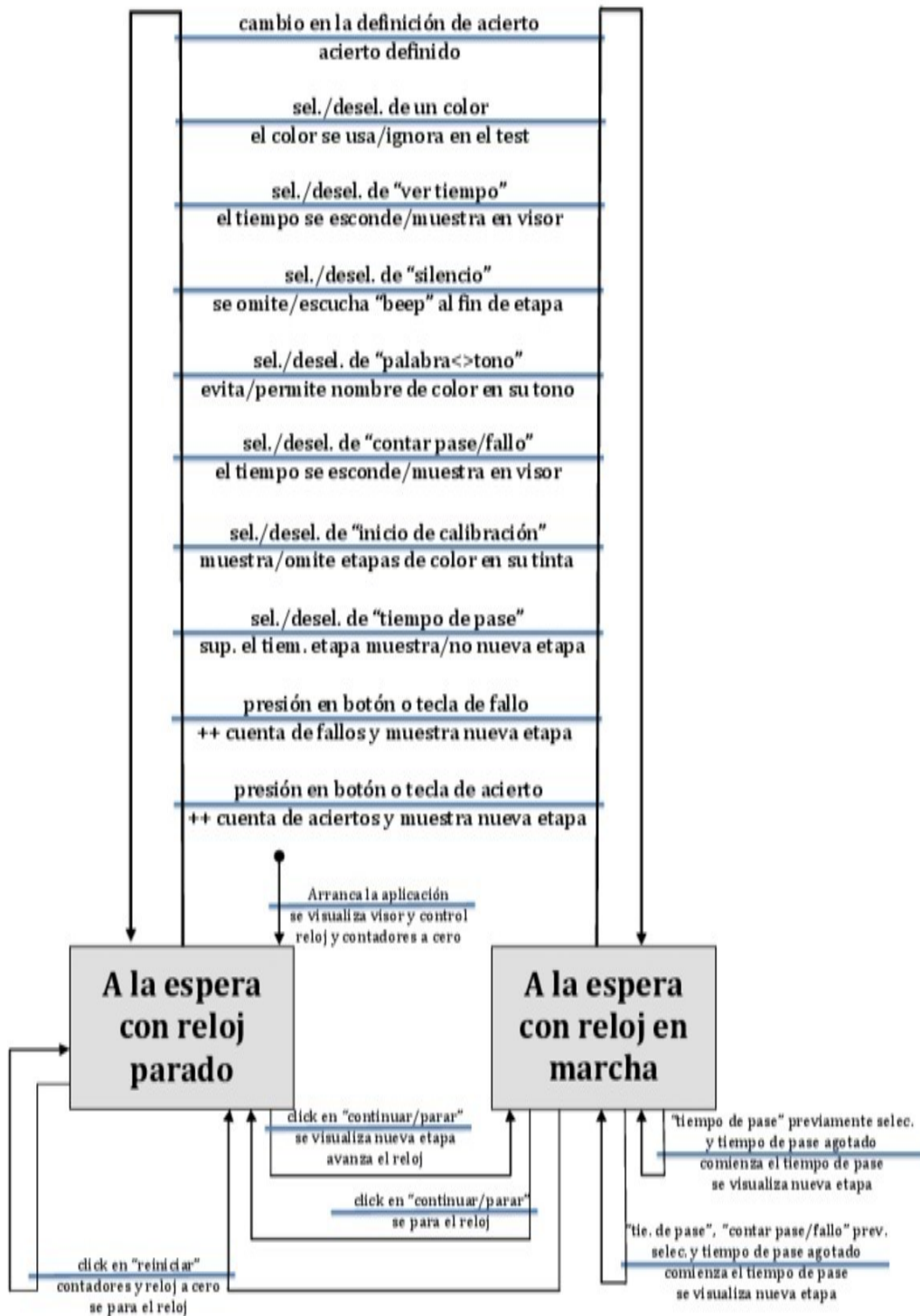
Es social porque ayuda a identificar un daño en una zona localizada del cerebro que provoca problemas difíciles de detectar.

Es sanitario porque apoya a la psicología y neuropsicología en el proceso de identificación de daño cerebral en seres humanos. Además, requiere el manejo de un experto para que pueda interpretar la información de forma eficiente y efectiva.



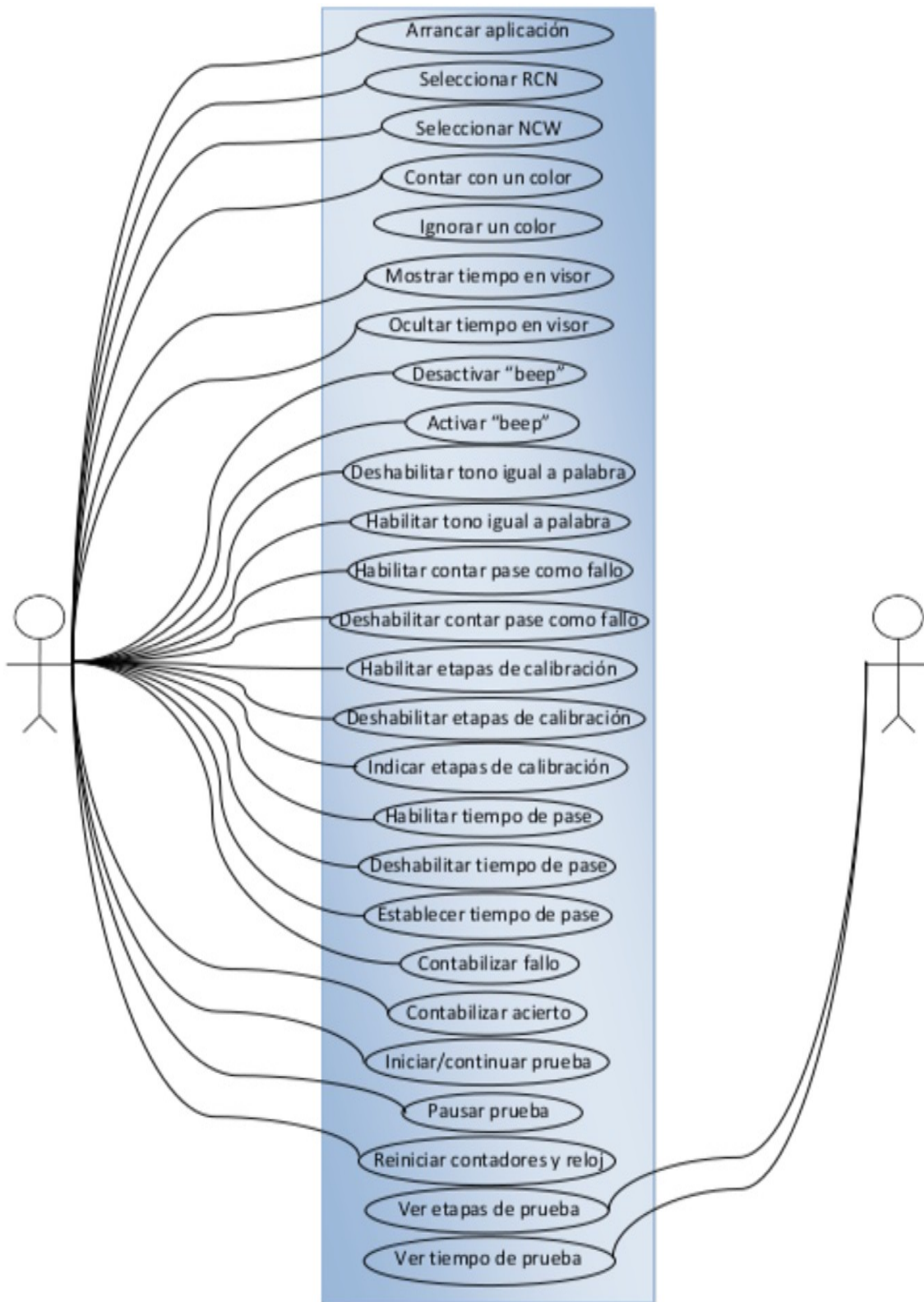
# 10 ANALISIS

## 10.1 DIAGRAMA DE ESTADOS





## 10.2 DIAGRAMA DE CASOS DE USO





### 10.3 COSTE Y ESFUERZO

Para calcular el coste, me he apoyado en un estudio de estimación de coste y esfuerzo del desarrollo del software. Para ello, debemos calcular su tamaño y previamente sus puntos de función, los cuales vienen determinados por las siguientes tablas:

#### 10.3.1 CLASIFICACIÓN DE TRANSACCIONES Y ARCHIVOS

Clasificación	Valores		
	Salidas Externas	Consultas Externas	Entradas Externas
Baja	4	3	3
Media	5	4	4
Alta	7	6	6

Clasificación	Valores		
	Archivo Interno	Lógico	Archivo de Interface Externo
Baja	7		5
Media	10		7
Alta	15		10

<b>Archivo Lógico Interno</b>	10
<b>Entradas Externas</b>	4
<b>Consulta Externa</b>	3
<b>Salida Externa</b>	4
<b>Archivo Interface Externo</b>	5

Puntos de Función sin Ajustar (UFP) = 10 + 4 + 3 + 4 + 5 = 26 puntos de función sin ajustar.



### 10.3.2 CÁLCULO DEL FACTOR DE COMPLEJIDAD

	Factor de Complejidad	Valor
1	Comunicación de Datos	0
2	Proceso Distribuido	0
3	Objetivos de Rendimiento	0
4	Configuración Explotación Compartida	1
5	Tasa de Transacciones	0
6	Entrada de Datos EN-LINEA	5
7	Eficiencia con el usuario final	2
8	Actualizaciones EN-LINEA	0
9	Lógica de proceso interno compleja	1
10	Reusabilidad del código	2
11	Contempla la conversión e instalación	0
12	Facilidad de Operación	0
13	Instalaciones Múltiples	0
14	Facilidad de Cambios	2
	FACTOR DE COMPLEJIDAD TOTAL (FCT)	13

$$\text{TDI} = 0,65 + (0,01 * \text{FCT}) = 0,76$$

Los puntos de función ajustados se calculan mediante la formula:  $\text{PF} = \text{UFP} * \text{TDI} = 26 * 0,78 = 20,28$

Con los puntos de función y el costo por programar en Visual Basic (valor 32), podemos tomar un coste del producto SED en función del valor de sus líneas de código y sus puntos de función, quedando así:

**Tamaño de SED** =  $\text{PF} * \text{Línea Visual Basic} = 20,28 * 32 = 886,16 \Rightarrow$  **648 líneas de código** aproximadamente.

**NOTA:** Recordar que esto es simplemente una estimación y como tal, los resultados son meramente orientativos, pudiendo variar si no se ha estimado con acierto la complejidad del software o la clasificación de transacciones y archivos.

Con la información anterior un valor de productividad por personal y mensual, podemos estimar cuanto tiempo nos llevará realizar el producto SED.

**Esfuerzo** = PF / Productividad (suponiendo una productividad de 8 PF/persona-mes) =  $20,28 / 8 = 2,535$  Persona-mes. Esto quiere decir que una persona tardaría 2 meses y medio en desarrollar la aplicación SED.

**NOTA:** Recordar que esto es simplemente una estimación y como tal, los resultados son meramente orientativos, pudiendo variar si no se ha estimado con acierto la complejidad del software o la clasificación de transacciones y archivos o varía la productividad de una persona.

**Coste** = **Esfuerzo** \* Salario. Si suponemos un salario de 1000 €/mes => Coste de SED =  $2,5 * 1000 = 2500$  € es lo que costaría la mano de obra para desarrollar SED.



## 11 REQUISITOS SOFTWARE

Para poder usar la aplicación, es necesario contar al menos con:

1. Windows XP.
2. Windows Installer 3.1
3. Microsoft Framework 4.0
4. DirectX 9.



## 12 REQUISITOS HARDWARE

Para el correcto funcionamiento de la aplicación, es necesario contar con Microsoft Framework 4.0 instalado. Según su hoja de requisitos de hardware, este framework como mínimo necesita:

1. Una arquitectura de 32 bits.
2. Pentium 1 GHz con 512 Mb RAM
3. Espacio en disco de 850 Mb.

Aunque para la ejecución no requiere espacio en disco, ya que sólo se ejecuta en memoria RAM.



## 13 DESARROLLO

### 13.1 METODOLOGÍA

Para el desarrollo de SED, la metodología de trabajo que se ha seguido es una metodología basada en prototipos denominada Modelo de Prototipos.

**Definición:** El Modelo de prototipos, en Ingeniería de software, pertenece a los modelos de desarrollo evolutivo. El prototipo debe ser construido en poco tiempo, usando los programas adecuados y no se debe utilizar muchos recursos. El diseño rápido se centra en una representación de aquellos aspectos del software que serán visibles para el cliente o el usuario final. Este diseño conduce a la construcción de un prototipo, el cual es evaluado por el cliente para una retroalimentación; gracias a ésta se refinan los requisitos del software que se desarrollará. La interacción ocurre cuando el prototipo se ajusta para satisfacer las necesidades del cliente. Esto permite que al mismo tiempo el desarrollador entienda mejor lo que se debe hacer y el cliente vea resultados a corto plazo.[11]

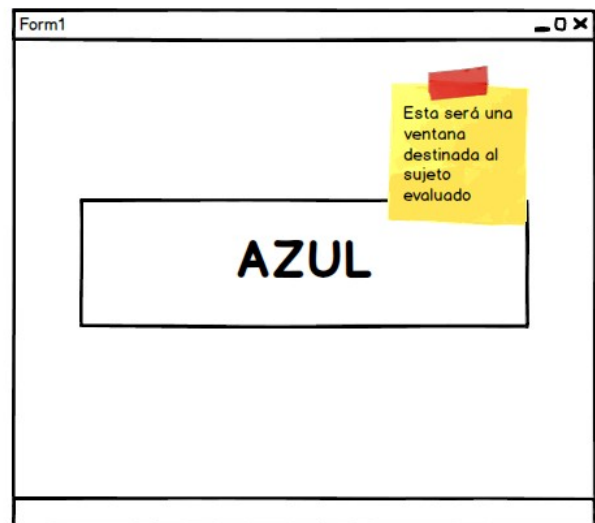
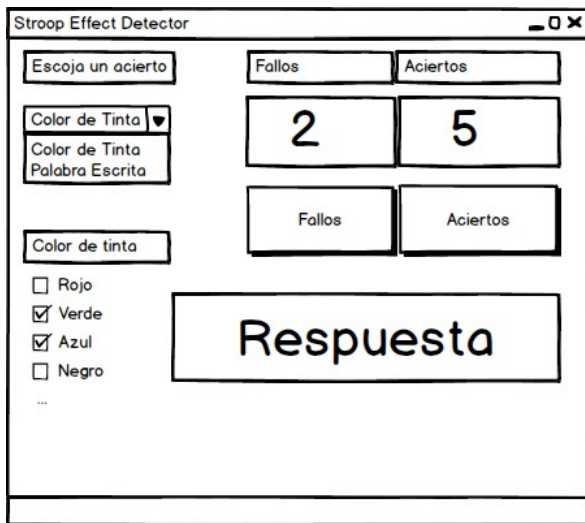
El uso de esta metodología permitió conocer de antemano los requisitos que mejor le venía a los posibles clientes a través de un diseño en papel, dónde indicaban donde querían qué información. Así se comenzó a crear un prototipo básico y funcional. Posteriormente, al modelo se le irán añadiendo gradualmente funcionalidades y corrigiendo errores produciendo una nueva versión, para ello se utilizará un método de desarrollo de progreso evolutivo hasta conseguir un producto viable, acabado y que satisfaga los requisitos propuestos.

Aquí se presentan las versiones que surgieron del desarrollo de SED.

### 13.2 VERSION ALPHA

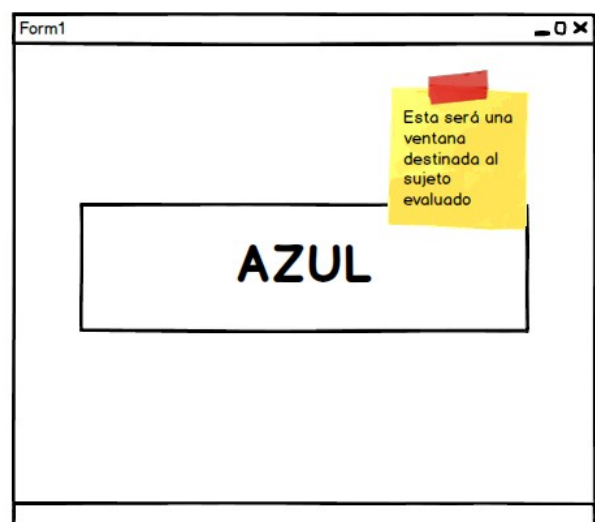
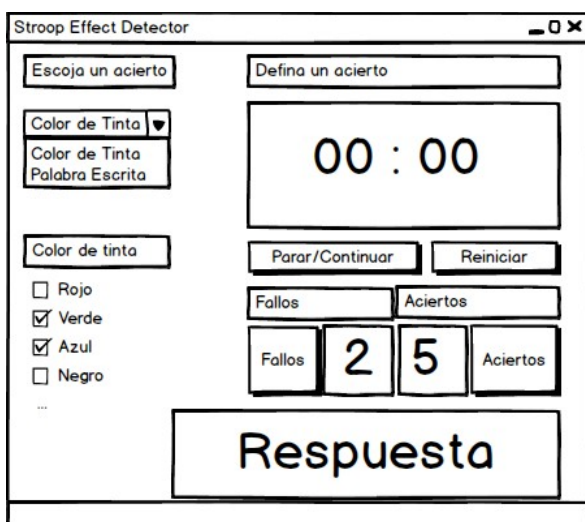
En la primera versión, el producto logra mostrar las 2 ventanas con sus interfaces, una para el evaluador y otra para el evaluado. Funcional sólo la del evaluado, en la cual se muestran palabras con el nombre de un color con tinta de otro color diferente (Ej: Rojo escrito en Verde). Los colores pueden ser elegidos de una lista y se cambia de palabra y color cuando el usuario (evaluador) pulsaba la tecla “flecha derecha”. Actualmente puede ocurrir casos como que una palabra y el color de esa palabra coincidan (Ej: Rojo escrito en Rojo). Ahora veremos el boceto del modelo del prototipo acordado con el cliente:





### 13.3 VERSION BETA 1.0

Primera versión funcional. Mínimo producto viable. El programa, corrige errores de la versión Alpha, como la posibilidad de coincidencia de la palabra con el color (Ej: Rojo escrito en Rojo) dejándolo como opción si el evaluador quiere activarla. Además se añade la funcionalidad de tiempo (incluyendo reiniciar contadores y reloj) y la ventana del evaluador, donde aparece la respuesta que debe dar el evaluado. Veamos como fue el boceto del modelo del prototipo de esta versión acordado con el cliente:



### 13.4 VERSION BETA 2.0

Versión final del prototipo. Corrección de errores de ejecución de la versión Beta 1.0 y añadido las funcionalidades de tiempo de pase, seleccionar el tipo RCN o NCW, habilitar y deshabilitar el pase como fallo, calibración de etapa, contabilizar fallos y aciertos con dispositivo bluetooth y teclado, pausar y reanudar la prueba. Veamos como era el boceto creado después de hablar con el cliente:

The screenshot shows the 'Stroop Effect Detector' application window. It features a control panel on the left with various settings and a central display area. The display area shows a digital timer at '00:00' and a large 'Respuesta' (Response) field. Below the timer, there are buttons for 'Continuar/Parar' (Continue/Stop) and 'Reiniciar' (Restart). The control panel includes checkboxes for ink colors (Rojo, Verde, Azul, Negro), a 'Palabra <-> Tinta' checkbox, and a 'Tiempo en sujeto' checkbox. There are also fields for 'Fallos' (2) and 'Aciertos' (5), and a 'calibrar' button with a '10' dropdown. At the bottom, there are 'opcionX' and 'tiempo' (5 seg / 10 seg) settings.

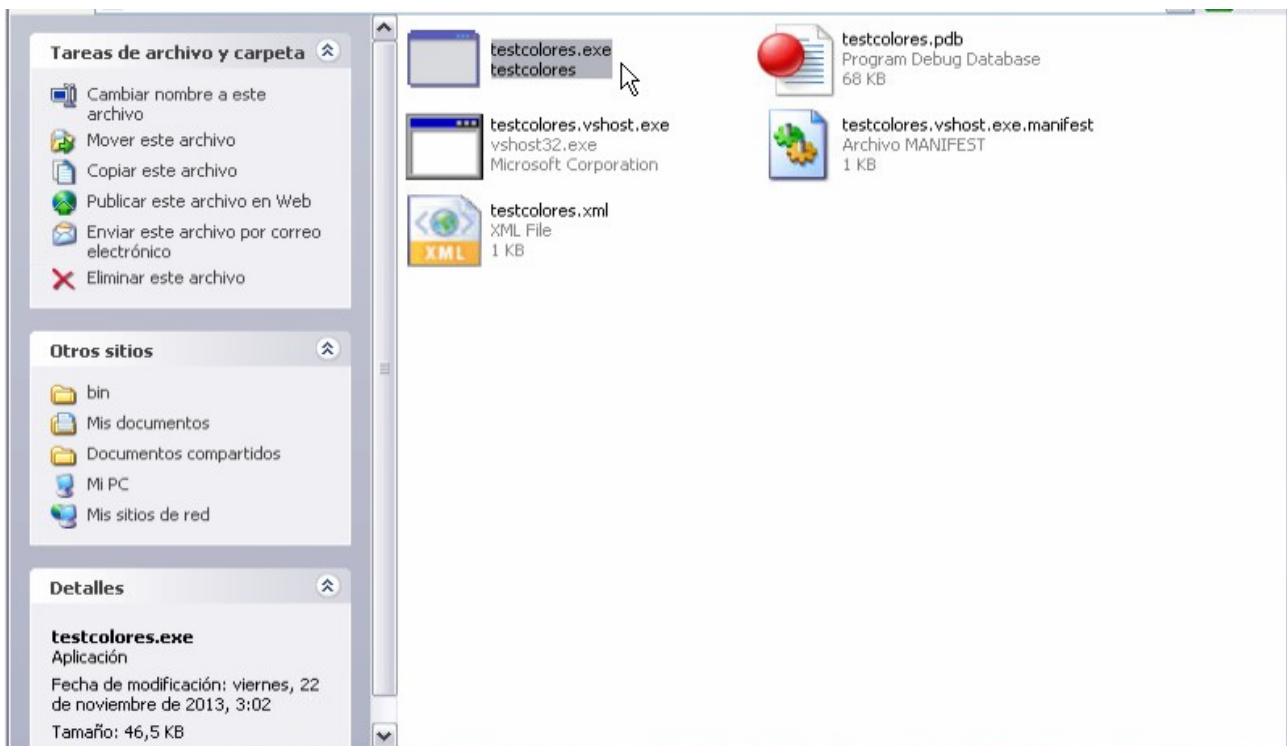
The screenshot shows a window titled 'Form1'. It contains a large rectangular box with the word 'AZUL' written inside. A yellow sticky note is attached to the top right corner of the window, with the text: 'Esta será una ventana destinada al sujeto evaluado'.



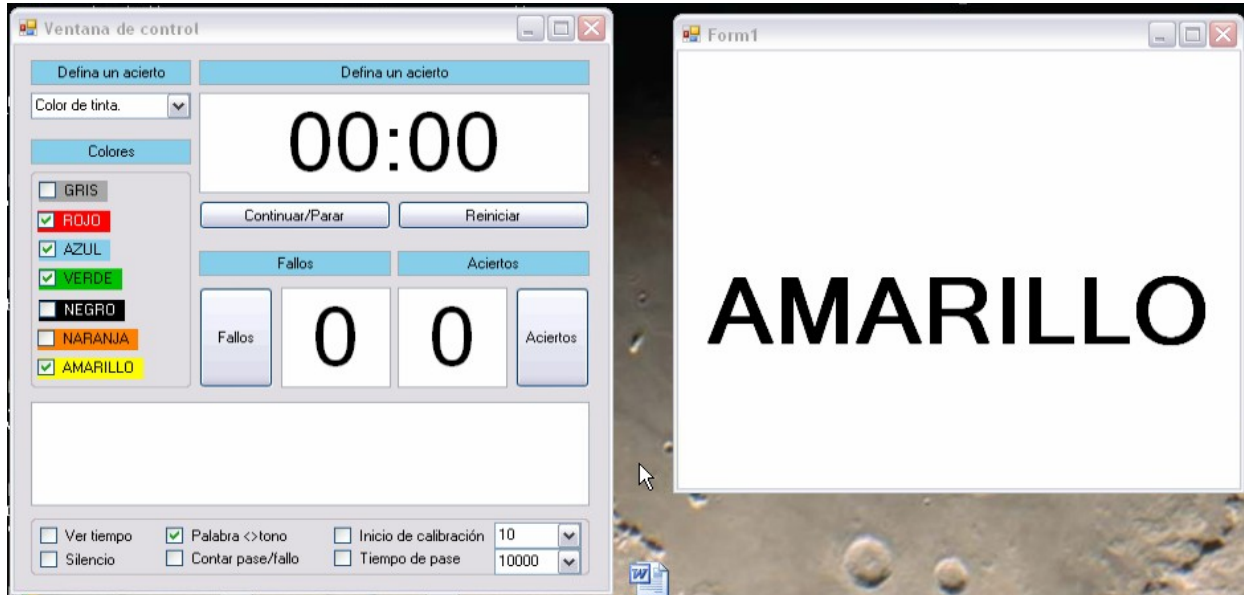
## 14 MANUAL DE USUARIO

### 1. Acceder a la aplicación.

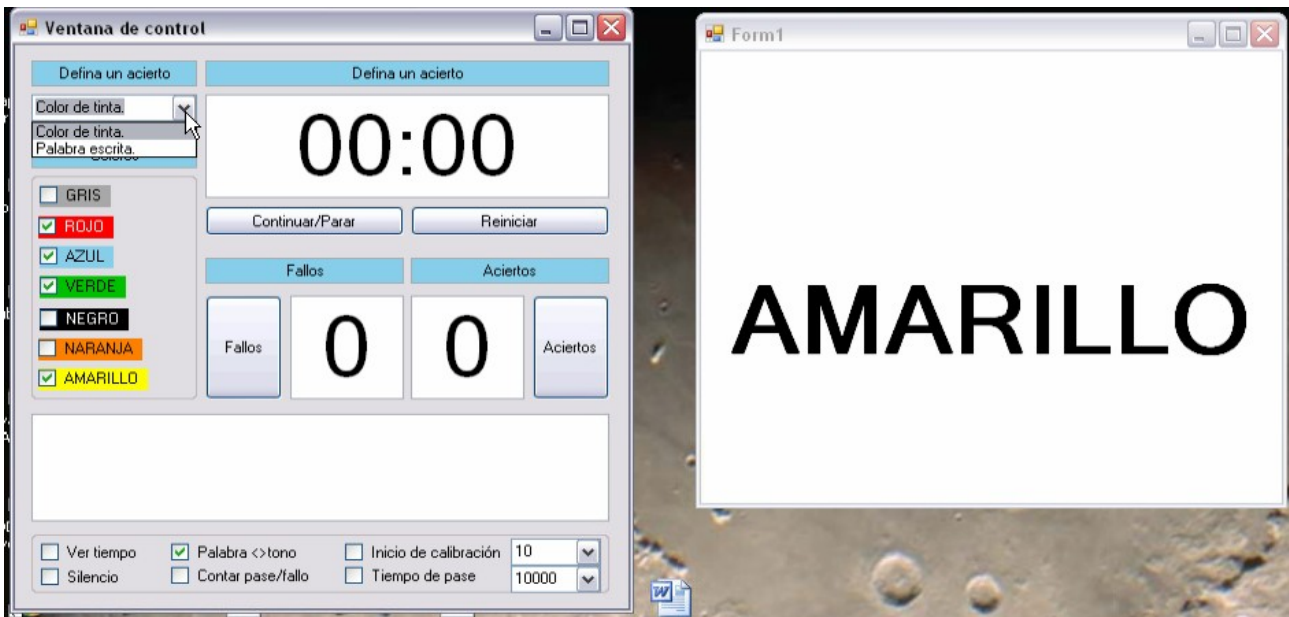
Haga click dos veces sobre el archivo de extensión ".exe" o selecciónelo y presione la tecla "enter".



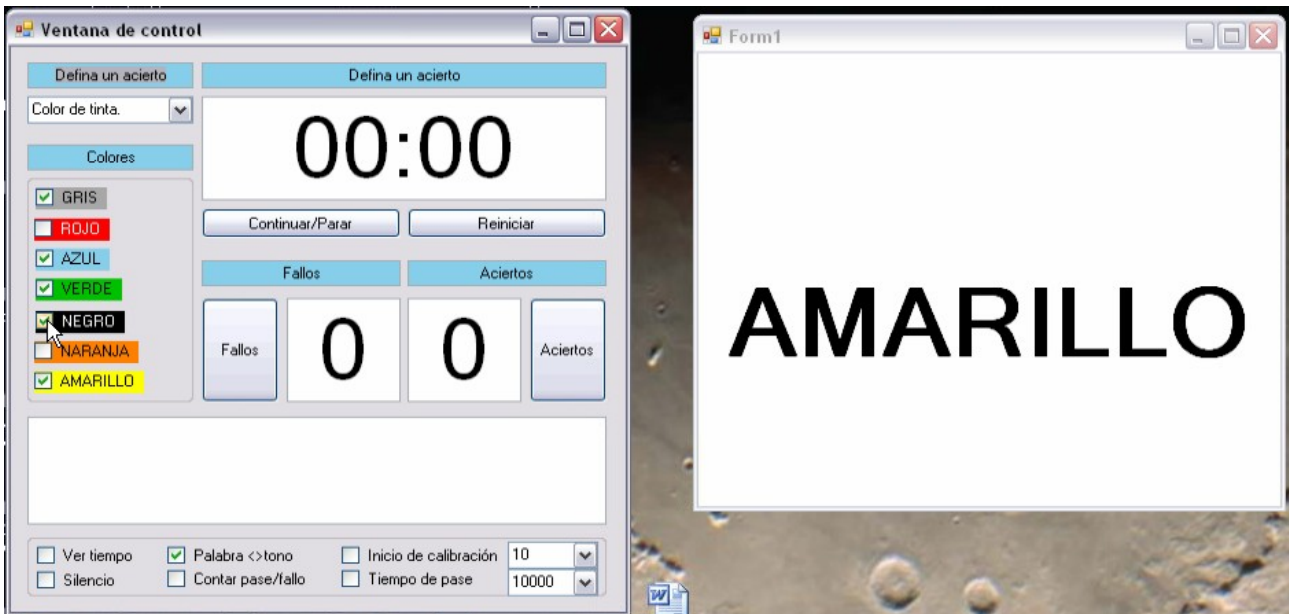
2. Dejar la aplicación lista para trabajar. Realice el paso 1 y dispondrá de la ventana de control y del visor (que podrá ser ubicado en un a pantalla secundaria si se dispone de escritorio extendido).



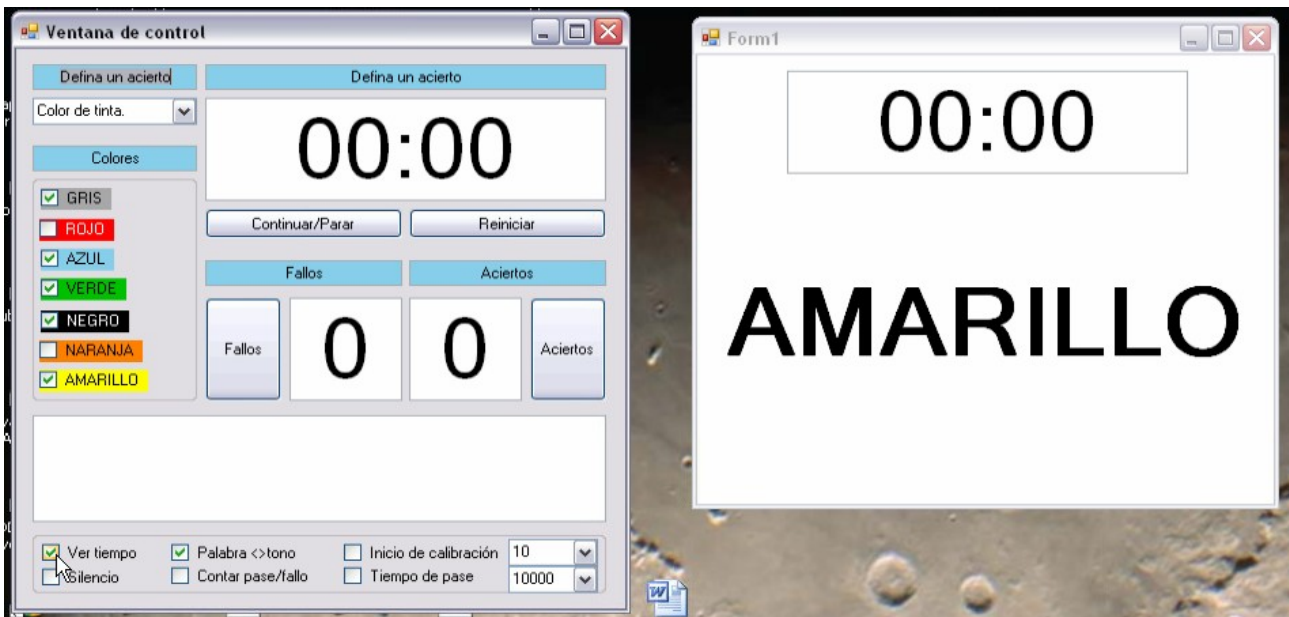
3. Cambiar de prueba RCN/NCW Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) diríjase a la lista desplegable identificada por el texto "defina un acierto" y seleccione en ella si desea entender como correcto la lectura de la palabra o mención del color de la tinta.



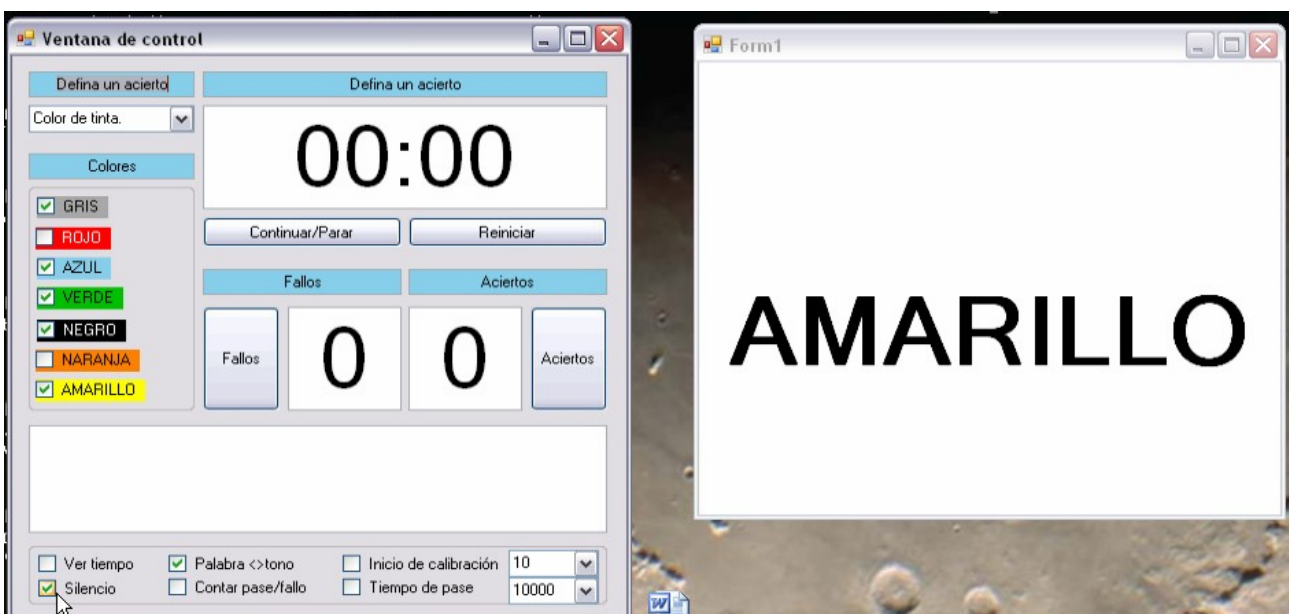
- Elegir colores del test. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) diríjase al marco identificado por el texto "colores" y seleccione de su interior las palabras y tonalidades de tinta que participarán en el test, dificultando o facilitando de ese modo su realización.



5. Mostrar u ocultar el tiempo de prueba en el visor. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) diríjase a la opción identificada por el texto "ver tiempo" y selecciónela o desecciónela en función de si desea que en el visor se muestre o esconda (respectivamente) el tiempo de prueba.



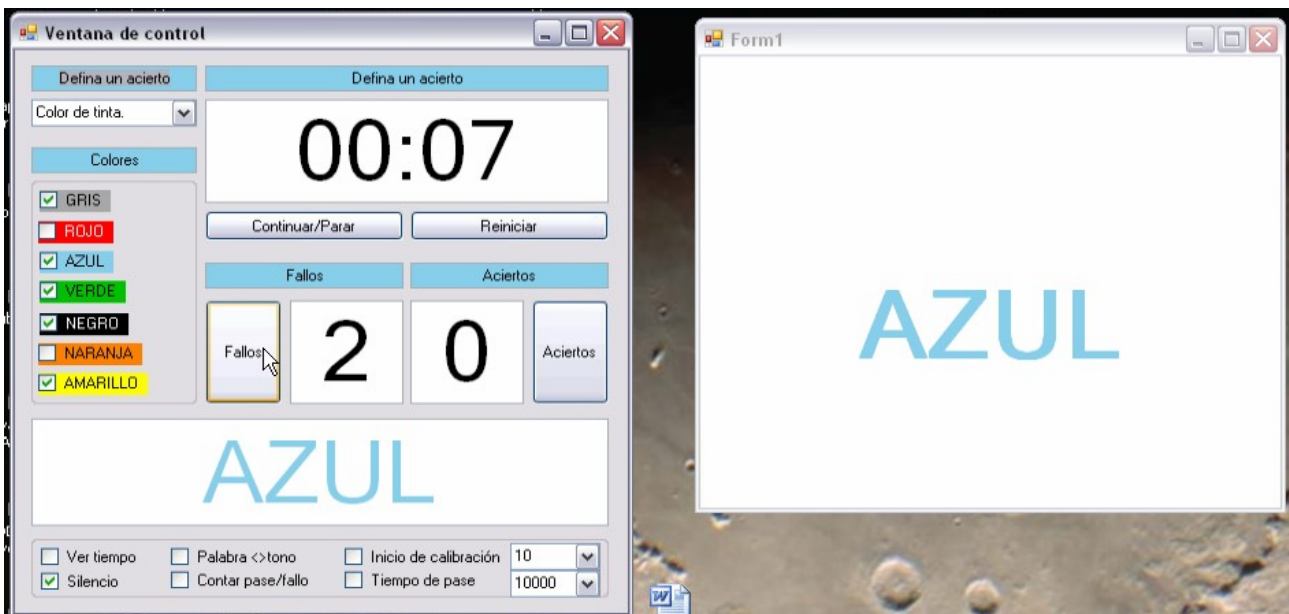
6. Silenciar o no la aplicación. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) diríjase a la opción identificada por el texto "silenciar" y selecciónela o desecciónela en función de si desea silenciar o no (respectivamente) el aviso acústico que se produce al final de una etapa del test.



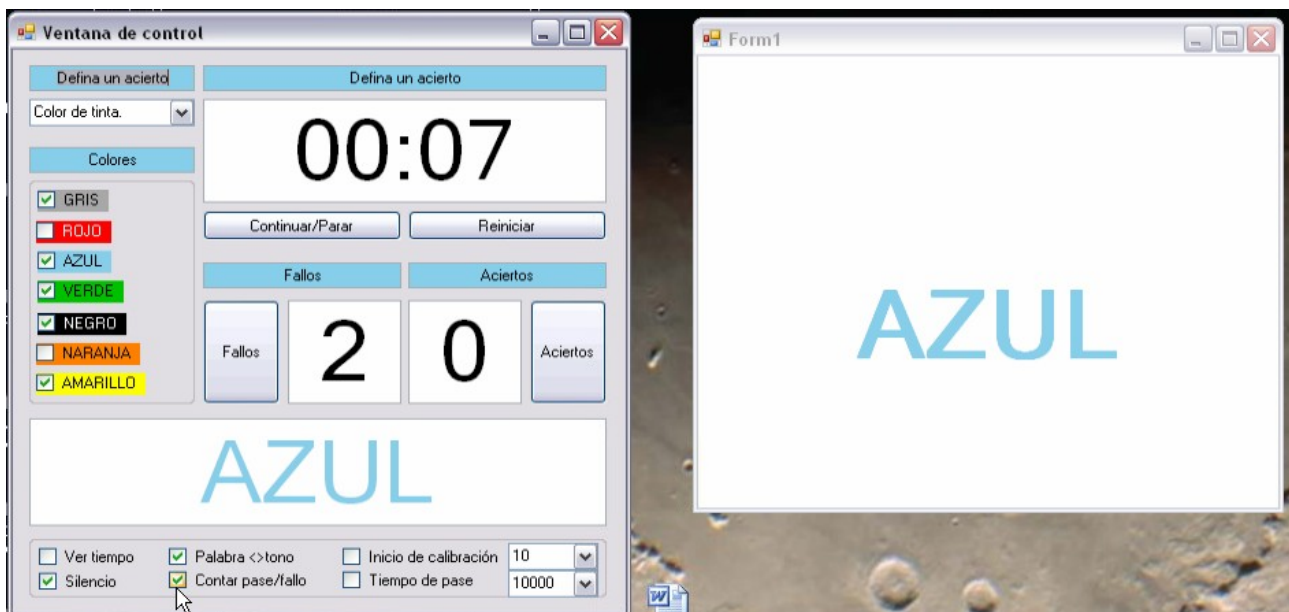


7. Permitir o no la concurrencia del nombre de un color escrito en la tinta de ese color.

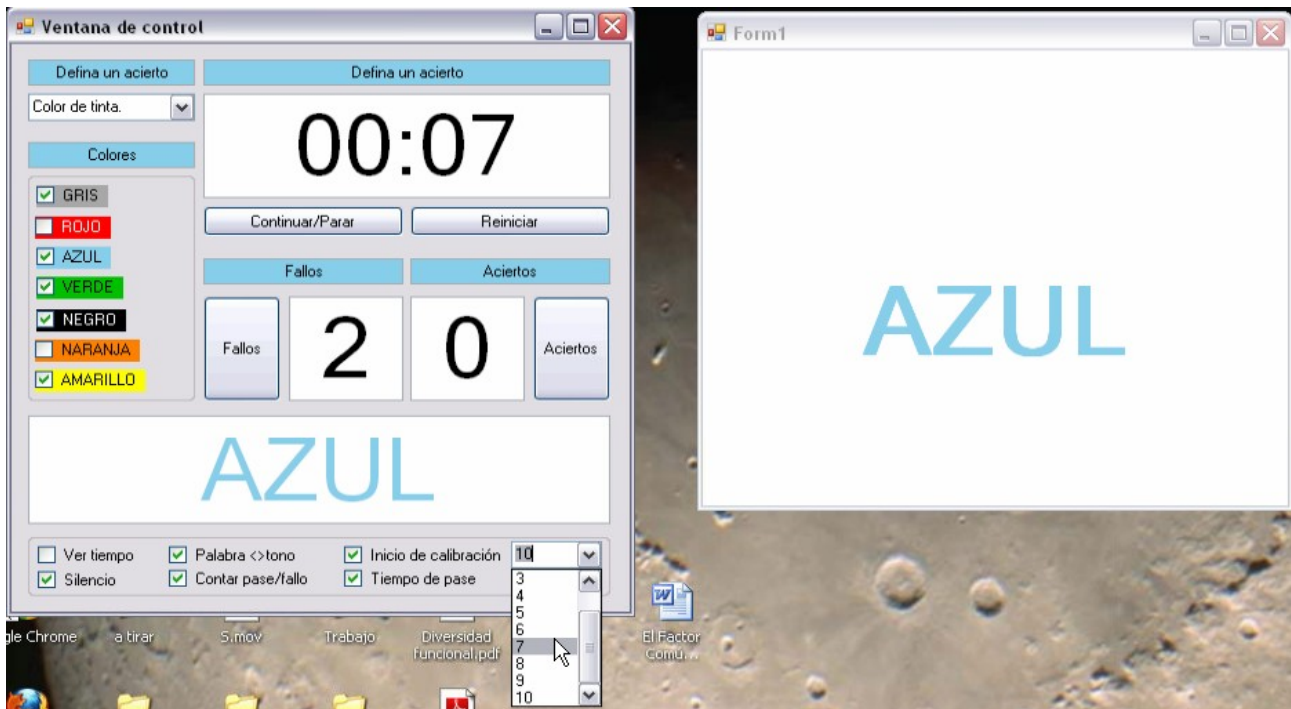
Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) dirijase a la opción identificada por el texto "palabra <> tono" y selecciónela o desecciónela en función de si desea evitar o permitir (respectivamente) que el nombre de un color aparezca en el visor escrito en la tinta a cuyo color hace referencia.



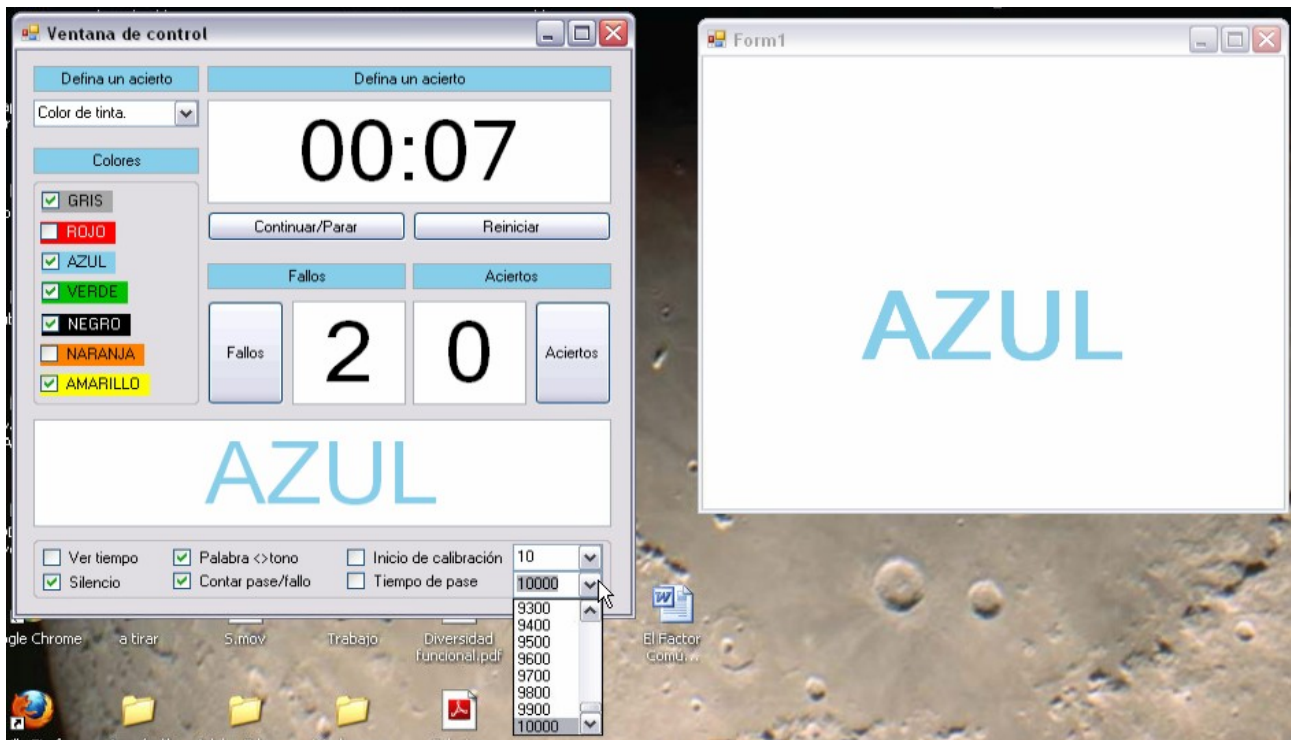
8. Contabilizar o no un "pase de etapa" como una etapa fallida. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) dirijase a la opción identificada por el texto "contar pase/fallo" y selecciónela o desecciónela en función de si desea o no (respectivamente) que en el pase de etapa por exceso de tiempo sea considerado un fallo.



9. Establecer o no un inicio de calibración. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) dirijase a la opción identificada por el texto "inicio de calibración" y selecciónela o desecciónela en función de si desea o no (respectivamente) establecer un número de etapas iniciales cuya palabra mostrada y color de tinta coincidan. Para decidir el número de etapas de calibración, seleccione un valor de la lista desplegable contigua.



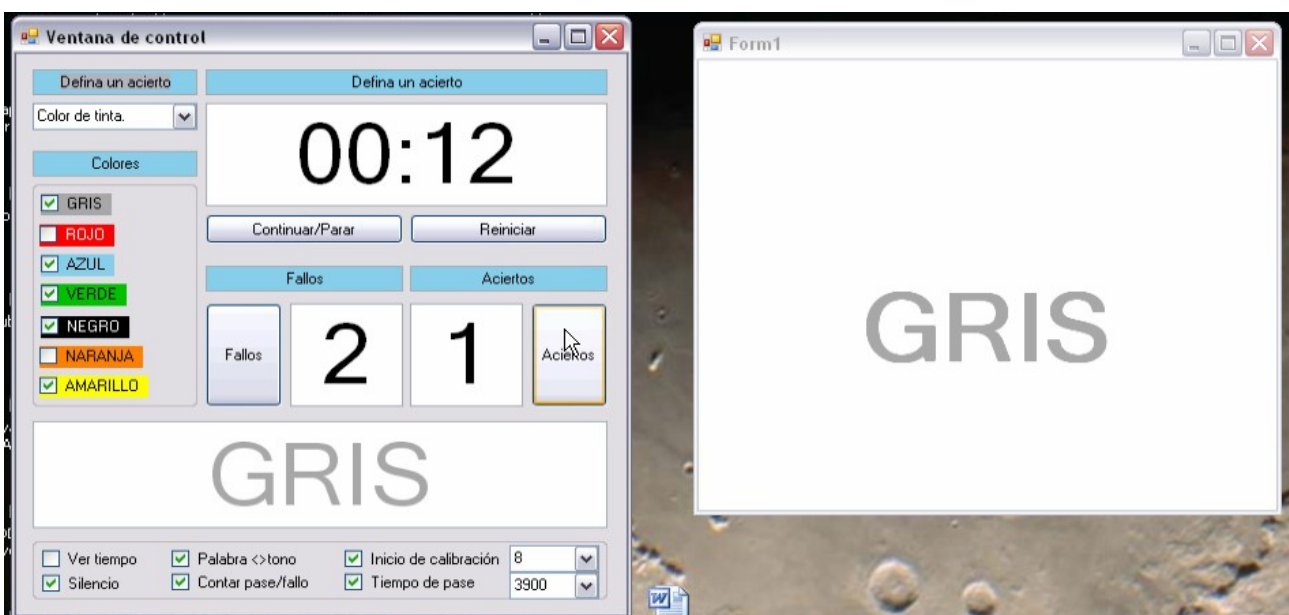
10. Establecer un tiempo de pase de etapa. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) dirijase a la opción identificada por el texto "tiempo de pase" y selecciónela o deselecciónela en función de si desea o (respectivamente) que exista un tiempo máximo de muestra de etapa. Para seleccionar el número de milisegundos que se mostrará una etapa, seleccione un valor de la lista desplegable contigua.



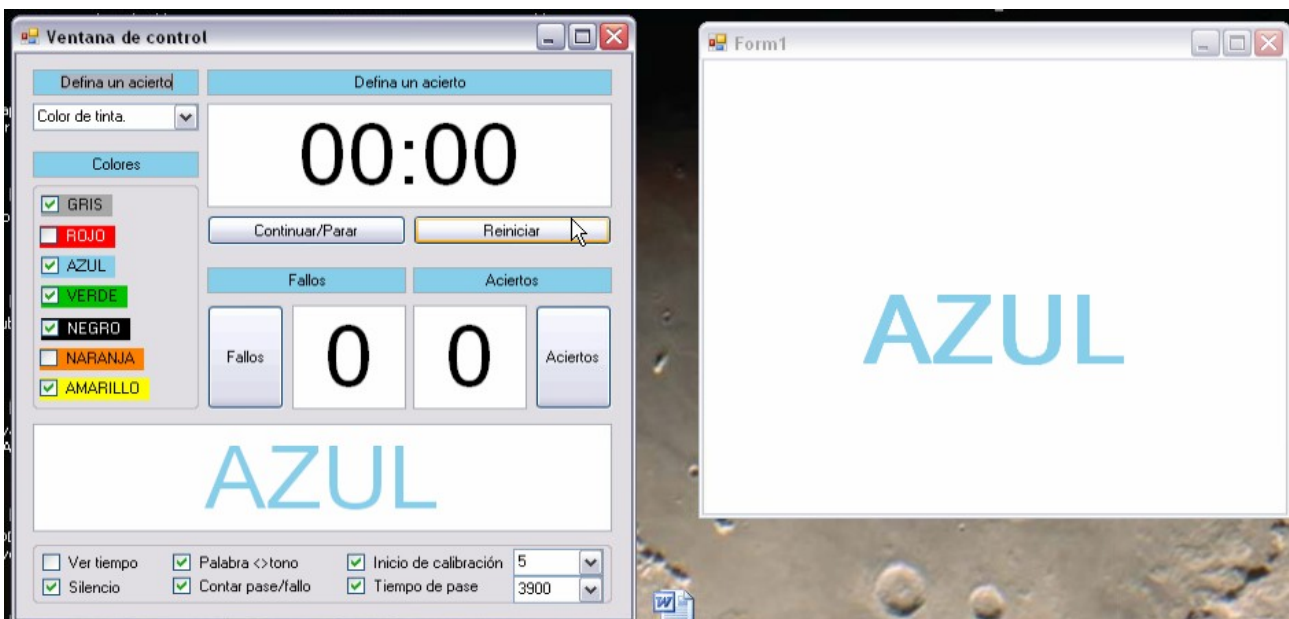
11. Marcar un fallo. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) diríjase al botón identificado por el texto "fallo" y haga click sobre él, consiguiendo aumentar así el número de errores computados. Alternativamente podrá contabilizar un fallo si presiona cualquier tecla del teclado salvo "space", "flecha derecha", "flecha abajo", "enter", "avpag" y "next". Si se usa un mando a distancia de pase de diapositivas, la aplicación marcará un error si se presiona el botón que retrocede a una diapositiva anterior.



12. Marcar un acierto. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) diríjase al botón identificado por el texto "acierto" y haga click sobre él, consiguiendo aumentar así el número de aciertos computados. Alternativamente podrá contabilizar un acierto si presiona las teclas "space", "flecha derecha", "flecha abajo", "enter", "avpag" o "next". Si se usa un mando a distancia de pase de diapositivas, la aplicación marcará un acierto si se presiona el botón que provoca un avance a la diapositiva siguiente.



13. Reiniciar contadores de tiempo y resultado. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) dirijase al botón identificado por el texto "reiniciar" y haga click sobre él para parar posibles ejecuciones en proceso y reiniciar tanto los contadores de respuesta como los de tiempo.



14. Iniciar o pausar un test. Realice el paso 1. A partir de ahora, en cualquier momento (incluso en medio de la ejecución de un test) dirijase al botón identificado por el texto "continuar/parar" y haga click sobre él para iniciar o parar (según la aplicación se encuentre en el paso 2 o esté ejecutando un test respectivamente). Si la aplicación se encontrada parada continuara con una nueva etapa. En caso contrario, se mostrará a la espera de reanudarse o reiniciarse (paso 13).



## 15 REFERENCIAS BIBLIOGRÁFICAS

- [1] Luria AR. Frontal lobe syndromes. Handbook of Clinical Neurology, vol. 2.
- [2] Nauta WJH. The problem of the frontal lobe: a reinterpretation. J Psychiatr Res 1971; 8: 167-187.
- [3] Cavada C, Schultz W. "The mysterious orbital prefrontal cortex". Cerebral Cortex 2000, en prensa.
- [4] Damasio AR, Anderson SW. The frontal lobes. Clinical Neuropsychology. Heilman KM, Valenstein E eds. Nueva York, Oxford University Press, 1993.
- [5] Mesulam M-M. Frontal cortex and behavior. Ann Neurol 1986; 19: 320-325.
- [6] Damasio H, Damasio AR. Lesion analysis in neuropsychology. Oxford University Press, Nueva York, 1989.
- [7] Cummings JL. Frontal-Subcortical circuits and human behavior. Arch Neurol 1993; 50: 873-880.
- [8] [http://centrodeartigos.com/articulos-enciclopedicos/article\\_93703.html](http://centrodeartigos.com/articulos-enciclopedicos/article_93703.html)
- [9] Lhermitte F, Pillon B, Serdaru M. Human autonomy and the frontal lobes. I. Imitation and utilization behavior. A neuropsychological study of 75 cases. Ann Neurol 1986; 19: 335-343.
- [10] Hanninen T, Hallikainen M, Koivisto K, Partanen K, et al. Decline of frontal lobe functions in subjects with age-associated memory impairment. Neurology 1997; 48: 148-153.
- [11] [http://es.wikipedia.org/wiki/Modelo\\_de\\_prototipos](http://es.wikipedia.org/wiki/Modelo_de_prototipos)





## 16 BIBLIOGRAFÍA

García de Jalón Javier, Rodríguez José Ignacio, Brazález Alfonso. Aprenda Visual Basic como si estuviera en primero.

<http://www.aulaclie.es/visualbasic/>

[http://es.wikipedia.org/wiki/Corteza\\_orbitofrontal](http://es.wikipedia.org/wiki/Corteza_orbitofrontal)

[http://es.wikipedia.org/wiki/Efecto\\_Stroop](http://es.wikipedia.org/wiki/Efecto_Stroop)

<http://www.logicortex.com/sobre-el-test-de-stroop-sus-formulas-y-la-interpretacion-de-los-resultados/>

Gómez Beldarrain, Mariam. Síndromes Disejcutivos: Bases, Clínica y Evaluación.



## 17 ANEXO

### 17.1 CODIGO FUENTE

#### 17.1.1 Visor.vb

```
Public Class Visor
End Class
```

#### 17.1.2 VisorDesigner.vb

```
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()> _
Partial Class Visor
    Inherits System.Windows.Forms.Form

    'Form reemplaza a Dispose para limpiar la lista de componentes.
    <System.Diagnostics.DebuggerNonUserCode()> _
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        Try
            If disposing AndAlso components IsNot Nothing Then
                components.Dispose()
            End If
        Finally
            MyBase.Dispose(disposing)
        End Try
    End Sub

    'Requerido por el Diseñador de Windows Forms
    Private components As System.ComponentModel.IContainer

    'NOTA: el Diseñador de Windows Forms necesita el siguiente
    procedimiento
    'Se puede modificar usando el Diseñador de Windows Forms.
    'No lo modifique con el editor de código.
    <System.Diagnostics.DebuggerStepThrough()> _
    Private Sub InitializeComponent()
        Me.TBVisor = New System.Windows.Forms.TextBox()
        Me.TBTime = New System.Windows.Forms.TextBox()
        Me.SuspendLayout()
        '
        'TBVisor
        '
        Me.TBVisor.BorderStyle = System.Windows.Forms.BorderStyle.None
        Me.TBVisor.Font = New System.Drawing.Font("Microsoft Sans Serif",
54.75!, System.Drawing.FontStyle.Bold,
```

```

System.Drawing.GraphicsUnit.Point, CType(0, Byte))
    Me.TBVisor.Location = New System.Drawing.Point(12, 166)
    Me.TBVisor.Name = "TBVisor"
    Me.TBVisor.Size = New System.Drawing.Size(400, 83)
    Me.TBVisor.TabIndex = 0
    Me.TBVisor.Text = "AMARILLO"
    Me.TBVisor.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
    '
    'TBTime
    '
    Me.TBTime.BackColor =
System.Drawing.SystemColors.ControlLightLight
    Me.TBTime.Font = New System.Drawing.Font("Microsoft Sans Serif",
48.0!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
    Me.TBTime.Location = New System.Drawing.Point(66, 12)
    Me.TBTime.Name = "TBTime"
    Me.TBTime.ReadOnly = True
    Me.TBTime.Size = New System.Drawing.Size(295, 80)
    Me.TBTime.TabIndex = 4
    Me.TBTime.Text = "00:00"
    Me.TBTime.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
    Me.TBTime.Visible = False
    '
    'Visor
    '
    Me.AutoScaleDimensions = New System.Drawing.SizeF(6.0!, 13.0!)
    Me.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font
    Me.BackColor = System.Drawing.Color.White
    Me.ClientSize = New System.Drawing.Size(424, 348)
    Me.Controls.Add(Me.TBTime)
    Me.Controls.Add(Me.TBVisor)
    Me.Name = "Visor"
    Me.Text = "Form1"
    Me.ResumeLayout(False)
    Me.PerformLayout()

End Sub
Friend WithEvents TBVisor As System.Windows.Forms.TextBox
Friend WithEvents TBTime As System.Windows.Forms.TextBox
End Class

```

### 17.1.3 SettingsDesigner.vb

```

Option Strict On
Option Explicit On

```

Namespace My

```
<Global.System.Runtime.CompilerServices.CompilerGeneratedAttribute(),  
-  
Global.System.CodeDom.Compiler.GeneratedCodeAttribute("Microsoft.VisualStudio.Editors.SettingsDesigner.SettingsSingleFileGenerator", "10.0.0.0"),  
-  
Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.ComponentModel.EditorBrowsableState.Advanced)> _  
    Partial Friend NotInheritable Class MySettings  
        Inherits Global.System.Configuration.ApplicationSettingsBase  
  
        Private Shared defaultInstance As MySettings =  
        CType(Global.System.Configuration.ApplicationSettingsBase.Synchronized(  
        New MySettings), MySettings)  
  
#Region "My.Settings Auto-Save Functionality"  
#If _MyType = "WindowsForms" Then  
    Private Shared addedHandler As Boolean  
  
    Private Shared addedHandlerLockObject As New Object  
  
    <Global.System.Diagnostics.DebuggerNonUserCodeAttribute(),  
    Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.ComponentModel.EditorBrowsableState.Advanced)> _  
        Private Shared Sub AutoSaveSettings(ByVal sender As  
        Global.System.Object, ByVal e As Global.System.EventArgs)  
            If My.Application.SaveMySettingsOnExit Then  
                My.Settings.Save()  
            End If  
        End Sub  
#End If  
#End Region  
  
    Public Shared ReadOnly Property [Default]() As MySettings  
        Get  
  
#If _MyType = "WindowsForms" Then  
        If Not addedHandler Then  
            SyncLock addedHandlerLockObject  
                If Not addedHandler Then  
                    AddHandler My.Application.Shutdown, AddressOf  
AutoSaveSettings
```

```

                addedHandler = True
            End If
        End SyncLock
    End If
#End If
    Return defaultInstance
End Get
End Property
End Class
End Namespace

Namespace My

    <Global.Microsoft.VisualBasic.HideModuleNameAttribute(), _
    Global.System.Diagnostics.DebuggerNonUserCodeAttribute(), _
    Global.System.Runtime.CompilerServices.CompilerGeneratedAttribute(>
    Friend Module MySettingsProperty

    <Global.System.ComponentModel.Design.HelpKeywordAttribute("My.Settings")>
    Friend ReadOnly Property Settings() As
    Global.testcolores.My.MySettings
        Get
            Return Global.testcolores.My.MySettings.Default
        End Get
    End Property
    End Module
End Namespace

```

### 17.1.4 ResourceDesigner.vb

```

Option Strict On
Option Explicit On

```

```

Namespace My.Resources

```

```

    'This class was auto-generated by the StronglyTypedResourceBuilder
    'class via a tool like ResGen or Visual Studio.
    'To add or remove a member, edit your .ResX file then rerun ResGen
    'with the /str option, or rebuild your VS project.
    <summary>
    " A strongly-typed resource class, for looking up localized strings, etc.
    </summary>

```

```

<Global.System.CodeDom.Compiler.GeneratedCodeAttribute("System.Reso

```

```

urces.Tools.StronglyTypedResourceBuilder", "4.0.0.0"), _
    Global.System.Diagnostics.DebuggerNonUserCodeAttribute(), _
    Global.System.Runtime.CompilerServices.CompilerGeneratedAttribute(),
- Global.Microsoft.VisualBasic.HideModuleNameAttribute())> _
Friend Module Resources

    Private resourceMan As Global.System.Resources.ResourceManager

    Private resourceCulture As Global.System.Globalization.CultureInfo

    ""<summary>
    "" Returns the cached ResourceManager instance used by this class.
    ""</summary>

<Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.
ComponentModel.EditorBrowsableState.Advanced)> _
    Friend ReadOnly Property ResourceManager() As
Global.System.Resources.ResourceManager
    Get
        If Object.ReferenceEquals(resourceMan, Nothing) Then
            Dim temp As Global.System.Resources.ResourceManager =
New Global.System.Resources.ResourceManager("testcolores.Resources",
GetType(Resources).Assembly)
            resourceMan = temp
        End If
        Return resourceMan
    End Get
End Property

    ""<summary>
    "" Overrides the current thread's CurrentUICulture property for all
    "" resource lookups using this strongly typed resource class.
    ""</summary>

<Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.
ComponentModel.EditorBrowsableState.Advanced)> _
    Friend Property Culture() As Global.System.Globalization.CultureInfo
    Get
        Return resourceCulture
    End Get
    Set(ByVal value As Global.System.Globalization.CultureInfo)
        resourceCulture = value
    End Set
End Property
End Module
End Namespace

```



## 17.1.5 Control.vb

Public Class Control

```
    Dim colorespal() As String = {"GRIS", "ROJO", "AZUL", "VERDE",  
    "NEGRO", "NARANJA", "AMARILLO"}  
    Dim coloreston() As System.Drawing.Color  
    Dim coloresche() As Boolean = {False, True, True, True, False, False,  
    True}  
    Dim coloresnum As Integer = 4  
    Dim rnd As New Random  
    Dim tonoAnt As System.Drawing.Color  
    Dim palaAnt As String  
    Dim calibrando As Integer
```

```
    Private Sub Control_KeyDown(ByVal sender As Object, ByVal e As  
    System.Windows.Forms.KeyEventArgs) Handles Me.KeyDown
```

```
        Dim bHandled As Boolean = False  
        Select Case e.KeyCode  
            Case Keys.Right  
                BAcierto_Click(sender, e)  
                TextBox1.focus()  
                e.Handled = True  
            Case Keys.Left  
                BFallo_Click(sender, e)  
                TextBox1.focus()  
                e.Handled = True  
            Case Keys.Up  
                BFallo_Click(sender, e)  
                TextBox1.focus()  
                e.Handled = True  
            Case Keys.Down  
                BAcierto_Click(sender, e)  
                TextBox1.focus()  
                e.Handled = True  
            Case Keys.Return  
                BAcierto_Click(sender, e)  
                TextBox1.Focus()  
                e.Handled = True  
            Case Keys.Space  
                BAcierto_Click(sender, e)  
                TextBox1.Focus()  
                e.Handled = True  
            Case Keys.Next  
                BAcierto_Click(sender, e)
```

```

        TextBox1.Focus()
        e.Handled = True
    Case Keys.PageDown
        BAcierto_Click(sender, e)
        TextBox1.Focus()
        e.Handled = True
    Case Keys.PageUp
        BFallo_Click(sender, e)
        TextBox1.Focus()
        e.Handled = True
    Case Else
        BFallo_Click(sender, e)
        TextBox1.Focus()
        e.Handled = True
End Select

```

End Sub

Private Sub Control\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

```

    CBAcierto.SelectedIndex = 0
    CBTiempoPase.SelectedIndex = 99
    CBNegroCalibracion.SelectedIndex = 9

```

```

    TimerPase.Interval = CInt(CBTiempoPase.Text)

```

```

    Visor.Show()
    Visor.SetDesktopLocation(700, 70)

```

```

    coloreston = {CBGris.BackColor, CBRojo.BackColor,
    CBAzul.BackColor, CBVerde.BackColor, CBNegro.BackColor,
    CBNaranja.BackColor, CBAmarrillo.BackColor}

```

```

    tonoAnt = CBNegro.BackColor
    palaAnt = "Amarillo"

```

End Sub

Private Sub CBTiempoPase\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles CBTiempoPase.KeyPress

```

    e.Handled = True

```

End Sub

```
Private Sub CBTiempoPase_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBTiempoPase.SelectedIndexChanged
```

```
    TimerPase.Interval = CInt(CBTiempoPase.Text)
    TextBox1.Focus()
```

End Sub

```
Private Sub BPlayPause_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BPlayPause.Click
```

```
    TimerTime.Enabled = Not (TimerTime.Enabled)
```

```
    If TimerTime.Enabled = True Then
        calibrando = CInt(CBInicioCalibracion.Text)
        presentar()
```

```
    Else
        TimerPase.Enabled = False
    End If
```

```
    TextBox1.Focus()
```

End Sub

```
Private Sub TimerTime_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TimerTime.Tick
```

```
    Dim min As String = TBTime.Text.Substring(0, 2)
    Dim seg As String = TBTime.Text.Substring(3, 2)
```

```
    seg = (CInt(seg) + 1).ToString
```

```
    If seg.Length = 1 Then
        seg = "0" & seg
    End If
```

```
    If CInt(seg) > 59 Then
        seg = "00"
        min = (CInt(min) + 1).ToString
    End If
```

```
If min.Length = 1 Then
    min = "0" & min
End If
```

```
TBTime.Text = min & ":" & seg
```

```
Visor.TBTime.Text = TBTime.Text
```

```
End Sub
```

```
Private Sub CBGris_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CBGris.CheckedChanged,
CBRojo.CheckedChanged, CBAzul.CheckedChanged,
CBVerde.CheckedChanged, CBNegro.CheckedChanged,
CBNaranja.CheckedChanged, CBAmarrillo.CheckedChanged
```

```
If sender.text <> "" Then
```

```
    coloresche(sender.Tag) = sender.Checked
```

```
    If sender.Checked = True Then
        coloresnum = coloresnum + 1
```

```
    Else
        coloresnum = coloresnum - 1
```

```
    End If
```

```
End If
```

```
TextBox1.Focus()
```

```
End Sub
```

```
Private Sub PalabraColorAlea(ByRef TonoUno As System.Drawing.Color,
ByRef PalaUno As String, ByRef TonoDos As System.Drawing.Color,
ByRef PalaDos As String)
```

```
    Dim count As Integer = 0
```

```
    Dim coloritospal(coloresnum - 1) As String
```

```
    Dim coloritoston(coloresnum - 1) As System.Drawing.Color
```

```
    For i As Integer = 0 To 6
```

```
        If coloresche(i) Then
```

```
            coloritospal(count) = colorespal(i)
```

```
            coloritoston(count) = coloreston(i)
```

```
            count = count + 1
```

```

    End If
Next

Randomize()
Dim alea As Integer = rnd.Next(coloresnum)
TonoUno = coloritoston(alea)
PalaUno = coloritospal(alea)

Randomize()
Dim aleo As Integer = rnd.Next(coloresnum)
If CBDesparejar.Checked Then
    Do
        aleo = rnd.Next(coloresnum)
    Loop While alea = aleo
End If
TonoDos = coloritoston(aleo)
PalaDos = coloritospal(aleo)

End Sub

Private Sub presentar()

    If coloresnum > 1 Then

        If CBPasar.Checked = True Then
            TimerPase.Interval = CInt(CBTiempoPase.Text)
            TimerPase.Enabled = True
        End If

        Dim TonoUno As System.Drawing.Color
        Dim TonoDos As System.Drawing.Color
        Dim PalaUno As String = ""
        Dim PalaDos As String = ""

        PalabraColorAlea(TonoUno, PalaUno, TonoDos, PalaDos)

        While PalaUno = palaAnt And TonoDos = tonoAnt
            PalabraColorAlea(TonoUno, PalaUno, TonoDos, PalaDos)
        End While

        palaAnt = PalaUno
        tonoAnt = TonoDos

        If CBCalibracion.Checked = True And calibrando > 0 Then
            calibrando = calibrando - 1
            If calibrando > 0 Then

```

```

        CBIncioCalibracion.SelectedIndex =
CBIncioCalibracion.SelectedIndex - 1
    Else
        CBCalibracion.Checked = False
    End If
    TonoDos = TonoUno
    PalaDos = PalaUno
End If

Visor.TBVisor.Text = PalaUno
Visor.TBVisor.ForeColor = TonoDos

If CBAcierto.SelectedIndex <> 0 Then
    TBCorrecto.Text = PalaUno
    TBCorrecto.ForeColor = TonoUno
Else
    TBCorrecto.Text = PalaDos
    TBCorrecto.ForeColor = TonoDos
End If

If CBSilencio.Checked = False Then
    Beep()
End If

Else
    MsgBox("No puede seleccionar sólo un color")
End If

TextBox1.Focus()

End Sub

Private Sub BReinicio_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BReinicio.Click

    TimerPase.Enabled = False
    TimerTime.Enabled = False
    TBTime.Text = "00:00"
    TBAcierto.Text = "0"
    TBFallo.Text = "0"

    TextBox1.Focus()

End Sub

```

```
Private Sub TimerPase_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TimerPase.Tick
```

```
    TimerPase.Enabled = False  
    If CBPaseFallo.Checked = True Then  
        TBFallo.Text = (CInt(TBFallo.Text) + 1).ToString  
    End If  
    presentar()
```

```
End Sub
```

```
Private Sub BFallo_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BFallo.Click
```

```
    TimerPase.Enabled = False  
    TBFallo.Text = (CInt(TBFallo.Text) + 1).ToString  
    presentar()
```

```
    TextBox1.focus()
```

```
End Sub
```

```
Private Sub BAcierto_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BAcierto.Click
```

```
    TimerPase.Enabled = False  
    TBAcierto.Text = (CInt(TBAcierto.Text) + 1).ToString  
    TextBox1.focus()  
    presentar()
```

```
End Sub
```

```
Private Sub CBVerTiempo_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBVerTiempo.CheckedChanged
```

```
    Visor.TBTime.Visible = CBVerTiempo.Checked
```

```
    TextBox1.focus()
```

```
End Sub
```

```
Private Sub CBCalibracion_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBCalibracion.CheckedChanged
```

```
    If CBCalibracion.Checked = True Then  
        calibrando = CInt(CBInicioCalibracion.Text)  
    Else  
        calibrando = 0  
    End If
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub CBPasar_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBPasar.CheckedChanged
```

```
    TimerPase.Enabled = CBPasar.Checked
```

```
    If TimerTime.Enabled = False Then  
        TimerPase.Enabled = False  
    End If
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub CBInicioCalibracion_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles CBInicioCalibracion.KeyDown
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub CBInicioCalibracion_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles CBInicioCalibracion.KeyPress
```

```
    e.Handled = True
```

```
End Sub
```



```
Private Sub CBIncioCalibracion_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBIncioCalibracion.SelectedIndexChanged
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub CBDesperejar_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBDesperejar.CheckedChanged
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub CBPaseFallo_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBPaseFallo.CheckedChanged
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub CBSilencio_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CBSilencio.CheckedChanged
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub BAcierto_GotFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles BAcierto.GotFocus
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub BAcierto_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles BAcierto.KeyDown
```

```
        'If e.KeyCode = 13 Or e.KeyCode = 39 Or e.KeyCode = 40 Or  
e.KeyCode = 32 Then  
        ' BAcierto_Click(sender, e)  
        'Else  
        ' BFallo_Click(sender, e)  
        'End If
```

End Sub

```
Private Sub CBAcierto_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles CBAcierto.KeyPress  
    e.Handled = True  
End Sub
```

```
Private Sub CBAcierto_SelectedIndexChanged(ByVal sender As  
System.Object, ByVal e As System.EventArgs) Handles  
CBAcierto.SelectedIndexChanged
```

```
    TextBox1.Focus()
```

End Sub

```
Private Sub TBCorrecto_GotFocus(ByVal sender As Object, ByVal e As  
System.EventArgs) Handles TBCorrecto.GotFocus
```

```
    TextBox1.Focus()
```

End Sub

```
Private Sub TBCorrecto_KeyDown(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyEventArgs) Handles TBCorrecto.KeyDown
```

```
        'If e.KeyCode = 13 Or e.KeyCode = 39 Or e.KeyCode = 40 Or  
e.KeyCode = 32 Then  
        ' BAcierto_Click(sender, e)  
        'Else  
        ' BFallo_Click(sender, e)  
        'End If
```

End Sub

```
Private Sub GBColores_Enter(ByVal sender As System.Object, ByVal e
```

As System.EventArgs) Handles GBColores.Enter

    TextBox1.Focus()

End Sub

Private Sub TBTime\_GotFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles TBTime.GotFocus

    TextBox1.Focus()

End Sub

Private Sub TBTime\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TBTime.TextChanged

    TextBox1.Focus()

End Sub

Private Sub TBFallo\_GotFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles TBFallo.GotFocus

    TextBox1.Focus()

End Sub

Private Sub TBFallo\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TBFallo.TextChanged

    TextBox1.Focus()

End Sub

Private Sub TBAcuerdo\_GotFocus(ByVal sender As Object, ByVal e As System.EventArgs) Handles TBAcuerdo.GotFocus

    TextBox1.Focus()

End Sub

```
Private Sub TBAcierto_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles TBAcierto.TextChanged
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub TBCorrecto_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles TBCorrecto.TextChanged
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub BFallo_GotFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BFallo.GotFocus
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
End Class
```

### 17.1.6 ControlDesigner.vb

```
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()> _
```

```
Partial Class Control
```

```
    Inherits System.Windows.Forms.Form
```

```
'Form reemplaza a Dispose para limpiar la lista de componentes.
```

```
<System.Diagnostics.DebuggerNonUserCode()> _
```

```
Protected Overrides Sub Dispose(ByVal disposing As Boolean)
```

```
    Try
```

```
        If disposing AndAlso components IsNot Nothing Then
            components.Dispose()
```

```
        End If
```

```
    Finally
```

```
        MyBase.Dispose(disposing)
```

```
    End Try
```

```
End Sub
```

```
'Requerido por el Diseñador de Windows Forms
```

```
Private components As System.ComponentModel.IContainer
```

```
'NOTA: el Diseñador de Windows Forms necesita el siguiente
```

procedimiento

'Se puede modificar usando el Diseñador de Windows Forms.

'No lo modifique con el editor de código.

```
<System.Diagnostics.DebuggerStepThrough(> _
```

```
Private Sub InitializeComponent()
```

```
    Me.components = New System.ComponentModel.Container()
```

```
    Me.CBAcierto = New System.Windows.Forms.ComboBox()
```

```
    Me.TextBox1 = New System.Windows.Forms.TextBox()
```

```
    Me.GBColores = New System.Windows.Forms.GroupBox()
```

```
    Me.CBNaranja = New System.Windows.Forms.CheckBox()
```

```
    Me.CBGris = New System.Windows.Forms.CheckBox()
```

```
    Me.CBNegro = New System.Windows.Forms.CheckBox()
```

```
    Me.CBAmarillo = New System.Windows.Forms.CheckBox()
```

```
    Me.CBVerde = New System.Windows.Forms.CheckBox()
```

```
    Me.CBAzul = New System.Windows.Forms.CheckBox()
```

```
    Me.CBRojo = New System.Windows.Forms.CheckBox()
```

```
    Me.TBTime = New System.Windows.Forms.TextBox()
```

```
    Me.TextBox3 = New System.Windows.Forms.TextBox()
```

```
    Me.TextBox4 = New System.Windows.Forms.TextBox()
```

```
    Me.BPlayPause = New System.Windows.Forms.Button()
```

```
    Me.BReinicio = New System.Windows.Forms.Button()
```

```
    Me.TBFallo = New System.Windows.Forms.TextBox()
```

```
    Me.TextBox6 = New System.Windows.Forms.TextBox()
```

```
    Me.TextBox7 = New System.Windows.Forms.TextBox()
```

```
    Me.TBAcierto = New System.Windows.Forms.TextBox()
```

```
    Me.BFallo = New System.Windows.Forms.Button()
```

```
    Me.BAcierto = New System.Windows.Forms.Button()
```

```
    Me.GroupBox1 = New System.Windows.Forms.GroupBox()
```

```
    Me.CBTiempoPase = New System.Windows.Forms.ComboBox()
```

```
    Me.CBInicioCalibracion = New System.Windows.Forms.ComboBox()
```

```
    Me.CBDesparejar = New System.Windows.Forms.CheckBox()
```

```
    Me.CBCalibracion = New System.Windows.Forms.CheckBox()
```

```
    Me.CBPasar = New System.Windows.Forms.CheckBox()
```

```
    Me.CBPaseFallo = New System.Windows.Forms.CheckBox()
```

```
    Me.CBSilencio = New System.Windows.Forms.CheckBox()
```

```
    Me.CBVerTiempo = New System.Windows.Forms.CheckBox()
```

```
    Me.TBCorrecto = New System.Windows.Forms.TextBox()
```

```
    Me.TimerTime = New System.Windows.Forms.Timer(Me.components)
```

```
    Me.TimerPase = New System.Windows.Forms.Timer(Me.components)
```

```
    Me.GBColores.SuspendLayout()
```

```
    Me.GroupBox1.SuspendLayout()
```

```
    Me.SuspendLayout()
```

```
,
```

```
    'CBAcierto
```

```
,
```

```
    Me.CBAcierto.FormattingEnabled = True
```

```
    Me.CBAcierto.Items.AddRange(New Object() {"Color de tinta."},
```

```

"Palabra escrita."})
    Me.CBAcierto.Location = New System.Drawing.Point(12, 38)
    Me.CBAcierto.Name = "CBAcierto"
    Me.CBAcierto.Size = New System.Drawing.Size(121, 21)
    Me.CBAcierto.TabIndex = 1
    Me.CBAcierto.TabStop = False
    '
    'TextBox1
    '
    Me.TextBox1.BackColor = System.Drawing.Color.SkyBlue
    Me.TextBox1.Location = New System.Drawing.Point(12, 12)
    Me.TextBox1.Name = "TextBox1"
    Me.TextBox1.ReadOnly = True
    Me.TextBox1.Size = New System.Drawing.Size(121, 20)
    Me.TextBox1.TabIndex = 0
    Me.TextBox1.Text = "Defina un acierto"
    Me.TextBox1.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
    '
    'GBColores
    '
    Me.GBColores.Controls.Add(Me.CBNaranja)
    Me.GBColores.Controls.Add(Me.CBGris)
    Me.GBColores.Controls.Add(Me.CBNegro)
    Me.GBColores.Controls.Add(Me.CBAmarillo)
    Me.GBColores.Controls.Add(Me.CBVerde)
    Me.GBColores.Controls.Add(Me.CBAzul)
    Me.GBColores.Controls.Add(Me.CBRojo)
    Me.GBColores.Location = New System.Drawing.Point(12, 95)
    Me.GBColores.Name = "GBColores"
    Me.GBColores.Size = New System.Drawing.Size(121, 178)
    Me.GBColores.TabIndex = 2
    Me.GBColores.TabStop = False
    '
    'CBNaranja
    '
    Me.CBNaranja.AutoSize = True
    Me.CBNaranja.BackColor =
System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
    CType(CType(128, Byte), Integer), CType(CType(0, Byte), Integer))
    Me.CBNaranja.Location = New System.Drawing.Point(5, 131)
    Me.CBNaranja.Name = "CBNaranja"
    Me.CBNaranja.Size = New System.Drawing.Size(76, 17)
    Me.CBNaranja.TabIndex = 6
    Me.CBNaranja.Tag = "5"
    Me.CBNaranja.Text = "NARANJA"
    Me.CBNaranja.UseVisualStyleBackColor = False

```

```

'
'CBGris
'
Me.CBGris.AutoSize = True
Me.CBGris.BackColor = System.Drawing.Color.DarkGray
Me.CBGris.Location = New System.Drawing.Point(6, 13)
Me.CBGris.Name = "CBGris"
Me.CBGris.Size = New System.Drawing.Size(52, 17)
Me.CBGris.TabIndex = 5
Me.CBGris.Tag = "0"
Me.CBGris.Text = "GRIS"
Me.CBGris.UseVisualStyleBackColor = False
'
'CBNegro
'
Me.CBNegro.AutoSize = True
Me.CBNegro.BackColor = System.Drawing.Color.Black
Me.CBNegro.ForeColor =
System.Drawing.SystemColors.ControlLightLight
Me.CBNegro.Location = New System.Drawing.Point(6, 108)
Me.CBNegro.Name = "CBNegro"
Me.CBNegro.Size = New System.Drawing.Size(65, 17)
Me.CBNegro.TabIndex = 4
Me.CBNegro.Tag = "4"
Me.CBNegro.Text = "NEGRO"
Me.CBNegro.UseVisualStyleBackColor = False
'
'CBAmarillo
'
Me.CBAmarillo.AutoSize = True
Me.CBAmarillo.BackColor = System.Drawing.Color.Yellow
Me.CBAmarillo.Checked = True
Me.CBAmarillo.CheckState =
System.Windows.Forms.CheckState.Checked
Me.CBAmarillo.Location = New System.Drawing.Point(5, 154)
Me.CBAmarillo.Name = "CBAmarillo"
Me.CBAmarillo.Size = New System.Drawing.Size(80, 17)
Me.CBAmarillo.TabIndex = 2
Me.CBAmarillo.Tag = "6"
Me.CBAmarillo.Text = "AMARILLO"
Me.CBAmarillo.UseVisualStyleBackColor = False
'
'CBVerde
'
Me.CBVerde.AutoSize = True
Me.CBVerde.BackColor =
System.Drawing.Color.FromArgb(CType(CType(0, Byte), Integer),

```

```

CType(CType(192, Byte), Integer), CType(CType(0, Byte), Integer))
    Me.CBVerde.Checked = True
    Me.CBVerde.CheckState =
System.Windows.Forms.CheckState.Checked
    Me.CBVerde.Location = New System.Drawing.Point(6, 83)
    Me.CBVerde.Name = "CBVerde"
    Me.CBVerde.Size = New System.Drawing.Size(63, 17)
    Me.CBVerde.TabIndex = 1
    Me.CBVerde.Tag = "3"
    Me.CBVerde.Text = "VERDE"
    Me.CBVerde.UseVisualStyleBackColor = False
    '
    'CBAzul
    '
    Me.CBAzul.AutoSize = True
    Me.CBAzul.BackColor = System.Drawing.Color.SkyBlue
    Me.CBAzul.Checked = True
    Me.CBAzul.CheckState = System.Windows.Forms.CheckState.Checked
    Me.CBAzul.Location = New System.Drawing.Point(6, 60)
    Me.CBAzul.Name = "CBAzul"
    Me.CBAzul.Size = New System.Drawing.Size(54, 17)
    Me.CBAzul.TabIndex = 3
    Me.CBAzul.Tag = "2"
    Me.CBAzul.Text = "AZUL"
    Me.CBAzul.UseVisualStyleBackColor = False
    '
    'CBRojo
    '
    Me.CBRojo.AutoSize = True
    Me.CBRojo.BackColor = System.Drawing.Color.Red
    Me.CBRojo.Checked = True
    Me.CBRojo.CheckState = System.Windows.Forms.CheckState.Checked
    Me.CBRojo.ForeColor =
System.Drawing.SystemColors.ControlLightLight
    Me.CBRojo.Location = New System.Drawing.Point(5, 37)
    Me.CBRojo.Name = "CBRojo"
    Me.CBRojo.Size = New System.Drawing.Size(55, 17)
    Me.CBRojo.TabIndex = 0
    Me.CBRojo.Tag = "1"
    Me.CBRojo.Text = "ROJO"
    Me.CBRojo.UseVisualStyleBackColor = False
    '
    'TBTime
    '
    Me.TBTime.BackColor =
System.Drawing.SystemColors.ControlLightLight
    Me.TBTime.Font = New System.Drawing.Font("Microsoft Sans Serif",

```



```

48.0!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
    Me.TBTime.Location = New System.Drawing.Point(139, 38)
    Me.TBTime.Name = "TBTime"
    Me.TBTime.ReadOnly = True
    Me.TBTime.Size = New System.Drawing.Size(295, 80)
    Me.TBTime.TabIndex = 3
    Me.TBTime.Text = "00:00"
    Me.TBTime.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
    ,
    'TextBox3
    ,
    Me.TextBox3.BackColor = System.Drawing.Color.SkyBlue
    Me.TextBox3.Location = New System.Drawing.Point(139, 12)
    Me.TextBox3.Name = "TextBox3"
    Me.TextBox3.ReadOnly = True
    Me.TextBox3.Size = New System.Drawing.Size(295, 20)
    Me.TextBox3.TabIndex = 4
    Me.TextBox3.Text = "Defina un acierto"
    Me.TextBox3.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
    ,
    'TextBox4
    ,
    Me.TextBox4.BackColor = System.Drawing.Color.SkyBlue
    Me.TextBox4.Location = New System.Drawing.Point(12, 75)
    Me.TextBox4.Name = "TextBox4"
    Me.TextBox4.ReadOnly = True
    Me.TextBox4.Size = New System.Drawing.Size(121, 20)
    Me.TextBox4.TabIndex = 5
    Me.TextBox4.Text = "Colores"
    Me.TextBox4.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
    ,
    'BPlayPause
    ,
    Me.BPlayPause.Location = New System.Drawing.Point(139, 124)
    Me.BPlayPause.Name = "BPlayPause"
    Me.BPlayPause.Size = New System.Drawing.Size(145, 23)
    Me.BPlayPause.TabIndex = 6
    Me.BPlayPause.Text = "Continuar/Parar"
    Me.BPlayPause.UseVisualStyleBackColor = True
    ,
    'BReinicio
    ,
    Me.BReinicio.Location = New System.Drawing.Point(289, 124)

```

```

Me.BReinicio.Name = "BReinicio"
Me.BReinicio.Size = New System.Drawing.Size(145, 23)
Me.BReinicio.TabIndex = 7
Me.BReinicio.Text = "Reiniciar"
Me.BReinicio.UseVisualStyleBackColor = True
'
'TBFallo
'
Me.TBFallo.BackColor =
System.Drawing.SystemColors.ControlLightLight
Me.TBFallo.Font = New System.Drawing.Font("Microsoft Sans Serif",
48.0!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.TBFallo.Location = New System.Drawing.Point(201, 193)
Me.TBFallo.Name = "TBFallo"
Me.TBFallo.ReadOnly = True
Me.TBFallo.Size = New System.Drawing.Size(83, 80)
Me.TBFallo.TabIndex = 8
Me.TBFallo.TabStop = False
Me.TBFallo.Text = "0"
Me.TBFallo.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
'
'TextBox6
'
Me.TextBox6.BackColor = System.Drawing.Color.SkyBlue
Me.TextBox6.Location = New System.Drawing.Point(139, 164)
Me.TextBox6.Name = "TextBox6"
Me.TextBox6.ReadOnly = True
Me.TextBox6.Size = New System.Drawing.Size(145, 20)
Me.TextBox6.TabIndex = 9
Me.TextBox6.Text = "Fallos"
Me.TextBox6.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
'
'TextBox7
'
Me.TextBox7.BackColor = System.Drawing.Color.SkyBlue
Me.TextBox7.Location = New System.Drawing.Point(289, 164)
Me.TextBox7.Name = "TextBox7"
Me.TextBox7.ReadOnly = True
Me.TextBox7.Size = New System.Drawing.Size(145, 20)
Me.TextBox7.TabIndex = 10
Me.TextBox7.Text = "Aciertos"
Me.TextBox7.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
'

```

```

'TBAcierto
'
Me.TBAcierto.BackColor =
System.Drawing.SystemColors.ControlLightLight
Me.TBAcierto.Font = New System.Drawing.Font("Microsoft Sans
Serif", 48.0!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.TBAcierto.Location = New System.Drawing.Point(289, 193)
Me.TBAcierto.Name = "TBAcierto"
Me.TBAcierto.ReadOnly = True
Me.TBAcierto.Size = New System.Drawing.Size(83, 80)
Me.TBAcierto.TabIndex = 11
Me.TBAcierto.TabStop = False
Me.TBAcierto.Text = "0"
Me.TBAcierto.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
'
'BFallo
'
Me.BFallo.Location = New System.Drawing.Point(139, 193)
Me.BFallo.Name = "BFallo"
Me.BFallo.Size = New System.Drawing.Size(56, 80)
Me.BFallo.TabIndex = 12
Me.BFallo.TabStop = False
Me.BFallo.Text = "Fallos"
Me.BFallo.UseVisualStyleBackColor = True
'
'BAcierto
'
Me.BAcierto.Location = New System.Drawing.Point(378, 193)
Me.BAcierto.Name = "BAcierto"
Me.BAcierto.Size = New System.Drawing.Size(56, 80)
Me.BAcierto.TabIndex = 13
Me.BAcierto.TabStop = False
Me.BAcierto.Text = "Acieros"
Me.BAcierto.UseVisualStyleBackColor = True
'
'GroupBox1
'
Me.GroupBox1.Controls.Add(Me.CBTiempoPase)
Me.GroupBox1.Controls.Add(Me.CBInicioCalibracion)
Me.GroupBox1.Controls.Add(Me.CBDesparejar)
Me.GroupBox1.Controls.Add(Me.CBCalibracion)
Me.GroupBox1.Controls.Add(Me.CBPasar)
Me.GroupBox1.Controls.Add(Me.CBPaseFallo)
Me.GroupBox1.Controls.Add(Me.CBSilencio)
Me.GroupBox1.Controls.Add(Me.CBVerTiempo)

```

```

Me.GroupBox1.Location = New System.Drawing.Point(12, 371)
Me.GroupBox1.Name = "GroupBox1"
Me.GroupBox1.Size = New System.Drawing.Size(422, 51)
Me.GroupBox1.TabIndex = 14
Me.GroupBox1.TabStop = False
'
'CBTiempoPase
'
Me.CBTiempoPase.FormattingEnabled = True
Me.CBTiempoPase.Items.AddRange(New Object() {"100", "200",
"300", "400", "500", "600", "700", "800", "900", "1000", "1100", "1200",
"1300", "1400", "1500", "1600", "1700", "1800", "1900", "2000", "2100",
"2200", "2300", "2400", "2500", "2600", "2700", "2800", "2900", "3000",
"3100", "3200", "3300", "3400", "3500", "3600", "3700", "3800", "3900",
"4000", "4100", "4200", "4300", "4400", "4500", "4600", "4700", "4800",
"4900", "5000", "5100", "5200", "5300", "5400", "5500", "5600", "5700",
"5800", "5900", "6000", "6100", "6200", "6300", "6400", "6500", "6600",
"6700", "6800", "6900", "7000", "7100", "7200", "7300", "7400", "7500",
"7600", "7700", "7800", "7900", "8000", "8100", "8200", "8300", "8400",
"8500", "8600", "8700", "8800", "8900", "9000", "9100", "9200", "9300",
"9400", "9500", "9600", "9700", "9800", "9900", "10000"})
Me.CBTiempoPase.Location = New System.Drawing.Point(350, 30)
Me.CBTiempoPase.Name = "CBTiempoPase"
Me.CBTiempoPase.Size = New System.Drawing.Size(66, 21)
Me.CBTiempoPase.TabIndex = 7
'
'CBInicioCalibracion
'
Me.CBInicioCalibracion.FormattingEnabled = True
Me.CBInicioCalibracion.Items.AddRange(New Object() {"1", "2", "3",
"4", "5", "6", "7", "8", "9", "10"})
Me.CBInicioCalibracion.Location = New System.Drawing.Point(350, 9)
Me.CBInicioCalibracion.Name = "CBInicioCalibracion"
Me.CBInicioCalibracion.Size = New System.Drawing.Size(66, 21)
Me.CBInicioCalibracion.TabIndex = 6
'
'CBDesparejar
'
Me.CBDesparejar.AutoSize = True
Me.CBDesparejar.Checked = True
Me.CBDesparejar.CheckState =
System.Windows.Forms.CheckState.Checked
Me.CBDesparejar.Location = New System.Drawing.Point(102, 12)
Me.CBDesparejar.Name = "CBDesparejar"
Me.CBDesparejar.Size = New System.Drawing.Size(98, 17)
Me.CBDesparejar.TabIndex = 5
Me.CBDesparejar.Text = "Palabra <>tono"

```

```

Me.CBDesparejar.UseVisualStyleBackColor = True
'
'CBCalibracion
'
Me.CBCalibracion.AutoSize = True
Me.CBCalibracion.Location = New System.Drawing.Point(229, 12)
Me.CBCalibracion.Name = "CBCalibracion"
Me.CBCalibracion.Size = New System.Drawing.Size(120, 17)
Me.CBCalibracion.TabIndex = 4
Me.CBCalibracion.Text = "Inicio de calibración"
Me.CBCalibracion.UseVisualStyleBackColor = True
'
'CBPasar
'
Me.CBPasar.AutoSize = True
Me.CBPasar.Location = New System.Drawing.Point(229, 30)
Me.CBPasar.Name = "CBPasar"
Me.CBPasar.Size = New System.Drawing.Size(102, 17)
Me.CBPasar.TabIndex = 3
Me.CBPasar.Text = "Tiempo de pase"
Me.CBPasar.UseVisualStyleBackColor = True
'
'CBPaseFallo
'
Me.CBPaseFallo.AutoSize = True
Me.CBPaseFallo.Location = New System.Drawing.Point(102, 30)
Me.CBPaseFallo.Name = "CBPaseFallo"
Me.CBPaseFallo.Size = New System.Drawing.Size(107, 17)
Me.CBPaseFallo.TabIndex = 2
Me.CBPaseFallo.Text = "Contar pase/fallo"
Me.CBPaseFallo.UseVisualStyleBackColor = True
'
'CBSilencio
'
Me.CBSilencio.AutoSize = True
Me.CBSilencio.Location = New System.Drawing.Point(7, 31)
Me.CBSilencio.Name = "CBSilencio"
Me.CBSilencio.Size = New System.Drawing.Size(63, 17)
Me.CBSilencio.TabIndex = 1
Me.CBSilencio.Text = "Silencio"
Me.CBSilencio.UseVisualStyleBackColor = True
'
'CBVerTiempo
'
Me.CBVerTiempo.AutoSize = True
Me.CBVerTiempo.Location = New System.Drawing.Point(7, 12)
Me.CBVerTiempo.Name = "CBVerTiempo"

```

```

Me.CBVerTiempo.Size = New System.Drawing.Size(76, 17)
Me.CBVerTiempo.TabIndex = 0
Me.CBVerTiempo.Text = "Ver tiempo"
Me.CBVerTiempo.UseVisualStyleBackColor = True
'
'TBCorrecto
'
Me.TBCorrecto.BackColor =
System.Drawing.SystemColors.ControlLightLight
Me.TBCorrecto.Font = New System.Drawing.Font("Microsoft Sans
Serif", 50.25!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.TBCorrecto.Location = New System.Drawing.Point(12, 284)
Me.TBCorrecto.Name = "TBCorrecto"
Me.TBCorrecto.ReadOnly = True
Me.TBCorrecto.Size = New System.Drawing.Size(422, 83)
Me.TBCorrecto.TabIndex = 15
Me.TBCorrecto.TabStop = False
Me.TBCorrecto.TextAlign =
System.Windows.Forms.HorizontalAlignment.Center
'
'TimerTime
'
Me.TimerTime.Interval = 1000
'
'TimerPase
'
'Control
'
Me.AutoScaleDimensions = New System.Drawing.SizeF(6.0!, 13.0!)
Me.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font
Me.ClientSize = New System.Drawing.Size(448, 434)
Me.Controls.Add(Me.TBCorrecto)
Me.Controls.Add(Me.GroupBox1)
Me.Controls.Add(Me.BAcierto)
Me.Controls.Add(Me.BFallo)
Me.Controls.Add(Me.TBAcierto)
Me.Controls.Add(Me.TextBox7)
Me.Controls.Add(Me.TextBox6)
Me.Controls.Add(Me.TBFallo)
Me.Controls.Add(Me.BReinicio)
Me.Controls.Add(Me.BPlayPause)
Me.Controls.Add(Me.TextBox4)
Me.Controls.Add(Me.TextBox3)
Me.Controls.Add(Me.TBTime)
Me.Controls.Add(Me.CBAcierto)

```

```
Me.Controls.Add(Me.GBColores)
Me.Controls.Add(Me.TextBox1)
Me.KeyPreview = True
Me.Name = "Control"
Me.Text = "Ventana de control"
Me.GBColores.ResumeLayout(False)
Me.GBColores.PerformLayout()
Me.GroupBox1.ResumeLayout(False)
Me.GroupBox1.PerformLayout()
Me.ResumeLayout(False)
Me.PerformLayout()
```

End Sub

```
Friend WithEvents CBAcierto As System.Windows.Forms.ComboBox
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
Friend WithEvents GBColores As System.Windows.Forms.GroupBox
Friend WithEvents CBNaranja As System.Windows.Forms.CheckBox
Friend WithEvents CBGris As System.Windows.Forms.CheckBox
Friend WithEvents CBNegro As System.Windows.Forms.CheckBox
Friend WithEvents CBAmarillo As System.Windows.Forms.CheckBox
Friend WithEvents CBVerde As System.Windows.Forms.CheckBox
Friend WithEvents CBAzul As System.Windows.Forms.CheckBox
Friend WithEvents CBRojo As System.Windows.Forms.CheckBox
Friend WithEvents TBTime As System.Windows.Forms.TextBox
Friend WithEvents TextBox3 As System.Windows.Forms.TextBox
Friend WithEvents TextBox4 As System.Windows.Forms.TextBox
Friend WithEvents BPlayPause As System.Windows.Forms.Button
Friend WithEvents BReinicio As System.Windows.Forms.Button
Friend WithEvents TBFallo As System.Windows.Forms.TextBox
Friend WithEvents TextBox6 As System.Windows.Forms.TextBox
Friend WithEvents TextBox7 As System.Windows.Forms.TextBox
Friend WithEvents TBAcierto As System.Windows.Forms.TextBox
Friend WithEvents BFallo As System.Windows.Forms.Button
Friend WithEvents BAcierto As System.Windows.Forms.Button
Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
Friend WithEvents CBPasar As System.Windows.Forms.CheckBox
Friend WithEvents CBPaseFallo As System.Windows.Forms.CheckBox
Friend WithEvents CBSilencio As System.Windows.Forms.CheckBox
Friend WithEvents CBVerTiempo As System.Windows.Forms.CheckBox
Friend WithEvents CBTiempoPase As
System.Windows.Forms.ComboBox
Friend WithEvents CBIncioCalibracion As
System.Windows.Forms.ComboBox
Friend WithEvents CBDesperejar As System.Windows.Forms.CheckBox
Friend WithEvents CBCalibracion As System.Windows.Forms.CheckBox
Friend WithEvents TBCorrecto As System.Windows.Forms.TextBox
Friend WithEvents TimerTime As System.Windows.Forms.Timer
```

```
Friend WithEvents TimerPase As System.Windows.Forms.Timer
```

```
End Class
```

### 17.1.7 AssemblyInfo.vb

```
<Assembly: ComVisible(False)>
```

```
'El siguiente GUID sirve como identificador de typelib si este proyecto se  
expone a COM
```

```
<Assembly: Guid("f854c77c-fd9f-4c30-89de-3e235aa13e57")>
```

```
' La información de versión de un ensamblado consta de los cuatro valores  
siguientes:
```

```
'  
'    Versión principal  
'    Versión secundaria  
'    Número de compilación  
'    Revisión  
,
```

```
' Puede especificar todos los valores o usar los valores predeterminados de  
número de compilación y de revisión
```

```
' mediante el asterisco (*), como se muestra a continuación:
```

```
' <Assembly: AssemblyVersion("1.0.*")>
```

```
<Assembly: AssemblyVersion("1.0.0.0")>
```

```
<Assembly: AssemblyFileVersion("1.0.0.0")>
```

### 17.1.8 Application\_Designer.vb

```
Option Strict On
```

```
Option Explicit On
```

```
Namespace My
```

```
'NOTE: This file is auto-generated; do not modify it directly. To make  
changes,
```

```
' or if you encounter build errors in this file, go to the Project Designer  
' (go to Project Properties or double-click the My Project node in  
' Solution Explorer), and make changes on the Application tab.  
,
```

```
Partial Friend Class MyApplication
```

```
    <Global.System.Diagnostics.DebuggerStepThroughAttribute()> _  
    Public Sub New()
```

```
MyBase.New(Global.Microsoft.VisualBasic.ApplicationServices.Authenticat
```



```
ionMode.Windows)
    Me.IsSingleInstance = false
    Me.EnableVisualStyles = true
    Me.SaveMySettingsOnExit = true
    Me.ShutDownStyle =
Global.Microsoft.VisualBasic.ApplicationServices.ShutdownMode.AfterMainFormCloses
End Sub

<Global.System.Diagnostics.DebuggerStepThroughAttribute()> _
Protected Overrides Sub OnCreateMainForm()
    Me.MainForm = Global.testcolores.Control
End Sub
End Class
End Namespace
```

