

## TEST DE CLASIFICACIÓN DE TARJETAS DE WISCONSIN USANDO INTERFAZ NATURAL DE USUARIO

Montúfar, Esteban<sup>1</sup>, Cabascango, Jenny<sup>2</sup> y Sánchez, Luis<sup>3</sup>.

{emontufar, jcabascango, lusanchezsi}@uide.edu.ec

ORCID: <https://orcid.org/0000-0002-1798-6641><sup>1</sup>, ORCID: <https://orcid.org/0000-0002-3461-9296><sup>2</sup>

ORCID: <https://orcid.org/0000-0002-8503-3471><sup>3</sup>

Escuela de Ingeniería Mecatrónica, Universidad Internacional del Ecuador-UIDE

Recibido (26/07/19), Aceptado (17/08/19)

---

**Resumen:** Este trabajo presenta una nueva aplicación desarrollada para resolver el problema de calificación del test de clasificación de tarjetas de Wisconsin, que es una herramienta utilizada por los psicólogos para ayudar a determinar trastornos en el lóbulo frontal. Se realiza una revisión de la literatura sobre tratamientos y evaluaciones de enfermedades con desarrollo anormal de las funciones ejecutivas que se producen tanto en la infancia como en la vida adulta. La tecnología ha tomado la neuropsicología como su campo de acción y durante los últimos diez años, varios estudios han demostrado la efectividad de su aplicación en el tratamiento y evaluación de enfermedades relacionadas. Se analizan los software que se encuentran en SourceForge y Github, así como aplicaciones que se encuentran en la Internet. La investigación está basada en el uso de la interfaz natural de usuario para el reconocimiento de gestos con las manos y desarrollada en Unity con C# para la adquisición de datos del sensor Kinect V2, finalmente los resultados muestran que este método es eficiente y efectivo para la calificación del test.

---

**Palabras Claves:** Test clasificación tarjetas de Wisconsin, Interfaz natural de usuario, Kinect V2, Reconocimiento gestos de la mano.

## WISCONSIN CARD SORTING TEST BY USING NATURAL USER INTERFACE

---

**Abstract:** This document presents a new application developed to solve the qualification problem of the Wisconsin Card Classification Test, which is a tool used by psychologists to help determine disorders in the frontal lobe. A review of the literature on treatments and evaluations of diseases with abnormal development of executive functions that occur both in childhood and in adult life was performed. Technology has taken neuropsychology as its field of action and during the last ten years, several studies have demonstrated the effectiveness of its application in the treatment and evaluation of related diseases. The software found in SourceForge and Github, as well as the applications found on the Internet, are analyzed. The research is based on the use of the natural user interface for the recognition of hand gestures and developed in Unity with C # for the acquisition of data from the Kinect V2 sensor, finally the results show that this method is efficient and effective for the test score

---

**Keywords:** Wisconsin Card Sorting Test, Natural user interface, Kinect V2, Hand gesture recognition.

## I. INTRODUCCIÓN

El experimento del Test de Clasificación de Tarjetas de Wisconsin (WCST) se desarrolló originalmente para evaluar el razonamiento abstracto y la capacidad de adaptarse a las modificaciones en el entorno, así como una forma de determinar daños en el lóbulo frontal [1] - [3]. Al evaluar las poblaciones de ancianos, también se observó que esta prueba resultó ser más compleja y estresante, por lo que también se usó para determinar un deterioro en las funciones ejecutivas en una edad adulta [4]. El WCST ha sufrido varias modificaciones, tanto en lo teórico como en lo metodológico, para dar resultados confiables [5].

La detección de un deterioro de las funciones ejecutivas actualmente no posee una herramienta que integre entornos virtuales desde su inicio de aplicación de tarjetas físicas para su evaluación, lo cual toma un tiempo considerable según tarda el usuario en ordenar las tarjetas y el profesional que lo aplica hasta obtener el resultado final para emitir un criterio adecuado.

Se encuentra el software CF Puzzle en el sitio web SourceForge, que consiste en un rompecabezas de flexibilidad cognitiva para medir la capacidad de adaptarse a las nuevas reglas. Según [6] esta aplicación fue validada previamente.

De los resultados encontrados en el sitio web de GitHub [7] se analiza a "Wisconsin Card Sort using the Working Memory toolkit"[8] que tiene el código para la simulación de clasificación de las tarjetas de Wisconsin, el código fuente está elaborado en C ++. El "Wisconsin Card Sorting Test Perseveration Scoring Test" es un script para la calificación WCST propuesto por [9]. Finalmente, "CogBeacon: Multimodal dataset for cognitive fatigue"[10], que es una recopilación de datos de usuarios que realizan diferentes versiones de WCST.

Desde la aparición de las computadoras, la forma en que los usuarios se comunican ha evolucionado desde la interfaz de línea de comandos (CLI) pasando por la interfaz gráfica de usuario (GUI) para llegar en la actualidad a la interfaz de natural de usuario (NUI) que se asocia con el reconocimiento de voz, movimientos corporales o partes de él; A partir de esto, se intenta demostrar la versatilidad de esta tecnología en el campo de la psicología con la implementación del test de clasificación de tarjetas de Wisconsin usando interfaz natural de usuario, reduciendo la calificación a milisegundos en contraste con los 15 minutos promedio que se tarda en calificar el test de la forma tradicional.

## II. DESARROLLO

Se continúa con la descripción del problema, las definiciones de las variables, símbolos y fórmulas aplica-

das en el desarrollo de la investigación.

### A. Test de clasificación de Tarjetas de Wisconsin – WCST

El procedimiento descrito por [11] consiste en colocar las 4 tarjetas de estímulo (ver Fig. 1), luego el evaluador coloca el primer conjunto de 64 tarjetas de respuesta que se componen de las posibles combinaciones de las 4 formas, 4 colores y 4 números; el evaluador da las indicaciones para realizar la prueba explicando que la tarea es clasificar cada tarjeta de respuesta debajo de la tarjeta de estímulo que crea apropiada y el sujeto recibirá una retroalimentación que será la palabra "correcto" o "incorrecto" con la que se debe determinar cuál es el criterio de clasificación. Los criterios de clasificación comienzan con el color, seguido de la forma y número. Después de 10 clasificaciones correctas consecutivas, el evaluador cambia el criterio sin avisar al sujeto, el proceso se repite hasta completar 6 categorías o al terminar con los dos conjuntos de 64 tarjetas de respuesta [12].



Figura 1. Tarjetas de estímulos.

El problema que surge al aplicar el test es que el evaluador debe tener en cuenta las respuestas correctas, los errores, las respuestas ambiguas, los errores perseverativos y los errores no perseverativos. En [13], se aclaran las reglas de puntuación cuando una respuesta se ajusta al "principio de perseverancia". El proceso de entrega de los resultados toma en promedio 15 minutos.

### B. Interfaz Natural de Usuario - NUI

Con el uso creciente de dispositivos electrónicos, la Interacción Humano Computador (HCI) [14] está encontrando nuevas formas de facilitar su uso; que ha pasado de los dispositivos mecánicos como el teclado y el ratón a la NUI [15] que está presente en productos que permiten interactuar con el sistema a través de movimientos del cuerpo o la voz, siendo el sensor Kinect V2 uno de ellos. La NUI es un área que aún necesita estudio para demostrar su potencial de acuerdo con Fang [16].

### C. Matrices de transformación de coordenadas

Para lograr un movimiento realista dentro del ambiente virtual se trabaja con las matrices de transformación que permiten el movimiento tridimensional de los objetos y se aplica la propiedad de física para que éstos

se comporten de acuerdo con las leyes de la física aplicando (1) para la traslación, (2), (3), y (4) para hacer la rotación al colisionar con un cuerpo rígido.

$$\begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{1}$$

Donde:

$t_x$  traslación en X;  
 $t_y$  traslación en Y;  
 $t_z$  traslación en Z.

$$R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & 0 \\ 0 & \sin\theta & 1 \end{pmatrix} \tag{2}$$

Donde:

$R_x$  rotación en X;  
 $\theta$  ángulo de rotación.

$$R_y = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \tag{3}$$

Donde:

$R_y$  rotación en Y;  
 $\theta$  ángulo de rotación.

$$R_z = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4}$$

Donde:

$R_z$  rotación en Z;  
 $\theta$  ángulo de rotación.

Luego, en la metodología se describe las ideas para resolver el problema, los elementos desarrollados y el algoritmo propuesto.

### III.METODOLOGÍA

Comienza con el desarrollo del concepto donde se definen los aspectos técnicos del sistema para avanzar hacia el desarrollo iterativo propuesto por la metodología Scrum [17] (ver Fig. 2) hasta que todos los requisitos se cumplan, una vez cerrado el proyecto, se elaboran las pruebas y los resultados del sistema. Para la NUI, se usa la Kinect V2 que soporta el reconocimiento de gestos con las manos de acuerdo a las pruebas efectuadas por Dong [18]; se utiliza a la mano como un cursor que

permite arrastrar y soltar objetos con detección de la mano cerrada y abierta respectivamente.

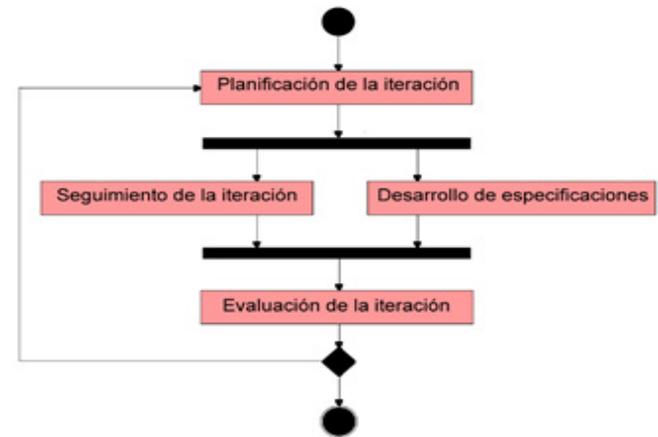


Figura 2. Diagrama de actividades.

#### A.Entorno de Trabajo - Unity

En el entorno de Unity [19] se crea un ambiente donde se muestran las cuatro tarjetas de estímulos y en la parte inferior los agujeros correspondientes para clasificar la tarjeta, el desarrollo se lo hace con C#. El diseño se realiza siguiendo criterios de usabilidad para hacer más efectiva la interacción con el usuario descritos por Donschewa [20].

#### B.Detección de gestos - Kinect V2

De las características de la Kinect V2 analizadas por Wiedemann [21] se tuvieron en cuenta la cámara de profundidad de 512 x 424 x 16 bits por píxel y la capacidad de rastrear 25 articulaciones (ver Fig. 3) y los ~ 50 ms de latencia que dan un buen rendimiento para el sistema desarrollado. Las pruebas desarrolladas por Canal [22] muestran que el sensor es confiable para la detección de gestos como una herramienta de interacción natural con las computadoras.

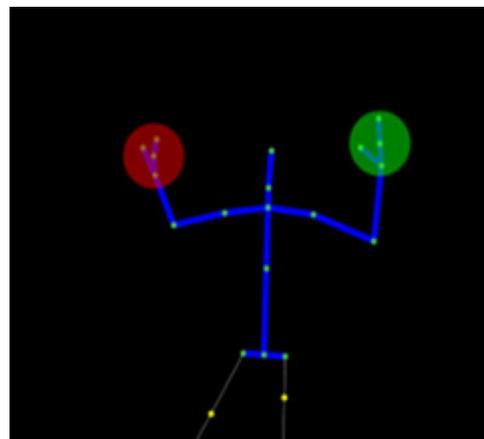


Figura 3. Gestos básicos detectados con una Kinect

– WPF.

### C. Controlador Kinect

El controlador Kinect consta de los scripts: Administrador Kinect, Administrador de interacción y el script arrastrar/soltar; estos scripts son responsables de la administración de los datos de la Kinect y del procesamiento para interactuar con los objetos en la pantalla [23].

- Administrador Kinect: Se encarga del control y adquisición de los datos de la Kinect.

- Administrador de Interacción: Realiza el reconocimiento de los gestos de la mano.

- Script arrastrar/soltar: Controla la interacción entre un objeto arrastrable y la mano aplicando las matrices de transformación de posición disponibles en la clase Vector3 para la traslación y Quaternion para la rotación.

#### D. Script de administración persistente

Mantiene los valores de las variables globales que el sistema administra.

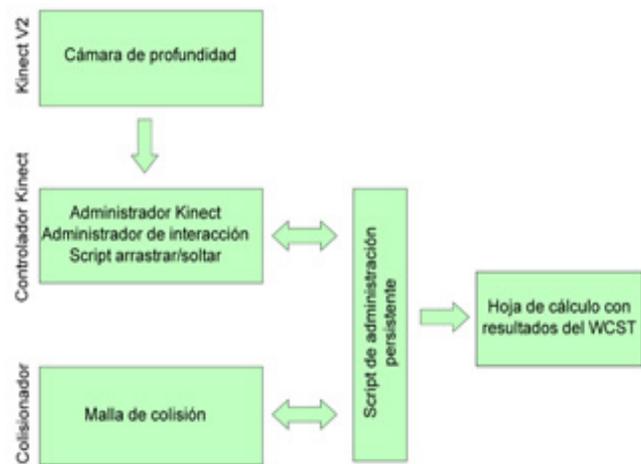
### E. Clasificación de respuestas - Colisionador

Se define al colisionador como una malla que se encuentra en el fondo de los agujeros de clasificación para detectar cuando existe una colisión con un objeto arrastrable.

Se adquieren la coordenada en X, color, forma, y número de la tarjeta. Estos parámetros pasan a los métodos probar color, probar forma o probar número respectivamente para hacer la contabilización de la respuesta en los vectores color, forma, número y otros según corresponde.

### F. Algoritmo desarrollado

La cámara de profundidad envía los datos al controlador Kinect para hacer el reconocimiento de la mano e iniciar la interacción con los objetos arrastrables, paralelamente el colisionador está a la espera para detectar una colisión y enviar los datos al script de administración persistente, una vez finalizado el test este envía el reporte de resultados (Fig. 4).



**Figura 4. Arquitectura de funcionamiento del sistema.**

## IV. RESULTADOS

Para garantizar el correcto funcionamiento del sistema, se desarrollaron pruebas unitarias, pruebas de integración y se analiza la convergencia.

### A. Pruebas unitarias

Consiste en verificar el código de los métodos desarrollados con la creación de los casos de prueba aplicados por Xie [24].

Para la prueba del sensor Kinect, se ejecuta el verificador de configuración Kinect, que analiza las posibles incompatibilidades con el hardware donde se obtuvo una advertencia en el controlador del puerto USB 3.0, lo que indica que no se pudo determinar el ancho de banda de este.

Para evaluar el funcionamiento de los principales métodos utilizados por el sistema, se envían los parámetros a cada método teniendo en cuenta los valores mínimos y máximos, y el método debe devolver el valor esperado.

En el caso del script de administración persistente, los métodos de prueba envían valores determinados para que calcule las dimensiones del WCST y verificar que sean los valores esperados. La tabla I muestra el resumen de los resultados de las pruebas unitarias.

**Tabla I. Pruebas unitarias**

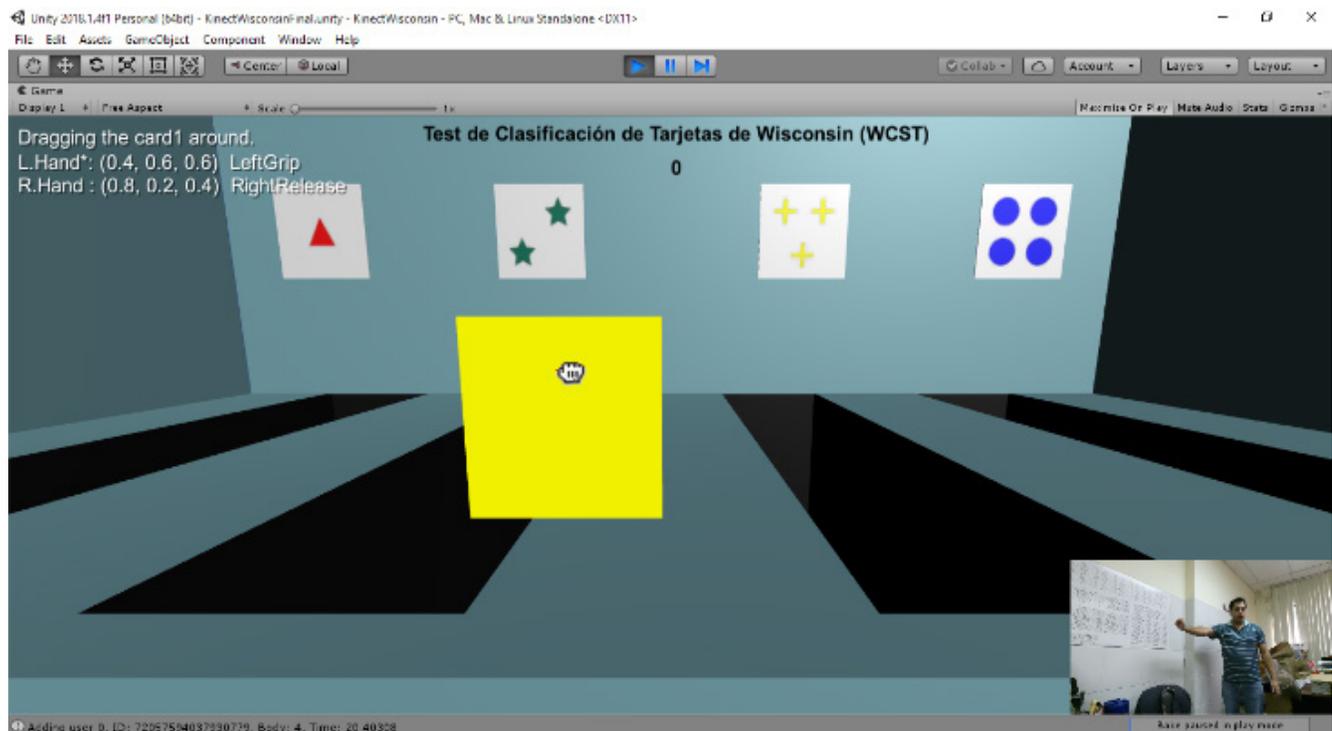
Acción	Resultado	Advertencias	Tiempo
Verificador de configuración Kinect	7/8	Controlador USB (ancho de banda)	<1000ms
Controlador Kinect	4/4	Ninguna	<1ms
Colisionador	8/8	Ninguna	<1ms
Script de administración persistente	10/10	Ninguna	<1ms

Los resultados del WCST se exportan a una hoja de cálculo en la que se procesan los datos recibidos para obtener un modelo de registro en el que el profesional de psicología debe completar los espacios correspon-

dientes a los datos del paciente y dar su diagnóstico con los resultados obtenidos del sistema.

### B.Pruebas de integración

Se comprueba el funcionamiento general de todos los métodos para los cuales la aplicación WCST se lleva a cabo con 10 usuarios cuyas edades van de los 30 a los 50 años distribuidos uniformemente; deben completar la prueba una vez, paralelamente el evaluador toma apuntes en la hoja de respuestas. Los resultados fueron satisfactorios en términos de la prueba, dos usuarios evaluadores necesitaron indicaciones para que el sensor los detecte y, por lo tanto completaran satisfactoriamente la prueba. En la Fig. 5 se muestra la interfaz del sistema en ejecución

**Figura 5. WCST en ejecución.**

### B.Convergencia

Una vez terminado el test se cotejan los resultados emitidos por el sistema con la hoja de respuestas del profesional de psicología donde se observa que hay un 100% de similitud de los resultados comprobando así que el código funciona adecuadamente.

### C.Discusión

CF Puzzle es una aplicación diseñada para medir la flexibilidad cognitiva, está basada en la creación de senders y la tarea de clasificación de tarjetas de Wiscon-

sin permitiendo a los investigadores crear y administrar múltiples experimentos, cuenta con interfaz en 2D y la HCI es usando el teclado y ratón.

El script “Wisconsin Card Sort using the Working Memory toolkit” desarrollado para simular el ordenamiento de las tarjetas de Wisconsin está desarrollado en C++, no posee interfaz gráfica y se puede usar o modificar el código bajo la licencia GNU.

El script “Wisconsin Card Sorting Test Perseveration Scoring Test” para hacer la calificación del WCST el que recibe un archivo en formato CSV para hacer los

cálculos, los resultados son gran cantidad de archivos de análisis individual y un archivo de resumen.

En “Cog – Beacon \_ Multimodal \_ dataset \_ for \_ cognitive \_ fatigue” se encontraron set de datos de la aplicación de las variaciones del WCST son 76 sesiones recopiladas de 19 usuarios masculinos y donde se analiza la fatiga de los usuarios.

Haciendo una comparación con el sistema desarrollado se observa que el uso de la NUI es nuevo en el WCST, posee las características para guardar los resultados en una hoja de cálculo, permite que el evaluador agregue los datos del sujeto y comentarios adicionales que puede ser usado para su posterior análisis. Con la arquitectura planteada El sistema es escalable lo que hace posible integrar un dispositivo de realidad virtual (VR) y convertirlo en un sistema de realidad virtual con lo que se podría hacer una comparación de los resultados con ésta nueva característica; además es posible aplicar ésta tecnología en otros test usados en psicología. Al hacer el test el evaluador puede prestar atención a las actitudes y gestos que presenta el sujeto a la NUI.

Como limitaciones se encuentran que el sistema no es multiplataforma y es necesario contar con el Sensor Kinect V2 para su funcionamiento. Para la aplicación del test es necesario hacer un entrenamiento con el usuario para que entienda el funcionamiento de la NUI.

## V.CONCLUSIONES

La NUI en este sistema proporciona una experiencia inmersiva en el entorno virtual del WCST, basado en esto se demuestra que esta tecnología es aplicable para ser utilizada como uno de los 3 pilares del WCST en la realidad virtual. Se corrobora la utilidad de la cámara de profundidad de la Kinect V2, siendo un dispositivo de bajo costo para el reconocimiento de gestos con las manos y, por lo tanto, abre la posibilidad de profundizar en las nuevas tecnologías disponibles para el NUI.

Las pruebas unitarias aplicadas mostraron que el desarrollo bajo la metodología SCRUM reduce los errores en el tiempo de ejecución así como el tiempo de codificación para obtener software de calidad. La plataforma de desarrollo de Unity resultó ser versátil para la implementación del WCST, ofreciendo herramientas para el desarrollo de gráficos de computadora y su compatibilidad con tecnologías de terceros.

La arquitectura propuesta para la gestión de datos fue fundamental para obtener los resultados del WCST, reduciendo el tiempo a milisegundos en contraste con los 15 minutos en promedio que toma la calificación de la prueba de la manera tradicional. El sistema demostró ser confiable al presentar los resultados del WCST, lo que le permite al evaluador centrar su atención en otros

aspectos del paciente, como su actitud y las reacciones al uso de esta tecnología.

## VI.REFERENCIAS

- [1]A. Robinson, R. Heaton, R. Lehman and D. Stilson. “The utility of the wisconsin card sorting test in detecting and localizing frontal lobe lesions,” *J Consult Clin Psychol*;48(5):605-14, 1980.
- [2]A. Borkowska, W. Drożdż, P. Jurkowski and J. Rybakowski. “The Wisconsin Card Sorting Test and the N-back test in mild cognitive impairment and elderly depression,” *The World Journal of Biological Psychiatry*, 10(4-3), 870-876, 2009.
- [3]S. Anderson, H. Damasio, R. Jones and D. Tranel, “Wisconsin card sorting test performance as a measure of frontal lobe damage,” *J Clin Exp Neuropsychol*, 13(6):909-22, 1991.
- [4]E. Koechlin and C. Summerfield, “An information theoretical approach to prefrontal executive function,” *Trends in Cognitive Sciences*, 11(6), 229-235, 2007.
- [5]E. Nyhus and F. Barceló, “The Wisconsin Card Sorting Test and the cognitive assessment of prefrontal executive functions: A critical update,” *Brain and Cognition*, 71(3), 437-451, 2009.
- [6]Cgonza12. (2013, May 04). CF Puzzle [Online]. Available: <https://sourceforge.net/projects/cfpuzzle/>
- [7]GitHub (2019, July 29). Build software better, together. [Online]. Available: <https://github.com/search?q=wisconsin+card+sort>
- [8]G. Dubois. (2016, Dec 05). A C++ program that performs the cognitive task known as the Wisconsin Card Sort using the Working Memory toolkit [Online] Available:<https://github.com/G-Dubois/Wisconsin-Card-Sort-using-WMtk/blob/master/wcst.cpp>