

BrainCap

V1.0.0.0

***Product Scope***

23.02.21

Huber Killian

Lecoultre Hugo

# Table des matières

[Table des matières 2](#_gjdgxs)

[Introduction 3](#_30j0zll)

[Epreuves 4](#_1fob9te)

[Aim trainer 4](#_3znysh7)

[Reaction time 4](#_2et92p0)

[Sequential memory test 4](#_3dy6vkm)

[Short-term memory / letters 4](#_1t3h5sf)

[Short-term memory / numbers 4](#_4d34og8)

[Color Brainer : 5](#_2s8eyo1)

[Click counter 5](#_17dp8vu)

[Flick trainer 5](#_3rdcrjn)

[Chimp test 5](#_26in1rg)

# Introduction

|  |
| --- |
| **En quoi ça consiste ?** |

BrainCap regroupe une série d’épreuves visant à tester les différentes capacités de votre cerveau. Temps de réaction, mémoire visuelle ou séquentielles, grâce à BrainCap vous serez capable de vous surpasser mentalement

# 

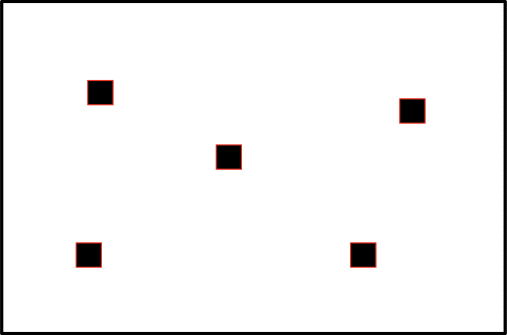
# 

# 

# Epreuves

## Aim trainer

Une cible apparaîtrait à un endroit randomisé sur la page, le but est de cliquer sur cette dernière pour la faire disparaître et ce en mettant le moins de temps possible. Si vous réussissez à le faire, elle réapparaîtra à un autre endroit et ceci pendant 1min30.

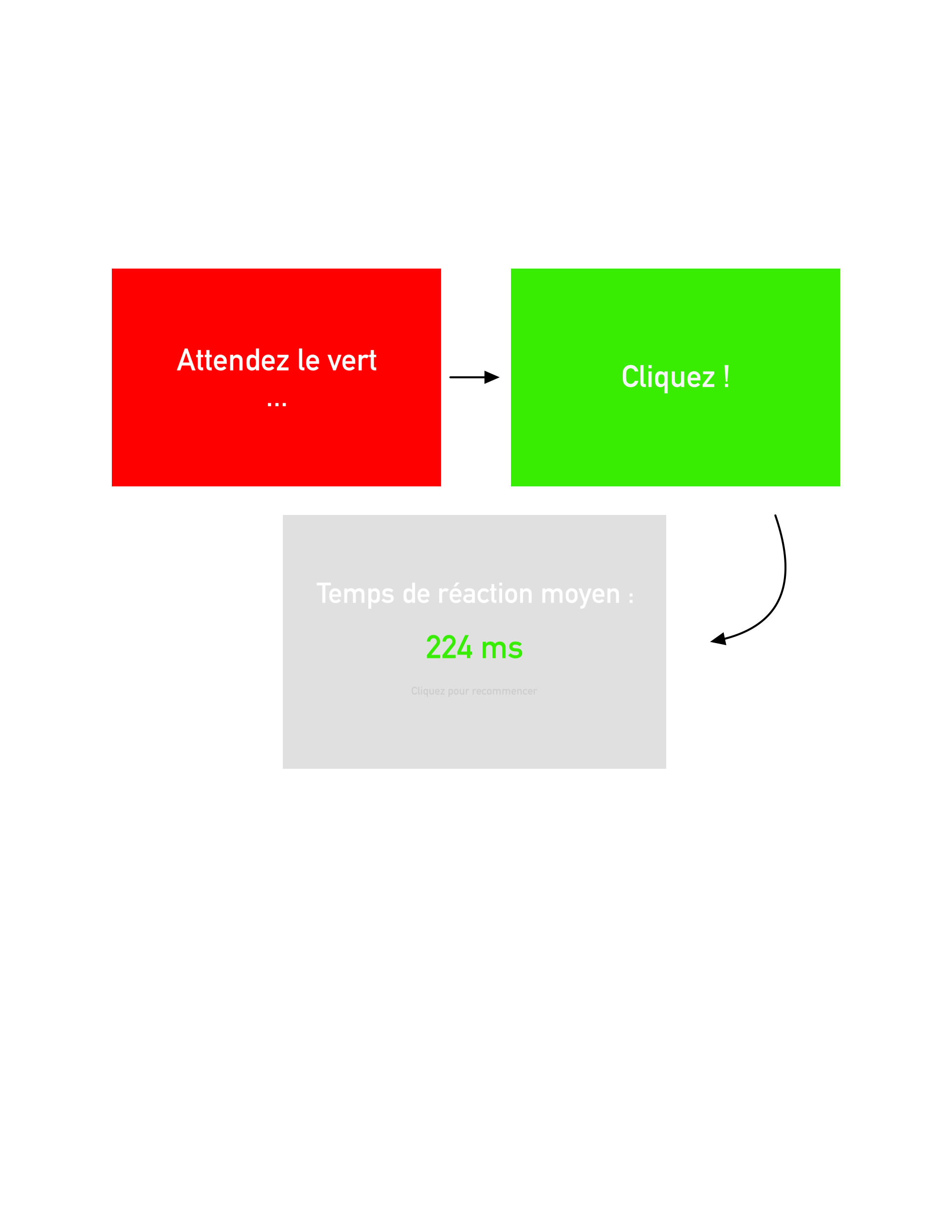


.

## Reaction time

Toute la page apparaît en rouge et vous dit d’attendre. Au bout d’un temps randomisé, elle apparaîtra en vert et vous devrez cliquer sur cette dernière le plus vite possible.

Votre temps de réaction apparaîtra le but étant qu’il soit le plus infime possible.

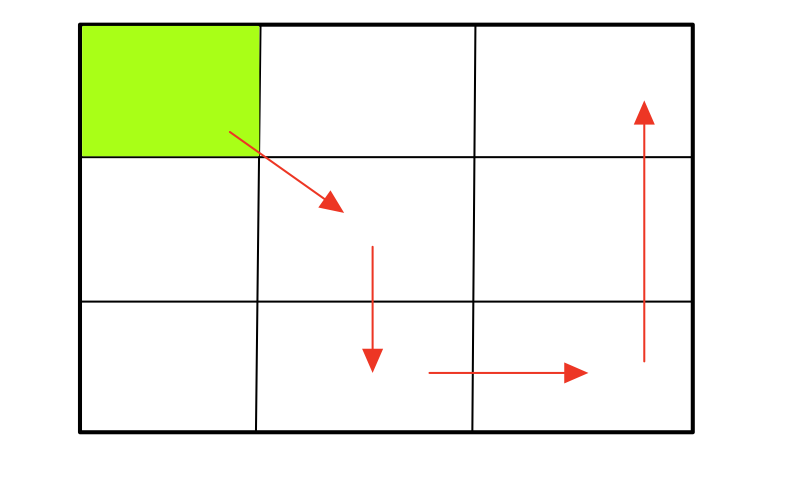


## Sequential memory test

Ce test se présente sous la forme d’une grille composée de 9 carrés. Les carrés vont chacun leur tour s'illuminer en vert en suivant une séquence générée aléatoirement.

Le but est de reproduire la séquence que la machine à créer en retenant l’ordre dans lequel les carrés se sont illuminés. Si le joueur réussit à reproduire la séquence, la machine ajoute un carré à la fin de la séquence. Si il rate, le test s'arrête et son score est affiché.

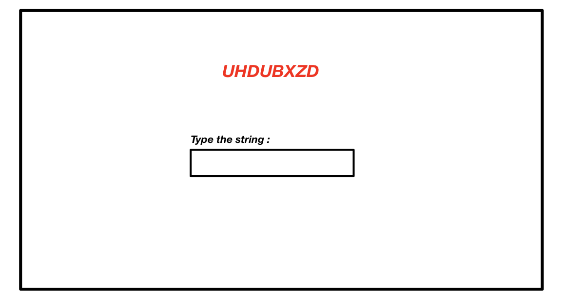
Le score est égal au nombre de carrés s’étant illuminés lors de la dernière séquence réussie.



## Short-term memory / letters

Pour ce test, le joueur doit retenir une suite de lettres aléatoire grandissante (texte rouge). La chaîne de caractère est affichée pendant un temps infini, mais dès que le joueur commence à écrire, elle disparaît. Si le joueur réussit à écrire correctement la chaîne, la machine ajoute une lettre à la fin. Néanmoins, si il se trompe lors de la validation, le test s'interrompt et le score s’affiche.

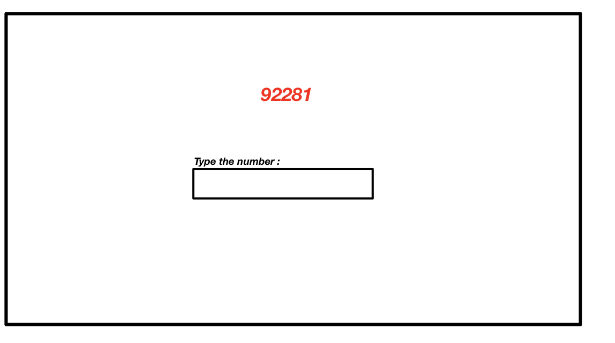
Le score est égal au nombre de lettres ayant été inscrites par l’utilisateur lors de la dernière manche réussie.



## Short-term memory / numbers

Pour ce test, le joueur doit retenir une suite de chiffres aléatoire grandissante (texte rouge). La suite de chiffre est affichée pendant un temps infini, mais dès que le joueur commence à écrire, elle disparaît. Si le joueur réussit à écrire correctement la suite, la machine ajoute un nouveau chiffre à la fin. Néanmoins, si il se trompe lors de la validation, le test s'interrompt et le score s’affiche.

Le score est égal au nombre de chiffres ayant été inscrites par l’utilisateur lors de la dernière manche réussie.



## Color Brainer :

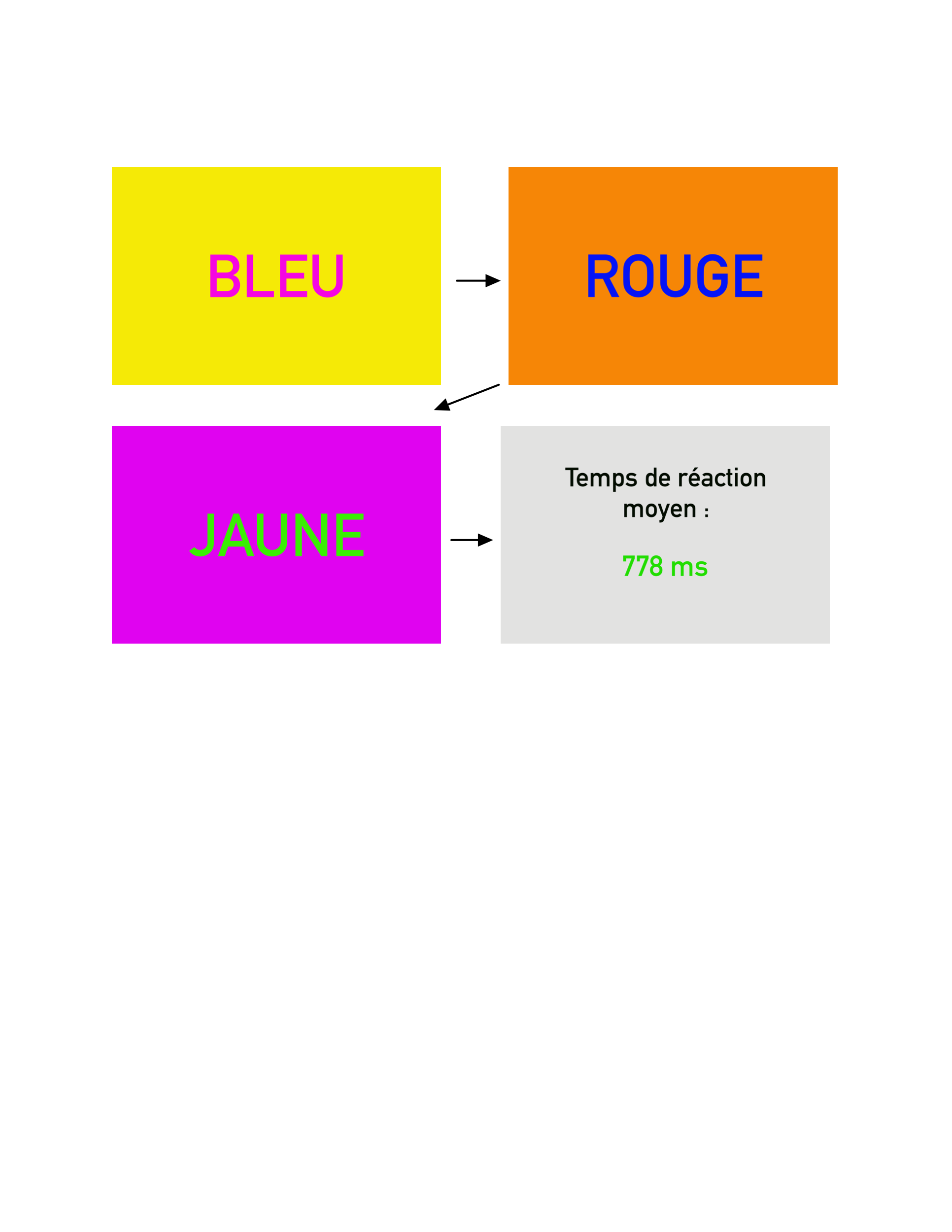
La fenêtre est composée d’un arrière-plan coloré avec, à l'intérieur, une couleur écrite (ex. “bleu”) étant elle-même d’une couleur différente (ex. “bleu” écrit en rose sur fond jaune).

Le but étant d’appuyer sur la touche qui correspond à la couleur du texte. Pour l’exemple cité (image n°1) il faudra appuyer sur la touche “R” du clavier pour “Rose”.

Image n°2 : “B” pour “Bleu”

Image n°3 : “V” pour “Vert”

Le score est calculé en fonction du temps de réaction du joueur.



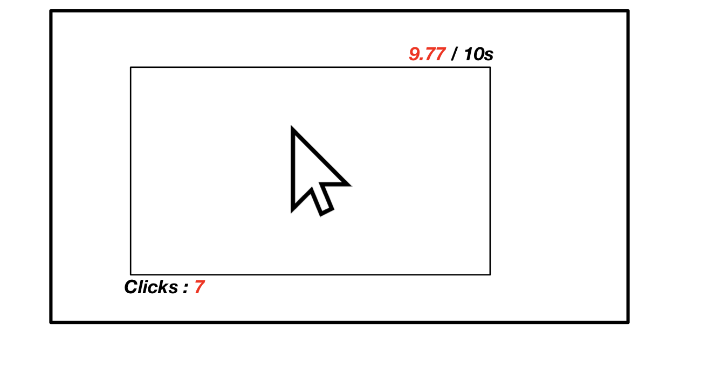
## 

## 

## 

## Click counter

Cette épreuve est simple. Vous avez 10 secondes pour cliquer le plus possible sur votre souris. Le score est affiché à la fin en nombre de clics par seconde.



## 

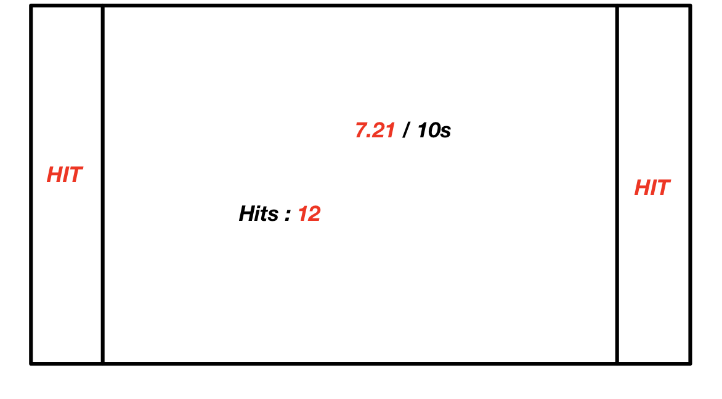
## 

## 

## 

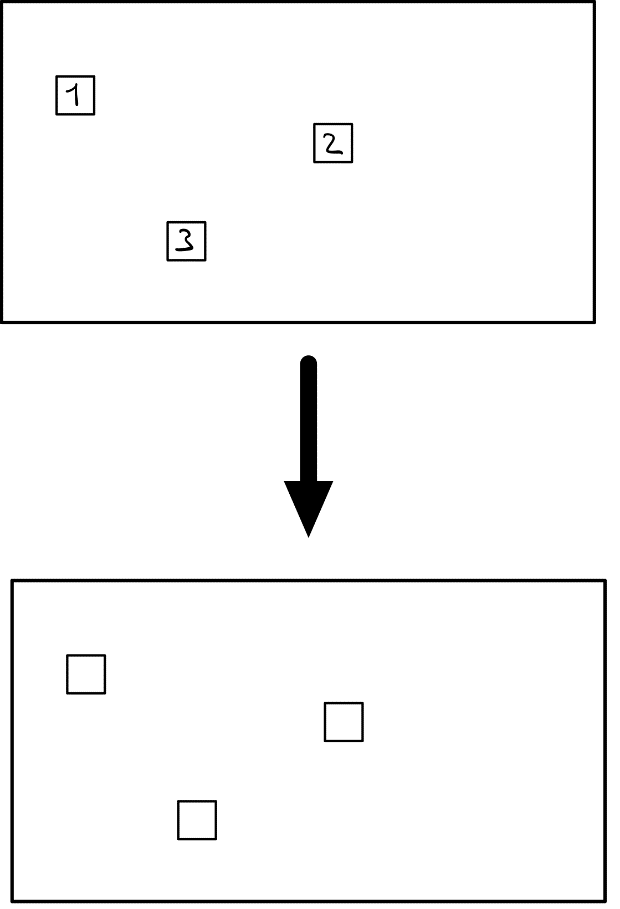
## Flick trainer

Deux bandes vont apparaître aux deux extrémités de la fenêtre et le but est de cliquer sur celle de gauche puis sur celle de droite le plus de fois possible en 10 secondes. Le score est calculé suivant le nombre de clics réussis effectués. Si le joueur clique en dehors des bandes ou si il clique deux fois de suite sur la même bande, il perds des points.



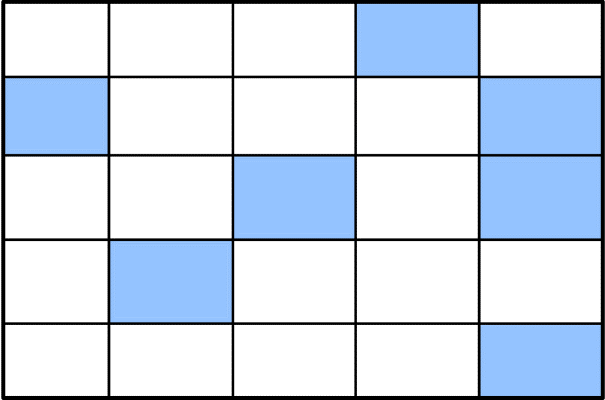
## Chimp test

Deux cases numérotés de 1 à 2 vont apparaitre pendant 5 secondes et les chiffre vont disparaitre et vous devrez vous rappelez de l’ordre et cliquez sur 1 puis deux si vous réussisse la machine vous ajoute le 3 et ainsi de suite.



## Case memory

Une grille de 25 carré. Pour commencer 1 carré passe en surbrillance pendant 3 secondes. Après ces 3 dites secondes le carré s’éteint et laisse place a votre mémoire sélectionnez le bon carré. Si vous gagnez deux carré sur brilleront et ainsi de suite.



# Planification

|  |
| --- |
| Sommaire : |

1. Documentation **(Killian, Hugo)**
2. Création du menu principal **(Hugo)**
3. Mise en relation des différentes formes entre elles **(Killian)**
4. Création du test : Aim Trainer **(Hugo)**
5. Création du test : Click counter **(Hugo)**
6. Création du test : Reaction Time **(Killian)**
7. Création du test : Memory test (chiffre) **(Killian)**
8. Tests **(Killian)**

|  |
| --- |
| Détail des activités : |

|  |
| --- |
| Documentation : |

Scope project -> 3h **(Killian)**

Diagramme de Gantt -> 2h **(Hugo)**

Documentation technique -> 4h **(Killian, Hugo)**

Diapositive de présentation -> 2h **(Killian, Hugo)**

|  |
| --- |
| Création du menu principal : |

Création de l’interface -> 1h

Création des composantes du menu -> 1.5h

|  |
| --- |
| Mise en relation des différentes formes entre elles : |

Adaptation du code pour correspondre au autres formes -> 2h

Correction des potentielles erreurs -> 30 min

|  |
| --- |
| Création du test : Aim Trainer |

Création de l’interface -> 1.5h

Développement -> 4h

|  |
| --- |
| Création du test : Click counter |

Création de l’interface -> 1h

Développement -> 3.5h

|  |
| --- |
| Création du test : Reaction Time |

Création de l’interface -> 1h

Développement -> 4h

|  |
| --- |
| Création du test : Memory test (chiffre) |

Création de l’interface -> 30 min

Développement -> 5h

|  |
| --- |
| Tests utilisateur |

Tester l’application -> 1h

|  |
| --- |
| Temps total estimé : |

34,5h

