|  |
| --- |
| https://s2.qwant.com/thumbr/0x0/5/a/8048bd0e3909c0d61fa4586efbcd99/b_1_q_0_p_0.jpg?u=http%3A%2F%2Fdircom.univ-amu.fr%2Fsites%2Fdircom.univ-amu.fr%2Ffiles%2Flogo_polytech.png&q=0&b=1&p=0&a=1 |
| Fitts’ Law sur téléphone portable |
| Cahier des charges |
|  |
| **Clément STAMEGNA / Guillaume MICHON** |
| **17/04/2018** |

|  |
| --- |
| Cahier des charges du projet ReVa : Fitts’ Law sur téléphone portable. |

[1) Introduction 3](#_Toc511747860)

[1.1. Définition 3](#_Toc511747861)

[1.2. Présentation 4](#_Toc511747862)

[2) Description des tâches 5](#_Toc511747863)

[2.1. Mise en évidence des tâches et fonctions 5](#_Toc511747864)

[2.1.1 - la zone de jeu 5](#_Toc511747865)

[2.1.2 - Les Cibles et les interactions 5](#_Toc511747866)

[2.1.3 - Les Calculs et l’analyse 6](#_Toc511747867)

[2.2. Priorisation des tâches 6](#_Toc511747868)

[2.3. Dépendances inter-tâches 7](#_Toc511747869)

[3) Contraintes Techniques 8](#_Toc511747870)

[Annexes 9](#_Toc511747871)

[Diagramme de Gantt 9](#_Toc511747872)

1) Introduction

# 1.1. Définition

En psychologie expérimentale, en ergonomie et en interaction Homme-machine, la loi de Fitts est un modèle du mouvement humain qui prend en compte un indice de la difficulté d'une tâche. La formulation la plus courante actuelle exprime le temps requis pour aller rapidement d'une position de départ à une zone finale de destination, en fonction de la distance à la cible et de la taille de la cible. La loi de Fitts est utilisée pour modéliser l'acte de « pointer », à la fois dans le vrai monde, par exemple avec une main ou un doigt, et sur les ordinateurs, par exemple avec une souris.

Publiée par Paul Fitts en 1954, elle ne s'applique pas nécessairement aux interfaces tactiles actuelles

Source : https://fr.wikipedia.org/wiki/Loi\_de\_Fitts

# 1.2. Présentation

# C:\Users\stamegna\Desktop\Scan.jpg

2) Description des tâches

# 2.1. Mise en évidence des tâches et fonctions

Lors de notre réunion d’explication du sujet, nous avons pu mettre en évidence plusieurs tâches principales et secondaires.

Nous avons décidé de les classer et de les regrouper en plusieurs parties.

## 2.1.1 - la zone de jeu

La zone de jeu est composée d’une « Page virtuelle », l’écran du téléphone permet d’avoir un aperçu de cette page.

Donc les tâches que nous devrons réaliser sont :

- La création de cette « Page virtuelle »

- La possibilité de déplacer notre écran sur cette page, le « Slide »

## 2.1.2 - Les Cibles et les interactions

Une fois la zone de jeu réalisée, nous avons besoin des cibles nécessaires à l’expérience. Ces deux cibles devront êtres placées sur la zone de jeu à une certaine distance avec une largeur fixe et nous devons pouvoir interagir avec.

Nous devons donc :

- Créer les Cibles

- Positionner les cibles sur la zone de jeu

- Mettre en place les interactions entre elles.

Les interactions sont définies comme suit :

- Si le joueur touche la cible il y a un « Hit »

- Si le joueur rate la cible il y a un « Miss »

- Si le joueur rate la cible mais glisse dessus cela reste un « Miss »

- Si le joueur touche la cible mais glisse en dehors cela compte comme un « Miss ».

## 2.1.3 - Les Calculs et l’analyse

Enfin nous devons utiliser les cibles et la zone de jeu précédemment construites pour effectuer les calculs de temps nécessaires pour l’expérimentation.

Pour cela nous devons mettre en place plusieurs mécanismes :

- Un « timer » pour calculer le temps

- Des structures de données utiles pour les calculs et les résultats

- Des Fonctions pour effectuer les calculs et analyses, notamment les TM (Temps Moyens), les moyennes des TM de tous les essais, les calculs de la Loi de Fitts, les calculs de la Loi de Steering et les comparaisons...

- Un mode d’administration pour paramétriser l’expérience avec le nombre d’essai pour chaque sujet ainsi que de fixer la distance entre les deux cibles.

- De nouvelles interactions sur les cibles pour l’expérience telles que le changement de distance entre les cibles.

## 2.1.4 - La sauvegarde et le transfert des résultats

Une fois les résultats obtenue il faut pouvoir les sauvegarder et les transféré pour des études ultérieures.

- il faut donc pouvoir utiliser un format pour organiser les données et écrire les fonctions associées pour la gestion de ces données.

Pour les Tâches optionnelles nous en avons défini quelques fonctions qui amélioreront l’expérience utilisateur telles que :

- La mise en place d’un menu

- L’apport de nouvelles règles de jeu, par exemple des cibles mouvantes, une zone de jeux en trois dimensions …

- Des graphismes améliorés

# 2.2. Priorisation des tâches

Maintenant que nous avons décrit toutes nos tâches, il faut prioriser les plus importantes et nécessaires au projet.

Suite à la réunion de présentation, nous avons séparé les tâches en tâches principales (cores) et tâches optionnelles.

Le Cœur du projet se situe dans les grandes parties 1, 2 et 3. Toutes les autres tâches sont optionnelles.

Ces parties seront donc traitées en priorité lors des sprints. Les tâches optionnelles seront traitées selon le temps restant pour le projet.

# 2.3. Dépendances inter-tâches

Les Parties principales possèdent des dépendances.

La partie 3 pour les calculs et l’analyse dépendent de la partie 2 les cibles et leurs interactions, en effet l’utilisation des cibles fait partie intégralement de l’expérience donc cette partie est nécessaire.

Elle doit donc être mise en place avant de commencer la partie 3, tous retards pris pour la partie 2 entrainent un retard sur la partie 3.

3) Contraintes Techniques

Voici une liste de paires « Problèmes - Solutions » rencontrées ou pensées par le binôme de projet.

L’Apprentissage du moteur de jeu Unity :

Nous utilisons cette technologie pour la première fois, nous allons donc devoir redoubler d’effort pour apprendre l’utilisation des fonctionnalités d’Unity, qui par chance possède une énorme communauté et beaucoup de tutoriaux ce qui facilitera notre apprentissage.

Le Stockage des données ainsi que leurs envois :

Pour stocker nos données, nous allons sans doute devoir utiliser de la sauvegarde sur fichier pour ceci nous devrons sérialiser nos structures. Ainsi nous pourrons facilement sauvegarder nos données et les recharger au lancement du jeu pour les compléter.

Annexes

# Diagramme de Gantt

