

***TIC – Filière Informatique***

**Projet de semestre 6**

**2018**

**Rapport**

**S.A.S**

**Développement d'une application Web de surveillance  
et de soutien contre l'addiction en ligne**

Logo

**Nicolas Fuchs  
Grégory Ducrey**

Superviseurs : **Sandy Ingram**

**Houda Chabbi**

Mandant : **Dr. Claire Korkmaz**



Fribourg, Mars 2018

Sommaire

[Historique des versions 3](#_Toc509220499)

[1. Introduction 4](#_Toc509220500)

[1.1. Contexte 4](#_Toc509220501)

[1.2. But du projet 5](#_Toc509220502)

[1.3. Document 5](#_Toc509220503)

[2. Analyse 6](#_Toc509220504)

[2.1. Introduction 6](#_Toc509220505)

[2.2. Exploration du sujet 7](#_Toc509220506)

[2.3. Idéation 7](#_Toc509220507)

[2.3.1. Brainstorming 7](#_Toc509220508)

[2.4. Etat de l’art 8](#_Toc509220509)

[2.5. Synthèse 13](#_Toc509220510)

[2.6. Tests technologiques 14](#_Toc509220511)

[2.7. Cahier des charges 14](#_Toc509220512)

[2.8. Conclusion 14](#_Toc509220513)

[3. Conception 15](#_Toc509220514)

[3.1. Introduction 15](#_Toc509220515)

[3.2. Architecture générale 16](#_Toc509220516)

[3.3. Diagrammes de cas d’utilisation 16](#_Toc509220517)

[3.4. Diagrammes de séquences 16](#_Toc509220518)

[3.5. Fiches descriptives 16](#_Toc509220519)

[3.6. Diagramme d’activité ? 16](#_Toc509220520)

[3.7. Conclusion 16](#_Toc509220521)

[4. Implémentation 17](#_Toc509220522)

[4.1. Introduction 17](#_Toc509220523)

[4.2. Conclusion 18](#_Toc509220524)

[5. Validation 18](#_Toc509220525)

[5.1. Introduction 18](#_Toc509220526)

[5.2. Conclusion 19](#_Toc509220527)

[6. Conclusion 19](#_Toc509220528)

[6.1. Conclusions personnelles 19](#_Toc509220529)

[7. Annexes 19](#_Toc509220530)

[7.1. Déclaration d’honneur 19](#_Toc509220531)

[7.2. Sources 19](#_Toc509220532)

# Historique des versions

Cet historique indique à quelles dates ont eu lieu les modifications du cahier des charges :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Semaine** | **Date** | **Personnes** | **Chapitres** | **Modification** |
| 0.1 | A3 | 05.03.18 | G. Ducrey N. Fuchs | 1, 1.1, 1.2, 1.3 | Création de la table des matières,  Introduction |
| 0.2 | A4 | 12.03.18 | G. Ducrey N. Fuchs | 2, 2.4 | Introduction à l’analyse,état de l’art et synthèse |

# Introduction

Ce chapitre présente les aspects suivants du projet :

* Le contexte et motivations de réalisation du projet
* Les objectifs
* Le document

## Contexte

Selon le monitorage suisse de l'addiction, un peu plus de 7% des jeunes Suisses entre 15 et 19 ans utilisent le Web de manière compulsive et problématique (addiction aux jeux de rôle en ligne, jeux d'argent, négligence des devoirs et des activités hors-ligne). L'objectif du projet est de concevoir et développer une application minimisant l'effet des stimulis addictifs, et redonnant le contrôle aux "consommateurs". Pour cela, un monitoring automatique des activités du consommateur (avec son consentement) doit être mise en place suivi de mesures de sensibilisation proposées en temps-réel.

Pour détecter si une personne est dépendante aux jeux vidéos, il y a plusieurs critères qui sont les suivants selon wikipédia :

* Monopolisation de la pensée par le projet de comportement addictif
* Intensité et durée des épisodes plus importants que souhaités à l'origine
* Tentatives répétées pour réduire, contrôler ou abandonner le comportement
* Temps important consacré à préparer les épisodes, à les entreprendre ou à s'en remettre
* L'engagement dans le comportement est tel que la personne ne peut plus accomplir des gestes élémentaires (se laver, se nourrir) et le conduit vers un désinvestissement social, professionnel et familial
* Poursuite du comportement malgré l'aggravation des problèmes sociaux et en dépit de la connaissance des conséquences négatives
* Tolérance marquée, c'est-à-dire besoin d'augmenter l'intensité ou la fréquence pour obtenir l'effet désiré, ou diminution de l'effet procuré par un comportement de même intensité
* Agitation, irritabilité et surtout angoisse si le passage à l'acte addictif est différé, empêché

On constate que pour ces gens-là, les conséquences néfastes peuvent être les suivantes :

* Appauvrissement de la vie sociale
* Manque de sommeil
* Trouble du comportement (saute d’humeur…)
* Mauvaise hygiène de vie
* Perte du sens des réalités
* Atteinte à la santé physique (maux de tête, yeux secs, douleurs au dos, manque d’activité physique, depression, suicide)

Ces liens contiennent quelques témoignages qui permettent de comprendre l’importance de la gestion des stimulis addictifs pour les joueurs. En effet, il n’est pas néfaste en soit de jouer, mais il est absoluement nécessaire de garder le contrôle.

<http://www.jeuxvideo.com/dossiers/00018270/l-addiction-aux-jeux-video-temoignages-de-joueurs-006.htm>

<https://www.vice.com/fr/article/ppvzd7/dependance-jeux-video-843>

## But du projet

Le projet consiste à développer un produit permettant aux utilisateurs de reprendre le contrôle lorsqu’ils sont soumis à des stimulis addictifs quand ils utilisent des jeux vidéos/internet.

L’enfant ou l’adolescent est encore en construction et il n’a pas encore tous les filtres lui permettant de comprendre les enjeux du temps passé sur un écran. C’est pourquoi cette application permettra de mettre un cadre où le joueur sera confronté à une responsabilisation de ses choix. A chaque âge le cadre doit être différent. Il sera donc plus restrictif pour des enfants que des adolescents.

Ce travail est réalisé dans le cadre du projet de semestre 6 en informatique à la haute école d’ingénieurs et d’architectes de Fribourg. Il est réalisé par M.Nicolas Fuchs et M.Grégory Ducrey pour la mandante Dr. Claire Korkmaz et il est supervisé par Mme Sandy Ingram ainsi que Mme Houda Chabbi.

## Document

Ce rapport fait partie intégrante du projet. Il contient des informations concernant les étapes de réalisation du produit final. La création de ce document se fait tout au long du projet. Il doit permettre au client de suivre les démarches de travail et si le projet devait être repris, le rapport donne toute les clés au développeur. Le document contient des informations sur les phases suivantes :

* Analyse
* Conception
* Implémentation
* Validation

Le rapport permet aux évaluateurs de vérifier la qualité et la rigueur du travail effectué.

# Analyse

## Introduction

Ces 5 semaines d’analyse permettent de débuter le projet et de nous diriger dans la bonne direction. Nous ne connaissons pas énormément le sujet de la prévention des addictions aux jeux vidéos et aux réseaux sociaux, c’est pourquoi il est important d’étudier la base de ce sujet avant de réfléchir à un produit pertinent.

Le client souhaite une application qui permet de monitorer et redonner le contrôle à l’utilisateur afin que ce dernier fasse des choix et en prenne l’entière responsabilité. Dans le chapitre 1.1 traitant du contexte du projet, des témoignages sont recueillis et ils attestent des dangers de ces addictions. En observant les critères d’addictions décrits, nous pouvons constater que le danger découle du temps passé sur les jeux ou les réseaux sociaux. C’est pourquoi nous allons en partie orienter nos recherches sur des moyens pour gérer le temps.

Nos recherches doivent nous aider à créer un produit répondant aux besoins de manière pertinente. Dans cette analyse nous allons effectuer les travaux suivants :

* Exploration du sujet 🡪 se renseigner sur le contexte, des statistiques
* Etat de l’art 🡪 Explorer les produits de prévention présents sur le marché
* Test de produits existants 🡪 Tester certains de ces produits
* Tests technologiques 🡪 Tester certaines méthodes qui pourraient nous aider à implémenter des fonctionnalités (Récupération des processus, récupération des URL…)
* Idéation 🡪 Sur la base du travail accompli, faire un brainstorming de ce qui serait pertinent d’implémenter pour ce produit
* Cahier des charges 🡪 après ces 5 semaines d’analyse, établir un cahier des charges final décrivant le produit que nous allons réaliser en ayant une meilleure connaissance du sujet

Les enjeux de cette analyse sont assez grands. C’est pourquoi elle dure 5 semaines sur les 12 prévues pour le projet.

## Exploration du sujet

## Idéation

Pour commencer ce projet, nous nous sommes dans un premier temps renseignés sur le sujet des addictions (voir chapitre « 1 introduction ») afin d’avoir une meilleure vue d’ensemble. Une fois fait, nous avons effectué un brainstorming des idées de fonctionnalités qui pourraient être intéressantes lors du développement de l’application. Ce brainstorming est effectué avant les tests d’applications existantes et de tests technologiques. Il a pour but de laisser aller notre créativité sans être limité par des recherches déjà effectuées.

### Brainstorming

Voici les résultats du brainstorming, nous avons classé les idées par catégories :

**Concept de l’application**

* Contrôle des sites visités
* Gestion du temps
* Architecture client/serveur et paramétrisation de l’application native depuis la page web du serveur
* Système de récompense
* Contrôle parental
* Encouragement à la mobilité physique
* Blocage de certains processus en fonction d’une localisation GPS
* Validation par l’utilisateur

**Fonctionnalités concrètes**

* Monitoring du temps
  + - Paramétrer en fonction de l’âge
* Module statistique —> pour l’utilisateur et/ou contrôleur
  + - Temps passé sur une semaine, et sur quel sujet
* Alerte visuelles et sonores ET Avertissement sur les dangers
  + - Forcer des pauses
    - Vidéo de sensibilisation
    - Blocage du processus durant la vidéo
    - L’utilisateur est prévenu de l’interruption
* Filtrer les sites visités (avec une liste, ou avec un thème)
* Vérification de l’USB (Si le jeux est lancé par disque)

**Moyens techniques**

* Récupération des processus
* Chronométrage du temps d’activité des processus
* Interruption de processus
* Récupération d’adresse IP et de localisation

**Plateforme**

* Application mobile multiplateforme
* Natif mac os
* Natif PC
* Natif Linux
* Jeux vidéos WEB

**Moyens de prévention**

* Liste des sites/processus sensibles a permettre/interdire
* Jeux

**À analyser:**

* Si l’on peut récupérer les processus sur les téléphones mobile
* Si on peut connaître le contenu des pages web ou des URL? (API, traitement de l’html…)
* Existe t’il des vidéos de prévention que l’on à le droit d’utiliser?

## Etat de l’art

Nous avons exploré plusieurs applications existantes permettant de mieux gérer le temps passé sur l’ordinateur ou le smartphone. Le but de cette observation est de se rendre compte de ce qui existe sur le marché et réfléchir à des fonctionnalités que nous pourrions apporter en plus. Pour pouvoir analyser correctement, nous avons examiné les aspects suivants :

* Le type de blocage
* Orienté web/natif/mobile 🡪 Si l’application travaille sur les navigateurs, sur les mobiles ou en natif sur l’ordinateur. Il est possible qu’une application native monitore des navigateurs.
* La plateforme sur laquelle le logiciel fonctionne
* Base volontaire ou restrictions🡪 est-ce que l’application a tendance à interdire ou à sensibiliser
* Public cible🡪 quel type de personne est succeptible d’utiliser le logiciel ?
* Paramétrer en fonction de l’âge 🡪 Est-ce possible de changer le monitoring ou les restriction en fonction de l’âge de l’utilisateur ?
* Redonne le contrôle 🡪 est-ce que l’utilisateur peut décider ce qu’il fait ?
* Difficulté de prise en main🡪 critère subjectif
* Administration depuis la même machine
* Fournit des statistiques d’utilisation
* Avis des utilisateurs

Ainsi, sur la base de ces observations nous pouvons comparer différentes applications et nous rendre compte de l’utiliter des fonctionnalités pour notre client. Nous ne souhaitons pas plagier mais nous inspirer de ce qui existe déjà et créer un produit adapté aux besoins de nos clients.

Les applications testées proviennent des sites suivants :

<http://humanetech.com/take-control/>

<http://selfcontrolapp.com/index.html>

|  |  |
| --- | --- |
| **Self control** | |
| **Utilisation** | |
| L'utilisateur choisit les sites webs ou les services mail à bloquer. Il choisit également la durée. Une fois fait il active la restriction et tant que le temps n'est pas écoulé c'est impossible d'accéder à ce qui est bloqué. (même en cas de suppression de l'application où de redémarrage). | |
| **Critère** | **Observation** |
| Blocage | Bloque les sites web (black list ou white list). Bloc les mails. Ne bloque pas des applications |
| Orienté web/natif/mobile | Application native orientée web |
| Plateforme | Fonctionne sur Mac OS |
| Base volontaire ou restrictions | L'utilisateur décide lui même ce qu'il autorise ou non |
| Public cible | Personnes ayant la maturité de savoir ce qui est bon pour lui ou non. L'administration depuis une autre machine n'est pas possible |
| Paramétrer en fonction de l'âge | Il est possible d'adater la liste des sites à bannir. |
| Redonne le contrôle | Redonne le contrôle au bout d'un certain temps |
| Difficulté de prise en main | Très facile à utiliser |
| Administration depuis la même machine | Il faut activer le logiciel sur la session à bloquer |
| Fournit des statistiques d'utilisation | Non, il ne fournit pas de statistiques |
| Avis des utilisateurs | Bon logiciel |
| Payant | Le logiciel est gratuit et opensource |
| Offline | Est inutile en hors ligne puisqu’il monitore uniquement ce qui est en ligne |
| **Remarque complémentaire** | |
| L'outil est vraiment très simple. Il serait certainement utile pour se couper des distractions durant une journée de travail par exemple. L'utilisateur doit avoir une certaine maturité. **On a accès au code de l'application** | |

|  |  |
| --- | --- |
| **Contrôle parental Mac os** | |
| **Utilisation** | |
| L'utilisateur de la session administrateur donne des droits aux utilisateurs des autres sessions en activant le contrôle parental. | |
| **Critère** | **Observation** |
| Blocage | Assignation de plages horaires et de durée d'utilisation de l'ordinateur. Black-List/white-List) de sites web et limitation des sites web pour adultes. Blocage d'applications et de stores |
| Orienté web/natif/mobile | Application native |
| Plateforme | Fonctionne sur Mac OS |
| Base volontaire ou restrictions | Ne fonctionne que sur la base de restrictions. L'utilisateur de la session contrôlée ne peut pas changer quoi que ce soit aux restrictions, il subit des interdits. |
| Public cible | Les enfants particulièrement. Ce serait plutôt sur un ordinateur familial où il y a plusieurs sessions que le contrôle parental serait pertinent |
| Paramétrer en fonction de l'âge | Tout est adaptable en fonction de ce que l'administrateur souhaite pour l'utilisateur |
| Redonne le contrôle | Ne donne aucun contrôle. L'application interdit mais n'apprend pas à gérer |
| Difficulté de prise en main | Simple d'utilisation, mais beaucoup d'options |
| Administration depuis la même machine | Les restrictions sont élaborées depuis la session administrateur de la machine |
| Fournit des statistiques d'utilisation | Il conserve les historiques de consultation de site webs |
| Avis des utilisateurs | Assez bon |
| Payant | Le logiciel est compris dans le prix de l’OS |
| Offline | Le contrôle parental fonctionne avec ou sans connexion internet |
| **Remarque complémentaire** | |
| Ce logiciel est très solide et permet de bien gérer d'un point de vue administrateur. Mais il ne permet pas à l'utilisateur de prendre conscience du temps qu'il passe sur telle ou telle application. L'idéal serait qu'il sorte des statistiques d'utilisations et qu’il permette des alertes sans créer de blocage | |

|  |  |
| --- | --- |
| **Rescue time** | |
| **Utilisation** | |
| L'utilisateur installe l'application sur son appareil. Ensuite il paramètre la surveillance sur la page web. L'application répertorie tout le temps passé sur chaque application par l'utilisateur. Elle permet de fixer des objectifs et en fonction de ceux-ci, elle affiche des alertes. L'application ne bloque rien | |
| **Critère** | **Observation** |
| Blocage | Ne bloque rien, fournit des statistiques |
| Orienté web/natif/mobile | Application native dont les statistiques sont disponibles sur une page web |
| Plateforme | Fonctionne sur Mac os, Linux, Windows et Android. Une version iOS est en cours de développement |
| Base volontaire ou restrictions | Base volontaire. Le "dashboard" ne bloque rien, il permet de fixer des objectifs donne des alertes. Ex: Objectif, <2h sur facebook. Si le temps est atteint, une alerte approche |
| Public cible | Tout public |
| Paramétrer en fonction de l'âge | Oui, le parent pourrait fixer des objectifs pour l'enfant, et ce dernier ne verrait que les alertes. |
| Redonne le contrôle | Il redonne le contrôle grâce aux alertes, cela met l'utilisateur devant un choix. |
| Difficulté de prise en main | Il y a quelques options à décider, et il faut indiquer pour certain sites web s'ils sont sources de distraction ou non |
| Administration depuis la même machine | L'application est installée sur une machine, et on peut voir les statistiques sur une page web. |
| Fournit des statistiques d'utilisation | Donne des statistiques à la seconde près sur le temps passé sur les pages web, les applications. |
| Avis des utilisateurs | Très bon logiciel |
| Payant | Parties gratuites et payantes |
| Offline | Offline, l’utilisation n’est pas monitorée. Sauf en partie payante |
| **Remarque complémentaire** | |
| Cette application correspond beaucoup à ce que l'on souhaite faire : Redonner le contrôle.  **Une API est disponible** pour travailler avec les données récupérées par ce dashboard. Nous allons explorer les possibilités dans le chapitre « 2.5 tests technologiques ». La partie payante de l’application permet l’utilisation d’un bracelet paramétrable pour envoyer des décharges électriques en cas d’utilisation nocive d’application (à gérer en fonction des objectifs). Le nom de l’app est « Pavlok ». Plusieurs autres plugin utilisent les données.  <https://www.rescuetime.com/anapi/setup/overview> | |
| **Moment** | |
| **Utilisation** | |
| L'utilisateur télécharge l'application sur son produit iOS. Il l'ouvre et il autorise les accès par l'app aux photos ainsi qu'à la localisation. Une fois fait, il doit encore activer le traçage des apps utilisées. | |
| **Critère** | **Observation** |
| Blocage | Ne bloque rien, monitore |
| Orienté web/natif/mobile | Application native pour iOS. Monitore les applications, non pas les sites web. |
| Plateforme | iOS |
| Base volontaire ou restrictions | Base volontaire. Ce n'est que du monitoring. Pas de blocage. |
| Public cible | Toute personne utilisant un produit avec iOS |
| Paramétrer en fonction de l'âge | Non |
| Redonne le contrôle | Oui, car il peut donner des alertes |
| Difficulté de prise en main | Très simple |
| Administration depuis la même machine | Les statistiques apparaissent sur le mobile |
| Fournit des statistiques d'utilisation | Oui |
| Avis des utilisateurs | Bonne application (4/5 sur l'apple store) |
| Payant | Gratuit |
| Offline | L’utilisation est la même offline |
| **Remarque complémentaire** | |
| L'application affiche le temps passé par app. Et elle donne un fil rouge pour la journée. Cependant elle n'est pas capable de filtrer le contenu web. Par contre, étant donné que les réseaux sociaux ont une app en général, c'est tout de même intéressant si les gens l'utilisent. Le produit est très semblable à Rescue Time, sauf qu’il est créé pour iOS. Une version Android est en cours de développement. L’application compte le nombre de fois que l’on dévérouille le smartphone. | |

### Synthèse

Durant ces recherches nous avons pu constater que les applications existantes permettent de réaliser déjà un bon monitoring ou contrôle par rapport à ce que fait l’utilisateur. Nous trouvons des produits qui interdisent l’accès à des ressources, d’autres qui informent l’utilisateur de son utilisation.

La plupart de ces produits sont gratuits, voir même opensource. L’application « Rescue Time » nous semble être celle qui est la plus complète. Elle permet dans la version gratuite de :

* Sortir des statistiques d’utilisation des applications desktop ou encore des pages web visitées (rapports journaliers) sur une page web
* Créer des alertes
* Classer les occupations par niveaux de productivité (mail = productif, Spotify != distraction), selon l’appréciation de l’utilisateur

L’application payante permet de :

* Monitorer le temps Offline
* Bloquer des sites web pendant une durée déterminée (comme self control)
* Avoir des rapports d’utilisations plus détaillés
* Créer des alertes personnalisées

Le tout pour 9$ par mois ou 72$ par année : <https://www.rescuetime.com/rescuetime-pro>

Nous trouvons vraiment puissant et efficace cette application. Cependant nous pourrions ajouter des services à partir de la récupération des données tels que :

* Envoie d’email à un parent lorsqu’un temps est dépassé
* Blocage volontaire d’applications desktop (pas uniquement des sites web)
* Les fonctionnalités seraient toutes gratuites (l’application est un pack complet), cela signifie que l’on peut reprendre l’idée de leur fonctionnalités payantes
* Sortir des graphiques d’utilisation

L’idéal serait un mélange des fonctionnalités de Self Control et de Rescue Time. Ainsi l’utilisateur aurait le contrôle sur ce qu’il fait, mais le contrôle lui serait suggéré en lui proposant de bloquer des sites et des applications durant un temps donné.

Rescue Time propose une API qui permet de récupérer les données. Nous allons en parler dans le chapitre « 2.6 tests technologiques ».

## Tests technologiques

Cette section présente les différents tests technologiques menés lors de la phase d'analyse afin de déterminer les fonctionnalités qui sont faisables ou non.

Le premier test concerne l'analyse des processus qui tournent sur un ordinateur portable. Le programme écrit en langage java fonctionne sur les systèmes d'exploitation suivants :

* Windows 10
* Mac OSX
* Kali Linux Rolling

Deux façons de récupérer les processus ont été implémentées. La première solution consiste à lancer une commande système depuis le programme java. Cependant, cette solution n'est pas générique. Il faut donc créer deux versions différentes : Une version pour windows et une autre version pour les systèmes UNIX (Mac OSX et Linux). Cette manière de faire demande un traitement (parsing) pour récupérer les noms des processus.

La deuxième solution utilise une librairie java nommé oshi qui fournit aux développeurs des informations sur le système d'exploitation. Cette solution est moins rapide mais offre des options de tris intéressantes de même que d'autres données importantes liées aux processus détectés, telles que le processID ou le chemin de l'exécutable par exemple.

Ces noms d'exécutables ainsi que leurs chemins seront par la suite analysés pour connaître la nature des applications qui tournent sur la machine (jeux/réseaux sociaux).

Une liste des processus peut aussi être récupérer sur des appareils mobiles, notamment avec le langage C#. Ce langage est utilisé par Xamarin pour le développement d'application cross-plateform. Sur Android, on peut aussi utiliser le langage java. En ce qui concerne IOS, l'accès aux informations système semble plus complexe car à partir de la version IOS 9, l'accès aux données des autres applications est très restreint.

Le deuxième test se concentre sur l'analyse des sites web visités. Il a aussi été codé avec le langage Java. Il n'a par contre été testé que sur Windows 10. Deux manières ont été implémentées : La première au travers d'une librairie et la deuxième au travers d'une extension de navigateur.

La première solution utilise la librairie libpcap (utilisée par l'outil wireshark) enrobée dans un wrapper java. Il en existe plusieurs sortes plus ou moins efficaces et faciles d'intégration dans un projet. La première librairie testée se nomme pcap4j. Elle est malheureusement pas utile pour ce projet car les protocoles que l'on désire capturer sont HTTP et HTTPS de la couche 7 du modèle OSI. Ces protocoles ne sont pas supportés. La seconde librairie testée se nomme jnetpcap. Le premier inconvénient est qu'elle est capable de capture le protocole HTTP mais pas le HTTPS qui est un protocole sécurisé et donc chiffré.

Après les différents tests avec ces deux librairies, la manière de capturer le traffic web a été remise en cause. Lors de la capture de paquets sur le réseau, on obtient uniquement les entêtes HTTP mais aucune information concernant les onglets sur lesquels l'utilisateur est actif ainsi que le temps passé sur ceux-ci individuellement. Grâce aux applications Rescue Time et le contrôle parental de Swisscom, on a pu découvrir une autre façon énormément plus efficace pour contrôler les sites sur lesquels va l'utilisateur.

Cette deuxième méthode fonctionne grâce à une extension google chrome. Cette extension est également développable sur d'autres navigateurs comme firefox par exemple. L'extension a accès directement aux onglets du navigateur ainsi qu'aux urls tapées. Le temps aussi peut être analysé. A chaque changement/ouverture d'onglet, l'extension récupère le temps passé sur l'onglet précédent et envoie ces informations à un serveur http codé en java.

Les extensions google chrome ne sont installables ni sur Android ni sur IOS. Par contre, les extensions firefox sont disponibles uniquement sur Android.

## Cahier des charges

## Conclusion

# Conception

## Introduction

## Architecture générale

## Diagrammes de cas d’utilisation

## Diagrammes de séquences

## Fiches descriptives

Avec les besoins IHM. Peut-être regrouper les diagrammes de séquences les fiches dans un seul chapitre.

## Diagramme d’activité ?

## Conclusion

# Implémentation

## Introduction

## Conclusion

# Validation

## Introduction

## Conclusion

# Conclusion

## Conclusions personnelles

# Annexes

## Déclaration d’honneur

## Sources

<http://www.ot-lab.ch/?p=5605>

<https://fr.wikipedia.org/wiki/Dépendance_au_jeu_vidéo>

<http://www.jeuxvideo.com/dossiers/00018270/l-addiction-aux-jeux-video-impact-de-l-addiction-aux-jeux-video-sur-la-vie-quotidienne-005.htm>

<http://tpe-addiction-jeux.e-monsite.com/>

<http://humanetech.com/problem#team>