



# DOCUMENTATION TECHNIQUE



**IOnlyThink**  
A BRAIN-COMPUTER INTERFACE SYSTEM

Septembre  
2012

## Documentation technique du projet I Only Think

*I Only Think* (IOT) est une interface neuronale directe permettant, à l'aide d'un casque électro-encéphalographique (casque EEG), de réaliser des actions simples sur ordinateur directement par la pensée. *IOT* se veut de rendre accessible le monde informatique aux personnes handicapées ainsi qu'aux utilisateurs ne pouvant pas se servir de périphériques d'entrée standard (Clavier et Souris).

---

**DOCUMENTATION TECHNIQUE DU PROJET I ONLY THINK**

<b>I – PRESENTATION DU PROJET</b> .....	<b>2</b>
I.I – Définitions, contexte et historique .....	2
I.II – Spécifications .....	2
I.III – Prérequis .....	3
I.III a – Configuration matérielle .....	3
I.III b – Configuration logicielle .....	3
I.IV – Sources.....	4
I.IV a – Téléchargement des sources .....	4
I.IV b – Documentation des sources.....	4
<b>II – DIAGRAMME GLOBAL DU PROJET</b> .....	<b>5</b>
<b>III - DETAILS SUR LE PROJET</b> .....	<b>6</b>
III.I – Le Contrôleur d’Environnement Graphique (CEG).....	6
III.II – Le <i>Builder</i> .....	7
<b>IV – DIAGRAMME DE COMMUNICATION</b> .....	<b>8</b>
<b>V – CONCEPTION</b> .....	<b>9</b>
V.I – Choix des technologies.....	9
V.II – UML / Diagramme de classes.....	9
<b>INFORMATIONS COMPLEMENTAIRES</b> .....	<b>10</b>
Licence .....	10
Contact .....	10
Auteurs .....	10

## I – PRESENTATION DU PROJET

### I.1 – Définitions, contexte et historique

*I Only Think* est un projet développé depuis 2009 en tant qu'EIP (EPITECH Innovative Project).

Un EIP est un projet interne à l'école d'informatique EPITECH. Il s'agit d'un projet déployé sur deux ans durant les deux dernières années du cursus (4<sup>ème</sup> et 5<sup>ème</sup> année). Il est la concrétisation du travail réalisé pendant les 5 années d'études effectuées au sein de cette école. Un EIP est un projet concret, établi de toutes parts par un seul groupe d'étudiants et suivi par une équipe pédagogique dédiée.

*IOT* est une interface neuronale permettant à l'aide d'un casque EEG de réaliser des actions simples sur ordinateur directement par la pensée. Il fut tout d'abord créé par un premier groupe d'étudiants en 2009. Il fut à cette époque développé en partenariat avec l'INRIA (Institut National de Recherche en Informatique et en Automatique), maison mère du projet OpenViBE.

OpenViBE est une solution logicielle permettant de concevoir, tester et utiliser des interfaces cerveau-machine. Il permet notamment dans le cadre d'*IOT* d'interfacer les différents types de casques EEG en assurant une totale interopérabilité avec ceux compatibles OpenViBE.

Un casque EEG est un système d'électrodes permettant de réceptionner certains signaux émis par l'activité cérébrale de notre cerveau. Ces signaux analogiques sont ensuite amplifiés et transformés en signaux numériques permettant d'être interprétés par un ordinateur à l'aide d'un boîtier d'acquisition.

En 2011, le projet fut repris par un second groupe d'étudiants, également dans le cadre d'un EIP à EPITECH jusqu'à mi-2013.

### I.2 – Spécifications

*IOT* utilise un casque électro-encéphalographique (casque EEG) permettant d'analyser et interpréter les pensées d'un utilisateur ainsi qu'une caméra (ou webcam) permettant l'émulation d'un curseur via un principe d'eye-tracking (suivi du mouvement des yeux). Ces deux technologies sont utilisables indépendamment ou simultanément.

Le projet se base sur le logiciel OpenViBE pour l'interprétation des signaux émanant du casque EEG, permettant une totale compatibilité avec les casques supportés par OpenViBE.

Le projet se découpe en deux parties :

- Le *Builder* : Un outil simple et ergonomique permettant de réaliser des calques personnalisés pour n'importe quel programme. Ces calques permettent de choisir les zones sélectionnables et cliquables de chaque application.
- Le *CEG* (Contrôleur d'Environnement Graphique) : Il s'agit du cœur du projet permettant de naviguer à l'aide du casque EEG et/ou de l'eye-tracking. Le *CEG* est capable de charger les calques générés à l'aide du *Builder* pour ensuite les associer aux programmes ou applications correspondantes. Le *CEG* possède également un calque standard utilisable avec tous les programmes.

---

## I.III – Prérequis

### I.III a – Configuration matérielle

IOT nécessite la configuration matérielle suivante afin de fonctionner de manière optimale :

- Un casque EEG compatible OpenViBE (liste de compatibilité : <http://openvibe.inria.fr/supported-hardware/>)
- Une caméra ou webcam haute résolution (permettant l'utilisation de l'eye-tracking)
- Un ordinateur avec processeur double cœur recommandé (nécessaire pour l'utilisation de l'eye-tracking)

### I.III b – Configuration logicielle

IOT nécessite également la configuration logicielle suivante :

- La suite OpenViBE (téléchargeable à l'adresse suivante : <http://openvibe.inria.fr/downloads/>)
- Un système d'exploitation Microsoft Windows 7 (32 bits ou 64 bits) ou Linux (testé sous Ubuntu et ArchLinux)

## I.IV – Sources

### I.IV a – Téléchargement des sources

Les sources de *IOT* sont disponibles sous forme d'une archive sous divers formats sur notre page Web rubrique "Téléchargement".

Vous pouvez les télécharger en vue de les compiler, les lire, ou les modifier à votre guise.

Les instructions d'installation concernant ces sources ainsi que la liste des dépendances externes font l'objet d'une documentation externe disponible au sein même de l'archive ou sur notre page Web rubrique "Documentation".

Des hash MD5 de contrôle de contenu sont générés à chaque publication afin de garantir la sécurité des utilisateurs. Nous vous encourageons vivement de vérifier tout fichier téléchargé à l'aide de ces hash de contrôle.

### I.IV b – Documentation des sources

Une documentation Doxygen est disponible afin de simplifier au maximum la compréhension des sources.

Vous retrouverez cette documentation sur notre page Web rubrique "Documentation" puis "Documentation des sources".

## II – DIAGRAMME GLOBAL DU PROJET



Le casque électro-encéphalographique (casque EEG), constituée d'un casque munis d'électrodes et de capteurs, couplé à un boîtier d'acquisition, a pour mission de convertir et de transmettre les ondes cérébrales en signaux numériques compréhensible par le serveur d'acquisition.

Le serveur d'acquisition est un composant logiciel qui récupère ces signaux, les transforme en données exploitables puis les envois à la partie OpenViBE.

La partie OpenViBE est un composant logiciel chargé d'analyser et traiter les informations. L'information traitée est transformée sous la forme d'évènement compréhensible par le CEG et est transmise à ce dernier.

Le CEG est un composant logiciel installé sur la machine destinée à être contrôlée. Il se charge d'appliquer au système d'exploitation l'action adéquate.

Se rattachant à ce composant, le *Builder* permettra de sélectionner personnellement des zones à l'écran afin de faciliter la navigation sur l'interface.

La partie eye-tracking s'intégrera quant à elle avec le composant logiciel CEG afin de rendre la navigation sur l'interface beaucoup plus fluide et ergonomique, pour les utilisateurs le désirant.

### III - DETAILS SUR LE PROJET

#### III.1 – Le Contrôleur d'Environnement Graphique (CEG)

Le CEG est la partie principale du projet, il permet de contrôler un système d'exploitation à l'aide d'un casque EEG et/ou d'une caméra ou webcam. Son fonctionnement repose sur la librairie graphique Qt 4 (voir la liste des dépendances externes pour plus d'informations) et est développé en C++.

Son principe est très simple : superposer un calque générique (quadrillage à taille paramétrable) à l'écran de l'utilisateur pour lui permettre, à l'aide d'actions très simples (à l'origine "Droite", "Bas" et "Action"), de naviguer au sein de son ordinateur. Il comporte également différents menus permettant l'exécution d'actions dites secondaires (voir la documentation utilisateur pour plus d'informations sur l'utilisation du CEG).

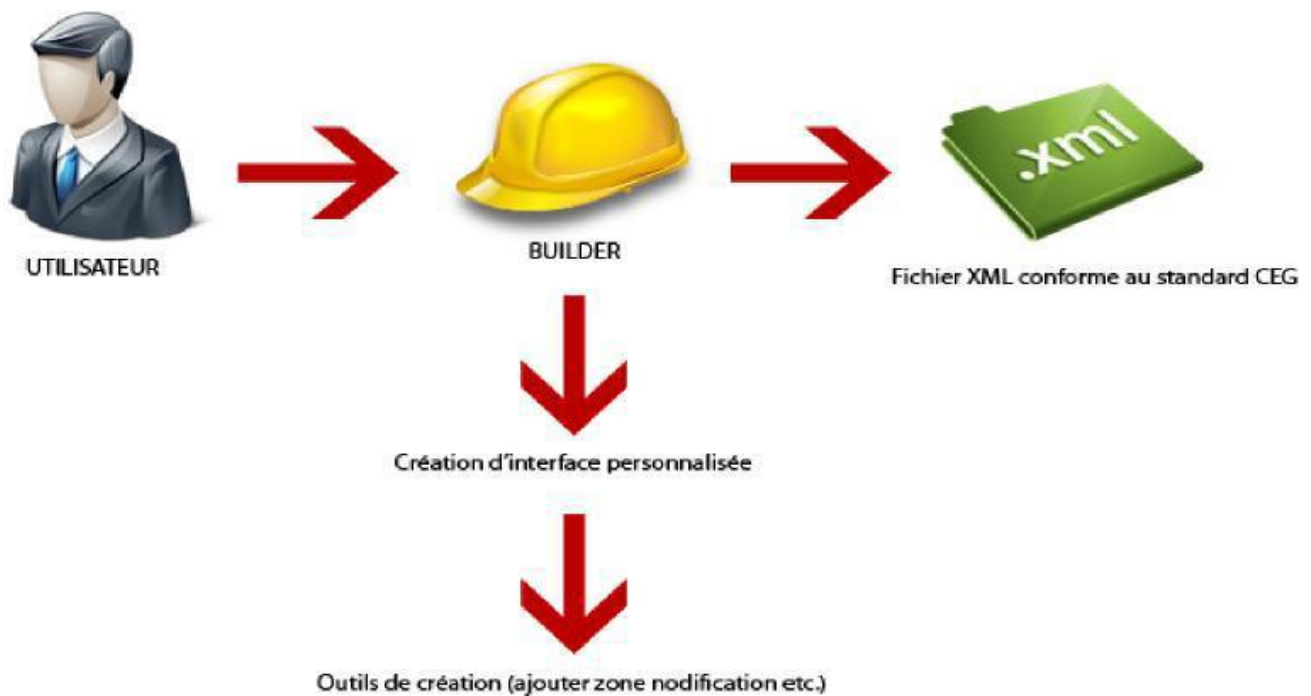
Il récupère les différentes actions envoyées par le casque EEG via le protocole VRPN (Virtual Reality Peripheral Network) géré par OpenViBE. Ce protocole est utilisable via la librairie VRPN (voir la liste des dépendances externes pour plus d'informations).

Il permet donc de faire le lien entre ce que nous pensons (interprété via le casque EEG) et ce que nous voyons (calque superposé à l'écran) en associant les événements émis par OpenViBE aux différentes actions décrites précédemment.

Ensuite, la partie eye-tracking apporte plus de flexibilité à l'utilisateur en lui permettant, s'il le désire, de remplacer une ou plusieurs actions par un suivi oculaire (mouvement ou clignement des yeux). La librairie openCV (voir la liste des dépendances externes pour plus d'informations) est utilisée pour l'acquisition et l'interprétation des images émises par la caméra ou webcam.

Enfin, dans le cadre d'utilisation du *Builder*, le calque générique est remplacé par un ensemble de zones définies au préalable dans le *Builder* et stocké sous la forme de fichier XML.

### III.II – Le *Builder*



Le *Builder*, quant à lui, se compose de deux parties, la partie sélection et la partie zone. Il permet de charger et sauvegarder sous forme de fichiers XML des ensembles de zones. Pour plus d'informations sur l'utilisation de celui-ci, veuillez-vous référer à la documentation utilisateur.

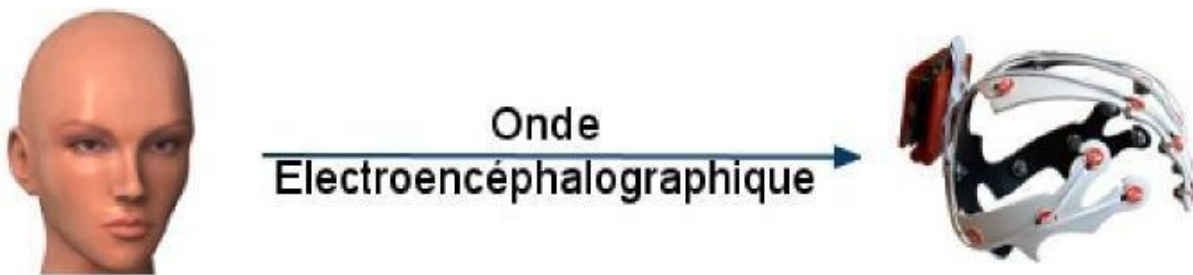
La partie sélection permet, pour une personne valide, de redimensionner, éditer, copier, couper ou coller des zones cliquables de l'application afin de remplacer le calque générique sous forme de quadrillage.

La partie zone permet d'éditer les propriétés d'une zone et d'en modifier les actions associées, comme un clic, un zoom ou toute autre action possible sur le CEG.

Le *Builder* est un programme assez simple, développé en Python et reposant sur la librairie graphique PyQt (voir la liste des dépendances externes pour plus d'informations).



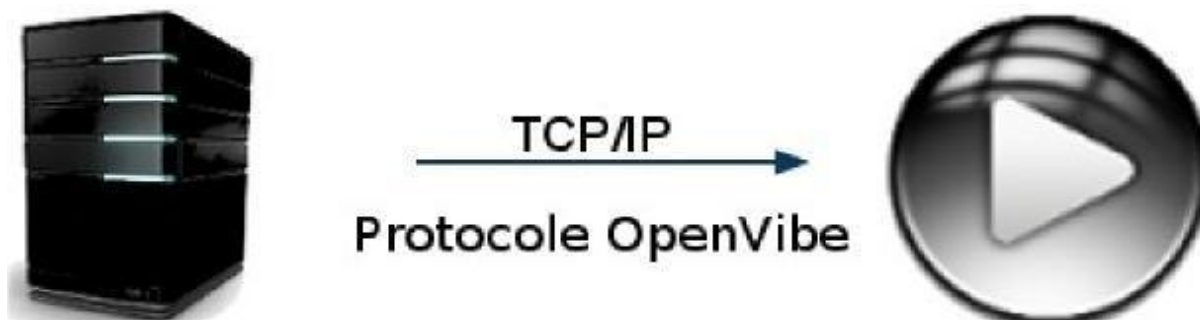
## IV – DIAGRAMME DE COMMUNICATION



Les signaux cérébraux sont récupérés sous forme d'ondes électro-encéphalographique émises par le cerveau puis convertis en signaux numériques via le casque EEG.



Le boîtier d'acquisition filtre, amplifie et convertit en numérique les signaux captés par le casque. Il les transmet ensuite au serveur d'acquisition.



Le serveur d'acquisition traduit les informations reçues pour les transmettre en TCP/IP via le protocole OpenViBE à la partie OpenViBE.

## V – CONCEPTION

### V.I – Choix des technologies

IOT repose en tout premier lieu sur l'utilisation de deux langages de développement, C++ pour le CEG et Python pour le *Builder*.

Le choix de l'utilisation du C++ pour le CEG fut originalement pris pour faciliter l'interopérabilité avec le logiciel d'acquisition, OpenViBE. Depuis, l'intégration du protocole VRPN au sein d'OpenViBE aurait permis le développement du CEG dans n'importe quel langage, mais la robustesse et la stabilité du C++ nous convainquent de conserver celui-ci. De plus, les bibliothèques externes utilisées (Qt, VRPN et openCV) sont compatibles avec un grand nombre de systèmes d'exploitation, ainsi que le code du CEG en lui-même depuis sa création. Il n'était donc d'aucun intérêt de modifier les technologies utilisés, même si cela serait parfaitement possible.

Au sujet des bibliothèques externes du CEG, chacune comporte son propre intérêt. Pour la partie graphique, Qt est utilisée car il s'agit d'une bibliothèque portable (compatible avec la plupart des systèmes d'exploitation existants), simple d'utilisation, disposant d'une très grande communauté et parfaitement adapté au C++. La bibliothèque VRPN est quant à elle utilisée car il s'agit d'une bibliothèque encore une fois portable, simple d'utilisation et permettant un dialogue avec la partie OpenViBE. Enfin, openCV est une bibliothèque permettant l'acquisition et le traitement d'images et vidéos. Cette bibliothèque est une des plus performantes dans le domaine et dispose d'une très grande communauté. En conclusion, toutes ces bibliothèques sont portables, simples d'utilisation, supportés sur un très long terme et gratuites d'utilisation pour un usage non commercial.

Le choix de l'utilisation de Python pour le *Builder* est avant tout fait afin d'assurer un développement simple et rapide. En effet, Python est un langage portable, ergonomique, simple d'utilisation et parfaitement adapté à notre demande. Couplé à la bibliothèque externe PyQt (Qt adapté au Python), il est capable de récupérer très facilement des informations complètes quant à l'environnement graphique de l'utilisateur.

En dernier point, le logiciel OpenViBE peut être considéré comme un des choix technologique le plus important au sein de notre projet. En effet, il assure l'acquisition et l'émission des signaux captés par le casque EEG et donc les pensées de l'utilisateur. Il permet une interopérabilité complète avec tous les casques EEG compatibles avec OpenViBE. Il permet également la simulation de scénarios afin de simplifier les tests et le développement sans nécessairement utiliser de casque EEG. OpenViBE est développé et supporté par l'INRIA, dispose d'une très grande communauté, d'un support à long terme, est open-source et répond parfaitement à nos attentes.

### V.II – UML / Diagramme de classes

La conception UML (diagramme de classes) fait l'objet d'une documentation externe, téléchargeable sur notre page Web rubrique "Documentation" puis "UML – Diagramme de classes".

## INFORMATIONS COMPLEMENTAIRES

### Licence

IOT est un projet open-sources sous licence GNU/GPL. Les termes complets de la licence GNU/GPL sont disponibles à l'adresse suivante : <http://www.gnu.org/licenses/gpl.html>.

### Contact

Veillez visiter notre site Web pour toute information complémentaire : <http://eip.epitech.eu/2013/iot/> .

Vous pouvez nous contacter directement via le formulaire disponible à cette adresse ou via l'adresse mail suivante : [iot\\_2013@labeip.epitech.eu](mailto:iot_2013@labeip.epitech.eu).

### Auteurs

Thibault BRONCHAIN  
Nicolas NIRLO  
Maxime NOUGAREDE  
Antoine RAPPINET  
Nicolas ROCHELLE  
Julien SADDOUR  
Jeremy VIENNE  
Charles YVANES