

Projet
iBrain

F* ME**

I AM AN OPEN API!

HACKING PARTY CO EVENEMENT 50A



coworking

50A ORGANISE LA

HACKING PARTY*

*THÉMATIQUE : OPEN DATA ET API

LE SAMEDI 19 FÉVRIER A 14.00 H

SUPER

REPARTEZ AVEC 1 CONTRAT D'EMBAUCHE !

BONUS

C'EST QUOI UNE HACKINGPARTY?

HISTOIRE/HACKING **EXPÉRIMENTATION**
BIDOUILLE **DES IDÉES SIMPLES...** **S'AMUSER**
UN HACKERSPACE NOMADE **PARTAGER LES CONNAISSANCES**
ORGANISATION AD-HOC **CRÉER DU CONCRET**
...ET DES VALEURS **VALORISER LES COMPÉTENCES ET LES IDÉES**
VISION LONG TERME **CRÉATION DE BIENS COMMUNS**
INDIVIDUS \ SOCIÉTÉS **FORMAT**
COMMUNAUTE \ LIEU
RECRUTEMENT DES PARTICIPANTS
ROADMAP **PRÉSENTATION EN 2 MINUTES**

ET... C'EST PARTI !

POUR EN ORGANISER

Trouvez un lieu et une date \ Criez une page sur HackingParty.org

QUELQUES ENVIES

Plus de lieux ouverts aux HackingParties \ Des HackingParties Synchrones

PLUS DE SUJETS NON-INFORMATIQUE

KitchenParty \ HackingCouture

Twitter @hackingparty

Facebook <http://facebook.com/hackingparty>



RÉSUMONS LA HACKINGPARTY

OPEN DATA ET API

En ce samedi pluvieux du 19 février, qui n'empêche pas la joie et la bonne humeur, nous accueillons 15 participants développeurs, chefs de projet, analyste, étudiants, explorateurs du web. Cette HackingParty s'articule autour de l'OpenData et des API.

3 OBJECTIFS :

- \\ Faire un état de l'art en regroupant les connaissances de chacun sur les API et l'Open Data, tout en ouvrant le sujet sur l'avenir.
- \\ Présenter le projet de R&D 50A.
- \\ Prendre contact avec des développeurs en vue de plusieurs embauches dans le domaine des API et de l'Open Data.

NICOLAS BERMOND, aka « Mr Orange » présente, en amont, le contexte de cette Hacking Party. après quoi Vincent Pinte rappelle, en quelques minutes, ce qu'est une information, comment la récupérer, la stocker, l'interpréter pour la traiter.

L'API IDEALE SELON CHACUN

- \\ Une API « User centric Data » traitant les données privées de santé et de business, et s'appuyant sur l'ergo design inspiré par l'application iPad FlipBoard.
- \\ Le Moodscope : en fonction de mon humeur et de mes comptes sur les réseaux sociaux, l'API m'orienterait ou me présenterait des personnes, des services, des entreprises, des loisirs...



CO-REFLEXION EVALUER LES CRITERES DE PERTINENCE

\ Une Api qui gère le fond et la forme : hyper-corrélation des données et assistance dans la prise de décisions. API capable d'orienter la prise de décision en fonction de facteurs de risques définis.

“ Un label de normalisation des API ”

\ Cloud Computing coopératif permettant de stocker et d'aider à partager les ressources.

\ Une boussole de vie pouvant gérer des restrictions et accompagner dans la prise de décisions. Cette API jouerait le rôle de parents et pourrait me responsabiliser en fonction des mes besoins, de mes envies...

\ Led's Computer : réalité augmentée +Api

Comment filtrer et afficher les informations par rapport aux critères de pertinence.

Il s'agit de formaliser en quoi consiste le déploiement d'« une bonne veille » pour traiter l'information ou, plus précisément, évaluer les critères de pertinence. Face à la surcharge d'informations (curation & visualisation) nécessité de :

- \ Bien organiser ses flux et les sélectionner
- \ Les stocker et/ou archiver

Le But étant d'organiser son savoir, de Piloter la prise de décisions, d'optimiser les relations professionnelles et de transmettre et diffuser des informations qualitatives. Il découle de cela la nécessité de normalisation des API. Tous s'accordent pour dire qu'il est nécessaire de créer un référentiel unifié.

À L'ISSUE DE CETTE RÉFLEXION, 3 AXES DE RECHERCHES ÉMERGENT :

\ Les 5 commandements d'une bonne veille

- Identifier les sources ? médias, blog, medias sociaux, data perso (personnal cloud, données subies/données choisies). Affinités géographiques, sociales, temporelles, sémantiques
- Comment émettre / partager l'information (ADN).
- Accepter d'être polluer pour revenir à l'essentiel.
- Pour faire de la bonne veille, on se doit d'émettre également.
- Phase d'apprentissage : quels outils ? quelles plateformes ?

\ Création d'un label API (inspiré du label W3C)

\ Développement de minis prototypes



NORMALISATION

Merci à Gwendal Van Hooland pour cet article !

Bien entendu, il n'est pas question de raboter les pieds des internautes pour qu'ils chaussent tous du 38, ni d'attacher la main gauche dans le dos pour obliger à cliquer (gauche) avec la bonne main (droite). Il y a 20 ans de cela, les applications étaient essentiellement monopostes, MS Office pour ne citer qu'elle, ou dans le meilleur des cas en client/serveur, sur des systèmes d'information d'entreprise totalement personnalisés et inexploitable en dehors du contexte qui leur est propre.

Internet de par son ouverture (je ne sais pas à la base qui se connecte sur mon site), de par ses possibilités étendues chaque jour (les sites plaquettes ont fait place à de réelles applications telle celle que vous utilisez actuellement pour me lire), a rendu plus que jamais nécessaire l'interopérabilité, les échanges d'informations entre systèmes, et les dialogues machine-machine afin de ne délivrer à l'utilisateur que le subtil nectar récolté sur de multiples plateformes.

Par «normalisation», on entendra le degré auquel une application est capable de se soumettre à une demande externe afin d'interagir avec elle, et de produire un contenu qu'elle-même ne maîtrise pas.

La normalisation «technique» est en place depuis les années 70 avec les protocoles TCP/IP par exemple, alors qu'au point de vue applicatif on a peut-être un peu rapidement botté en touche en se réfugiant derrière des XML ou Web Services, qui veulent tout et rien dire. Donc rien.

\ Pour schématiser les choses, une application est composée de trois blocs : Une base de données qui contient «mon univers» de façon atomique, mes grains de farine

\ Une couche applicative capable de transformer ces atomes en «matière exploitable», extraire de la farine puis la cuire pour faire du pain, par exemple

\ Une couche de restitution, capable de présenter une baguette à l'utilisateur

Soit mon application est totalement dédiée à une tâche, comme Ginette ma boulangère qui fait les meilleures baguettes de toute la région.

Soit mon application a vocation à s'insérer dans un écosystème, auquel cas j'ai besoin de savoir jusqu'où je peux collaborer avec Ginette.

Reprenons notre exemple.

- En niveau (3), je vais pouvoir insérer les photos des baguettes de Ginette sur mon site web.
- En niveau (2), en collaborant avec son applicatif, je vais diffuser un stream de sa webcam et afficher dans combien de temps les nouvelles baguettes seront disponibles, et donc en passant par un agrégateur de boulangerie avec géolocalisation, connaître la boulangerie la plus proche qui aura une baguette chaude sortant du fournil au moment où je franchirai sa porte.
- En niveau (1), en accédant «dans le dur» au stock de Ginette et à ses contraintes, je vais être capable de déterminer si elle est capable de me produire 25 croissants à l'eau de fleur d'oranger pour la prochaine Hacking Party.

L'API est l'abandon du dogme de l'interface propriétaire (le site web de Ginette) au profit de la communication avec d'autres machines (mes applicatifs) qui retraiteront l'information et se chargeront de sa diffusion et visualisation.

Le terme «Open» a été bien malmené ces dernières années, il faut l'entendre non pas comme une gratuité absolue de toute donnée ou un pillage potentiel de ce qui est ma richesse, mais bien comme une «nouvelle» forme de collaboration : si mon métier c'est la main, j'ai plus intérêt à échanger avec le pied et à proposer un corps (cf la main dans la Famille Addams).

Une norme, si petite soit-elle, devrait permettre à chaque acteur de se déclarer et de se positionner sur l'échiquier de plus en plus complexe de notre secteur.

Deux faits sont notables à ce niveau. D'une part l'hyper-spécialisation des mastodontes de l'Internet (Google ne fait pas de e-commerce versus Amazon qui est 100% e-commerce, Yahoo! devenu totalement communautaire, AOL



pure régie, FaceBook qui agite le chaudron du relationnel avant de mieux récolter les recettes publicitaires sans rien vendre, Groupon qui rappelle que la vie réelle et le offline ça existe encore, FourSquare qui évite d'appeler un pote qui déjeune à la table d'en face, etc...).

Ils sont pour la plupart massivement ouverts en termes d'API :
Google, Amazon, FS, FB...

Mais sortir des sentiers battus, c'est mon second point, conduit à prendre des risques, sans trop savoir quand on est dans le hack ou l'utilisation des données du site, et ne facilite pas les échanges commerciaux. Ginette n'a pas d'API, et je tape comme un sauvage à coups de requêtes HTTP ou Json pour avoir ce que je veux, sans aucune garantie.

En conclusion, Ricardo disait qu'en économie ouverte, un état ne pouvait être riche tout seul, sans ses voisins. Si mes voisins sont pauvres, je n'ai rien à leur vendre car ils ne peuvent pas acheter. On en est là, avec une formidable chance que nous offre Internet : si le micro-état de Ginette se conformait à quelques bonnes pratiques, elle serait de facto la reine sur l'AppStore, et je n'aurais pas à m'arracher les cheveux pendant des jours pour intégrer sa boulangerie dans l'appli de mon client.

Vive Ginette !

Merci à
Glenn Rolland
Hayat Outahar
Waine Kerr
Alexandre Stanislawski
Gwendal van Hoolan
Stéphanie Garcia
Damien Houlle
Christophe Ducamp
Pinte Deregnaucourt
Vincent Pinte
Luc
Maître Boucanier
Xavier Jaquemet
Idir Ibouchichene
Thibaut Brousse
Nicolas Bermond
pour leur participation,
et à Thibaut,
Pauline
et Fleur
pour la conception de ce fanzine

PROCHAIN RENDEZ-VOUS À LA MUTINERIE !!!