

# Site web statique dynamique !?

*Difficultés du projet :*

*Algorithmique : 2/5*

*Modélisation/Compréhension/Apprentissage de nouveaux concepts : 4/5*

*Programmation : 3 /5*

*Contact : [adrien.lebre@imt-atlantique.fr](mailto:adrien.lebre@imt-atlantique.fr)*

## Objectif :

L'objectif du projet est de réaliser un outil permettant d'ajouter dynamiquement des nouvelles entrées sur un site web construit au dessus de la solution `jekyll` (plus d'informations sur <https://jekyllrb.com>)

## Contexte, problématique :

L'équipe de recherche STACK de l'école a construit son site web sur une solution de type plain text : <http://stack.inria.fr>

L'ensemble du code source associé au site web (*i.e.*, les pages HTML ainsi que les pages markdown à partir desquelles elles sont générées) est hébergé sur une dépôt git.

Bien que cette solution soit simple au niveau infrastructure (elle ne nécessite pas la mise en place d'un serveur web avancé ni d'une base de données spécifique), elle comporte plusieurs lacunes pour les personnes en charge de faire évoluer le contenu du site web.

A titre d'exemple, il est nécessaire à chaque fois qu'un membre souhaite ajouter une nouvelle entrée dans la rubrique « news » de régénérer l'ensemble des pages webs à partir d'un terminal (*i.e.*, en ligne de commande). En plus d'être fastidieux, ce processus entraîne souvent des erreurs d'inattention dans la reconstruction des pages.

```
dhcp-10-44-192-157:~$ cat _posts/2018-03-30-Project-Accepted-ONCOSHARE.markdown
---
layout: post
title: "ONCOSHARE project (ONCOlogy big data SHaring for Research) accepted"
date: 2018-03-30
categories: projects
---

*By Mario Südholt*

The ONCOSHARE project (ONCOlogy big data SHaring for Research) will demonstrate, through a multidisciplinary cooperation within the Western CANCEROPole network, the feasibility and the added value of a Cancer Patient Centered Information Common for in-silico research. The STACK team will work on challenges to the security and the privacy of user data in this context.

This project is financed by three French regions and will start in the 3rd quarter of 2018.
```

The screenshot shows the STACK website with a teal header. The header contains the title "STACK" and the subtitle "Software — Stack — for Massively Geo-Distributed Infrastructures". Below the header are logos for IMT Atlantique, Inria, and LS2N. A navigation bar includes links for News, Jobs, Members, Publications, Projects, Platforms & Software, and Contact. The main content area features a news entry titled "ONCOSHARE project (ONCOlogy big data SHaring for Research) accepted - Mar 30, 2018" by Mario Südholt. The text of the entry matches the markdown code shown in the previous block.

**Exemple d'une nouvelle entrée en ligne de commande et de son rendu coté web**

Le but du projet est de fournir un utilitaire qui va permettre :

- de construire automatiquement la page `markdown` associée à une nouvelle entrée en guidant la personne dans la saisie des diverses informations qu'elle doit renseigner ;
- de recompiler de manière automatique les pages `jekyll` une fois la saisie terminée ;
- d'afficher le résultat dans un navigateur ;
- et de pousser vers le dépôt git officiel le nouveau code généré une fois celui-ci validé.

## Phases et Livrables :

La première phase va consister au développement d'un utilitaire en mode ligne de commande. L'utilisateur devra être en mesure de saisir une nouvelle entrée en indiquant le titre, la date et le contenu qu'il souhaite faire apparaître sur le site. A noter qu'il devra également être possible de définir une ou plusieurs catégories associées à cette nouvelle entrée. Une fois les informations saisies, l'utilitaire devra faire le nécessaire pour publier cette nouvelle entrée sur le dépôt git et donc le site web officiel.

Le travail de cette première phase pourra être découpé en plusieurs parties :

- la génération du fichier `markdown` à partir des éléments récoltés ;
- la compilation des pages `HTML` ;
- l'ouverture dans un navigateur afin de valider les changements ;
- la possibilité de modifier ou d'annuler l'opération si le rendu n'est pas celui escompté ;
- la synchronisation vers le dépôt git lorsque la saisie est validée.

La seconde phase consistera en la mise en place d'une couche graphique (i.e., GUI).

En fonction des progrès réalisés, il sera possible d'ajouter des fonctionnalités afin d'alimenter sur la même base d'autres parties du site web (annonces d'offres d'emplois, ajout, suppression de membres de l'équipe, ...).

Le code fourni devra être commenté et mis à disposition sur un dépôt github. Ce dernier devra inclure un fichier `readme.md` indiquant pour chacune des phases comment installer et lancer l'utilitaire.

## Environnement technologique :

L'utilitaire devra être codé en java et devra être indépendant des systèmes d'exploitation (c'est à dire qu'il devra être capable de fonctionner sous des environnements Windows, MAC OS ou Linux).

# Projet Image Médicale

Kimi est un projet d'assistance à l'apprentissage du diagnostic en imagerie endoscopique. Il a été conçu au LS2N et est utilisé au CHU de Nantes. Les internes peuvent y soumettre leurs images médicales pour demander conseil aux experts, ces images ne sont pas anonymisées – elles contiennent des données patient (par exemple le numéro sur l'image ci-dessous).

**Réalisation demandée :** Développer un éditeur d'image simple qui permette de masquer interactivement les zones de texte d'une image médicale.

**Etape 1** – développer un client interactif Angular2 (v6.1) / typescript.

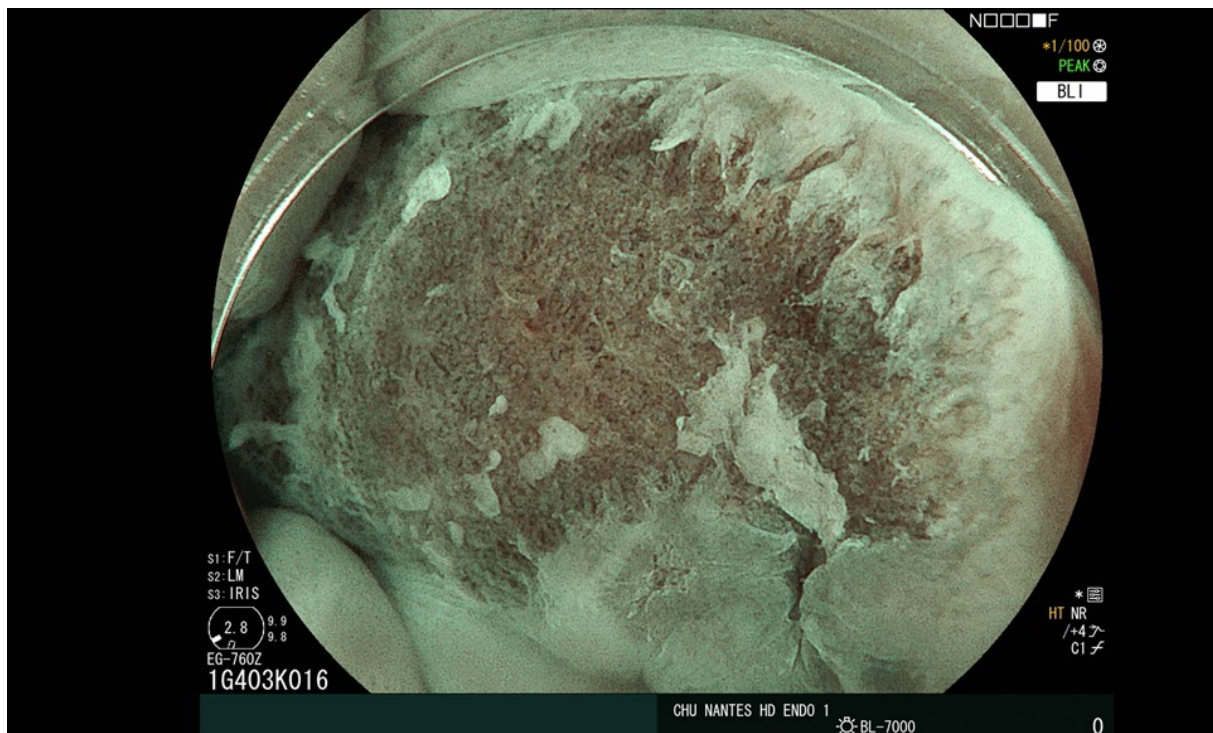


Figure 1 : image endoscopique avec numéro de patient.

**Etape 2** – proposer un algorithme de détection (semi) automatique des zones de texte.

Les images sont stockées sur firebase – le projet peut se faire simplement avec des images servies à partir d'un serveur web statique simple.

**Tuteur :** [Cedric.Dumas@imt-atlantique.fr](mailto:Cedric.Dumas@imt-atlantique.fr) bureau C111

# Projet Visualisation Firebase

Kimi est un projet d'assistance à l'apprentissage du diagnostic en imagerie endoscopique. Il a été conçu au LS2N et est utilisé au CHU de Nantes. Les internes peuvent y soumettre leurs images médicales pour demander conseil aux experts. Les images, les diagnostics proposés par les médecins et leurs commentaires, sont stockés sur firebase. Le front est une application Typescript / Angular2 (v6.x).

Ces commentaires forment des discussions sur chaque image, avec différents diagnostics proposés pour une image donnée.

**Réalisation demandée** : Développer un outil autonome (indépendamment de Kimi, pas de code à reprendre) de visualisation de l'activité des médecins le long des discussions : nombres d'avis proposés, qualité des diagnostics.

**Etape 1** – développer des services Angular qui permettent d'extraire les données de firebase.

**Etape 2** – concevoir un tableau de bord visuel (avec des composants Angular) pour afficher ces différents indicateurs.

**Tuteur** : [Cedric.Dumas@imt-atlantique.fr](mailto:Cedric.Dumas@imt-atlantique.fr) bureau C111

# Analyses biomédicales dans le nuage

Mario Südholt, DAPI, IMT Atlantique  
(<http://sudholt.eu>)

## 1. Introduction

Les analyses biomédicales ont pour objectif d'inspecter les cellules ou génomes humains afin de mieux comprendre le fonctionnement du corps humain.

Les analyses génomiques notamment servent souvent à détecter des maladies. L'approche d'analyse la plus largement adoptée dans ce contexte consiste à la comparaison statistique de deux ensembles de génomes dont un contient les génotypes appartenant à des individus portant la maladie tandis que le second ensemble contient des génotypes de témoins (individus sains).

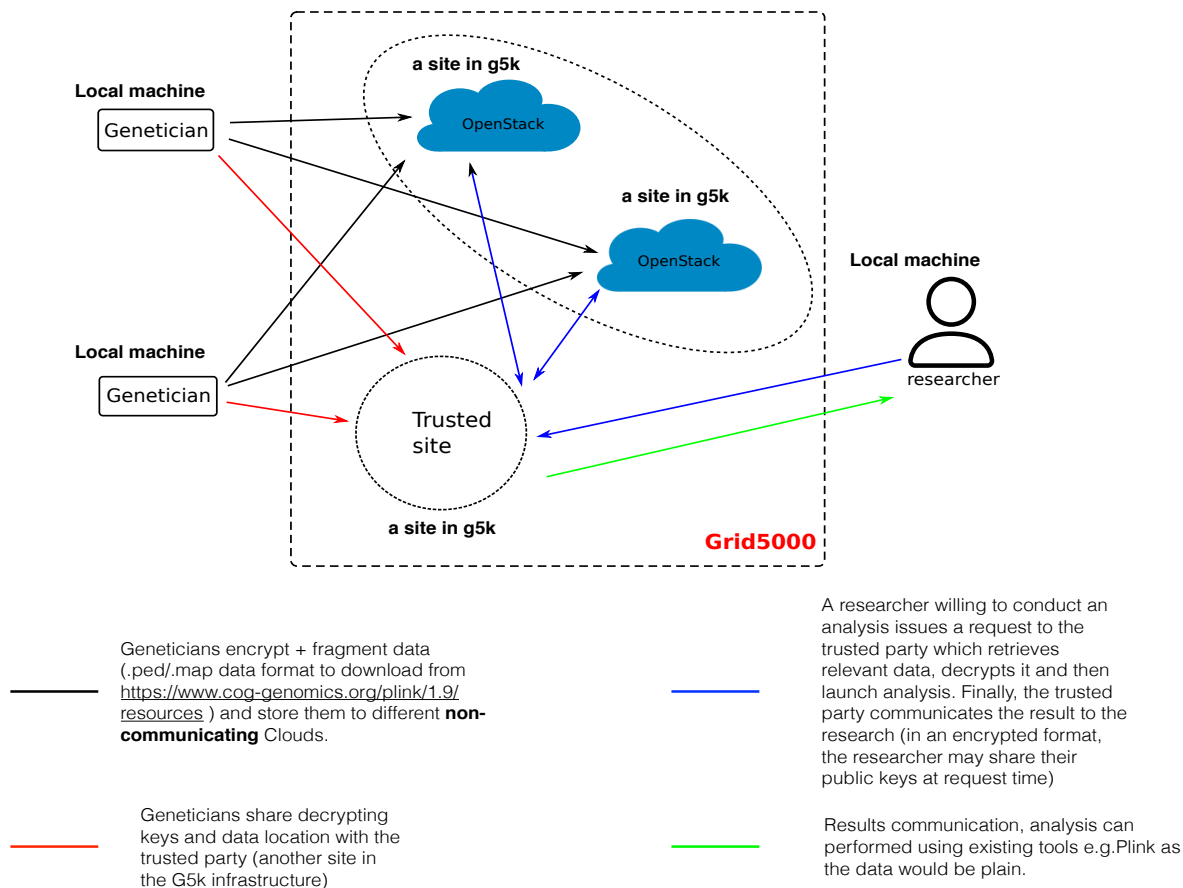
Il s'avère que les généticiens ont besoin de partager leurs données génétiques afin d'avoir des ensembles suffisamment larges pour une comparaison statistique significative. Dans ce sens, l'état de l'art nous propose de partager les données génétiques dans le nuage, or, des problèmes de transferts sécurisés de masses de données et des soucis concernant la vie privée des individus participants émergent avec cette solution.

## 2. Objectifs du projet

Dans ce projet, l'objectif sera d'implémenter une solution de partage de données génétiques qui sécurise les données et respecte la vie privée dans le nuage. Globalement, l'objectif du projet est l'implémentation en Java d'architecture de partage de données génomiques, pour un exemple voir l'illustration en bas de cet énoncé, en utilisant plusieurs techniques de sécurité :

- **fragmentation** : consiste en la partition d'une base de données ;
- **chiffrement** : est appliqué essentiellement au génotype comme technique d'anonymisation ;
- **utilisation d'une partie de confiance** : les analyses seront faites au sein d'une partie (particulièrement sécurisée) de confiance.

Des implémentations non distribuées de telles architectures et des opérations sont disponibles. L'implémentation distribuée se fera sur Grid 5000, un nuage public.



Septembre 2018 - Rémi Douence  
FIL A1 ACDC  
Le programme dont vous êtes le héros

Les livres dont vous êtes le héros sont des livres interactifs qui permettent au lecteur de choisir comment l'histoire se déroule.

The War-Torn Kingdom est le premier tome de la série des Fabled Lands.

Partie 1 :

Vous programmez en Java The War-Torn kingdom en mode texte (une version pdf et une version xml du livre vous sont fournies).

- Votre programme gère tout ce qui peut l'être et dispense le joueur de lancer des dés, noter des valeurs, voir des numéros de paragraphes, etc.
- Votre programme doit proposer des hiérarchie de classes Java réutilisables qui facilitent la programmation de livres.
- Votre programme permet de sauvegarder/charger une partie en cours.
- Votre programme offre une fonctionnalité undo/redo non bornée.

Partie 2 :

Vous récupérez le programme d'un autre élève et vous l'étendez :

- avec un test qui vérifie la cohérence (tous les paragraphes sont bien présents et accessibles, il n'y a pas de paragraphe d'indirection).
- avec une fonctionnalité permet de jouer avec un browser et sans Java (attention à bien faire *réapparaitre* ce qui doit être géré par le joueur (lancés de dés, notes manuscrites, etc.) et à soigner le visuel, vous avez dit « css » ? ) :
  - une première version génère plein de documents (une page html pour chaque paragraphe, avec hyperliens mais sans numéro de paragraphe).
  - une seconde version qui génère un document unique imprimable (une unique page html pour tout un livre, avec des hyperliens et des numéros de paragraphes). On s'assurera que deux paragraphes liés sont, dans la mesure du possible, ni trop proches ni trop éloignés.
- avec fonctionnalité inverse qui partant de pages html génère le programme Java correspondant.
- avec une fonctionnalité qui dessine le graphe correspondant (vous pouvez utiliser des bibliothèques ou des outils extérieurs).



# Jeux de Lettres



## Contexte et objectifs

Vous connaissez tous le jeu du Scrabble. Vous connaissez peut-être le fundox (<http://fundox.free.fr/> précédemment appelé wordox) qui lui aussi se joue au tour par tour sur un plateau avec des lettres mais qui possède également pas mal de petites différences (les cases du plateau, le comptage des points, l'appartenance temporaire des lettres à un joueur, le râtelier partagé, le vidage du plateau,...).

L'objectif de ce projet est de réaliser une implémentation de ces deux jeux partageant un maximum de code. Il ne s'agit pas de faire deux programmes indépendants, mais un programme permettant de jouer aux deux jeux. Le but est donc de cerner tout ce qu'il y a de commun entre ces jeux (un plateau, des dalles, un/des râteliers, vérifier un mot dans le dico, déposer des lettres, ...) et qui aboutira à du code partagé. Mais aussi d'étudier les spécificités qui conduiront à du code utile uniquement à l'un des deux jeux.

Il n'y a pas la nécessité d'implémenter un chronomètre : les joueurs auront tout le temps qu'ils souhaitent pour réfléchir.

## Réalisation

Un dictionnaire vous sera remis sous la forme d'un fichier texte ainsi que des images qui pourront vous servir pour l'interface graphique.

Première phase : implémentation d'un programme permettant de jouer au Scrabble et à Fundox **avec une interface purement textuelle** (uniquement via des affichages et saisies dans la console). Il devra être possible de jouer à plusieurs joueurs humains.

Deuxième phase : Ajout, à la réalisation qui vous sera remise en début de deuxième phase, d'une **interface graphique**. L'accent devra être mis sur la jouabilité (aussi agréable à jouer que possible). Vous devrez de plus ajouter la fonctionnalité d'enregistrer/charger une partie afin de la reprendre plus tard.

