



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Facultat d'Informàtica de Barcelona

Sistema per l'emissió, validació i signatura electrònica de consentiments informatats per aplicació clínica

GEP #5: Plec de condicions (Bloc Especialitats)

Autor:

MARC JUBERÓ SILVA

Director:

OSCAR FLORES GURI

Ponent:

TONI CORTÉS ROSSELLÓ

24 d'octubre de 2016

Índex

| | | |
|----------|-------------------------------------|----------|
| 1 | Especificació de requisits | 3 |
| 2 | Disseny arquitectònic | 4 |
| 3 | Implementació | 4 |
| 3.1 | Entorn de desenvolupament | 4 |
| 3.2 | Gestió de projecte | 5 |
| 4 | Gestió de riscos | 5 |
| 5 | Bibliografia | 6 |

1 Especificació de requisits

El consentiment informat és un procediment que, dins de l'àmbit mèdic, és pràcticament obligatori i serveix per a que els professionals mèdics informin als pacients d'un tractament i/o estudi sobre diferents aspectes del relacionats amb el procediment; aquesta informació és d'extrema rellevància i cal que el pacient en sigui conscient. Els temes que es tracten en el procés son tan diversos com riscos existents, alternatives, possibles millores resultants, etc.

El procés de consentiment informat, culmina amb l'emissió d'un document per part del professional mèdic i la posterior signatura d'aquest per part del pacient.

Amb aquesta signatura es posa en manifest que el pacient ha rebut la informació necessària per a tenir coneixement del tractament/estudi, que aquest mateix ha entès la informació que se li ha facilitat i que, de forma totalment voluntària i amb ple ús de les seves facultats, hi està d'acord.

Tal i com es pot suposar de les línies anteriors, el procés del consentiment informat és quelcom totalment "manual" que requereix de la interacció dos actors.

Per una banda un emissor que emet un missatge; en aquest cas, el professional mèdic i, per l'altre, un receptor, que en aquest cas és pacient que se sotmet a un tractament o participa en un estudi.

A dia d'avui, en un entorn mèdic convencional, aquest és el procediment que se segueix. No obstant, amb l'aparició de les noves tecnologies i, principalment amb l'aparició d'internet era qüestió de temps que aparegués la necessitat de dotar al procediment de consentiment informat de capacitat telemàtica sense perdre validesa legal pel simple fet de que el procediment no compta amb una signatura firma manuscrita.

És en aquest punt que apareix *Made of Genes*[1], una empresa de genòmica personalitzada que ofereix, per una banda desfer les dades genòmiques dels seus clients al núvol d'una forma segura, garantint que les dades són sempre de l'usuari i, per altre banda, una plataforma web a mode de *marketplace* on els usuaris poden comprar serveis basats en les seves dades genòmiques.

Aquestes dades desades al núvol tenen la categoria de dades clíniques i, per tant, la llei obliga l'ús de del consentiment informat; en aquest cas concret, per a que els usuaris que compren els serveis disponibles a la plataforma permetin que, aquells tercers que els ofereixen, puguin accedir a les dades genòmiques desades.

Queda patent doncs, la necessitat de dotar al consentiment informat d'una nova dimensió, per tal de que els usuaris de la plataforma *Made of Genes*[1] puguin fer-ne ús sense necessitat de visitar al professional mèdic responsable de cada servei.

És fruit d'aquesta necessitat que apareix el context en el qual es realitza aquest TFG, la necessitat de desenvolupar un mòdul dins de la plataforma que permeti, per una banda que els usuaris siguin capaços de firmar els documents i permetre l'accés dels professionals mèdics a les seves dades genòmiques i, per l'altre, dotar de validesa legal i jurídica al procediment telemàtic de firma del consentiment informat.

2 Disseny arquitectònic

Tot i formar part d'una plataforma, s'ha de pensar en aquest TFG com un mòdul que podria funcionar de forma independent.

A nivell d'arquitectura general del projecte, es comparteix l'arquitectura de la plataforma *Made of Genes*:

- un client (navegador web) desenvolupat amb *AngularJS*[2] un *framework* javascript mantingut per Google que aporta els béns de l'arquitectura Model-Vista-Controlador (MVC[4]) al món web.
- una API[5] Rest[6] desenvolupada amb *Symfony*[3], un dels frameworks PHP més emprats, que serveix les peticions del client amb la informació necessària en cada cas.

Per altre banda, i a un nivell més particular, el TFG en qüestió fa ús d'unes determinades tecnologies que permeten dotar de la validesa legal desitjada al procediment del consentiment informat.

Aquestes tecnologies són:

- Per un costat, per tal de poder firmar els documents es fa ús del que s'en diu contrassenya d'un sol ús, o com es diu en anglès, *One Time Password (OTP)* basat en l'especificació del paper RFC6238[7].
En aquest document s'especifica un algorisme basat en una marca de temps (o *timestamp*) que, mitjançant aquesta marca i una clau personal de l'usuari es genera un codi únic per a cada ocasió.

D'aquesta forma s'assegura que per a un usuari i un moment determinat, hi haurà un sol codi, assegurant així la identitat de l'usuari.
- Per altre banda es fa ús del que s'anomena *Blockchain*[8], una tecnologia bastant actual coneguda principalment per a ser la tecnologia sobre la que se sustenta la moneda virtual Bitcoin i que, gràcies a la seva arquitectura distribuïda, ens permet assegurar el no repudi del document del consentiment informat.

3 Implementació

3.1 Entorn de desenvolupament

Per dur a terme el desenvolupament s'ha fet ús, principalment, dels IDE[9] de l'empresa JetBrains[10]:

- Webstorm, per al desenvolupament de la part client (frontend)
- PHPStorm, IDE per a PHP per al desenvolupament de la part servidora

Un IDE, a diferència d'un editor de text normal que es podrien entendre com a eines de propòsit general, és una eina preparada per a treballar amb una tecnologia concreta, que integra tot un seguit de complements o eines addicionals que busquen agilitzar el desenvolupament amb una determinada tecnologia.

En el cas concret dels IDE emprats, Webstorm està pensat per treballar amb javascript, mentre que PHPStorm, tal i com es pot entreveure pel nom, està pensat per treballar amb

PHP.

Tot les particularitats de cada un dels IDE, molts cops permeten estendre les seves funcionalitats cap a altres llenguatges i/o frameworks amb l'ús de complements.

3.2 Gestió de projecte

Per tal de gestionar les tasques i les diferents iteracions marcades per l'ús de la metodologia àgil *Scrum*[11] es fa ús de la suite d'aplicacions d'Atlassian[12] que integra tot un seguit d'eines que permeten la gestió de projectes:

- **Jira** per a la gestió de les tasques i *sprints*.
- *Confluence* per a la documentació general del projecte.
- *Bitbucket* com a repositori de control de versions.

Per al versionat de codi, es fa servir Git[13] com a motor de control de versions i se segueix la metodologia GitFlow[14] per tal de mantenir el repositori organitzat d'una forma coherent.

4 Gestió de riscos

Al començament del projecte no es poden determinar els riscos existents, per això es fa ús de *Scrum*[11] que ens permet, mitjançant les iteracions, no només organitzar les tasques per a facilitar el desenvolupament, sinó veure si hi ha cap desviació dins dels temps establerts en el projecte.

Un cop detectada una desviació, donat que es tracta d'un projecte on el temps de durada està totalment fitat, es decideix juntament amb la resta de l'equip de desenvolupament de la plataforma, com actuar per tal de pal·liar la desviació detectada.

Les possibles vies d'actuació són les següents:

- Ampliació de l'equip de desenvolupament
- Aplazar la data final de lliurament
- Reformular els requisits del projecte

De totes les opcions anteriors, la que es durà a terme en cas d'aver-hi desviacions dins dels plaços, serà la de reformular els requisits inicials del projecte, per tal d'intentar trobar aquelles característiques que no siguin essencials per al bon funcionament del projecte, assegurant d'aquesta forma que un cop assolida la data de finalització del projecte, s'haurà obtingut un mínim producte viable.

5 Bibliografía

- [1] Made of Genes: <https://madeofgenes.com/>
- [2] AngularJS: <https://angularjs.org/>
- [3] Symfony: <https://symfony.com/>
- [4] MVC: https://developer.mozilla.org/en-US/Apps/Fundamentals/Modern_web_app_architecture/MVC_architecture
- [5] API: <http://www.webopedia.com/TERM/A/API.html>
- [6] Rest: <http://www.restapitutorial.com/lessons/whatisrest.html>
- [7] OTP: <https://tools.ietf.org/html/rfc6238>
- [8] Blockchain <https://www.blockchain.com/>
- [9] IDE: https://en.wikipedia.org/wiki/Integrated_development_environment
- [10] JetBrains: <https://www.jetbrains.com/>
- [11] Scrum: <http://scrummethodology.com/>
- [12] Atlassian: <https://www.atlassian.com/>
- [13] Git: <https://git-scm.com/>
- [14] GitFlow: <http://nvie.com/posts/a-successful-git-branching-model/>