

Planteamiento de la implementación del módulo.

Objetivos: Definir el cómo se implementará el módulo de consultas federadas en Apache Marmotta.

Resultados esperados: Documento que mencione cómo se hará la implementación del módulo.

Con base en la metodología se usará para llevar a cabo la implementación del módulo de Apache Marmotta, el presente documento aborda cómo se llevará a cabo su implementación.

La metodología de desarrollo de software a usar es *Extreme Programming (XP)*. Los roles dentro del equipo son:

- Clientes
 - Protocolo SPARQL.
 - Protocolo GeoSPARQL.
 - Objetivos y alcances del documento entregado en metodología de la investigación.
- Programador
 - Alumno Páez Ortega Oswaldo Emmanuel.
- Testers
 - 2 desarrolladores de Apache Marmotta contactados por el mánager Luis.
- Coach
 - Asesora Enríquez Ortiz Cyntia Eugenia.
- Manager
 - Asesor Vilches Blázquez Luis.

Cabe decir que, si bien los clientes no son personas, esos documentos serán los que definan los requerimientos y restricciones del proyecto.

Ya definido los roles que cada persona desempeñará dentro del equipo, a continuación, se describe como es que se realizará la implementación del módulo de consultas en Apache Marmotta.

XP, al igual que el de espiral, es una metodología iterativa lo que significa que las fases de desarrollo, descritas a continuación, se repetirán hasta que los requerimientos y/o restricciones del cliente sean atendidas y satisfechas.

- Planeación
 - Requerimientos
 - Tecnologías por usar
 - Java
 - Arquitectura SOA
 - Maven
 - TomCat
 - HTML
 - JavaScript
 - RDF/XML
 - *Triple store*

- Arquitectura *REST*
 - JSON
- Protocolos
 - SPARQL 1.1
 - GeoSPARQL
 - HTTP
- Historias
 - Las historias se llevarán a cabo con la ayuda de los desarrolladores de Marmotta y basándose en los protocolos considerados en los requerimientos usando una plataforma para su registro y a su vez establecer su prioridad de realización. La plataforma por usar para el registro de historias será Trello¹ ya que en ella se pueden registrar las historias, establecer su prioridad y mantener a los miembros del equipo en contacto para que estén enterados de los objetivos pendientes o cumplidos.
- Diseño
 - Las tarjetas Clase-Responsabilidad-Colaborador (CRC) serán realizadas mediante diagramas de caso de uso del módulo y del servicio Web
 - Aunque los diagramas de caso de uso serán entregados para el proyecto terminal 1, esto no implica que dichos diagramas puedan ser optimizados con el desarrollo del software en el proyecto terminal 2.
- Codificación
 - Con el objetivo comprobar el trabajo a lo largo de proyecto terminal 2 y de probar que los desarrollos funcionen, se llevarán a cabo pruebas unitarias de cada historia para verificar de manera independiente las clases, funciones y algoritmos antes de ser incorporadas al código fuente completo de Apache Marmotta.
 - Cada prueba unitaria que se realice, será enviada a los *testers* para que determinen si la prueba unitaria funcionó correctamente. En caso de no ser aprobada por los *testers*, la prueba unitaria se tendrá que corregir, hasta que los *testers* la consideren adecuada. En caso de ser aprobada, se continuará con las historias restantes para la actual fase de desarrollo. Esta práctica propiciará que el trabajo realizado por el desarrollador sea bajo las buenas prácticas de desarrollo de software [1] [2] y que se apeguen a los protocolos SPARQL y GeoSPARQL.
 - Cuando los *testers* estén de acuerdo con el funcionamiento correcto de todas las pruebas unitarias realizadas en la fase del proyecto, éstas se implementarán en el código fuente de Apache Marmotta.
- Pruebas
 - En esta etapa se contará con el apoyo de los desarrolladores de Marmotta, ya que ellos al tener mayor experiencia, podrán verificar exhaustivamente las pruebas unitarias enviadas por el desarrollador para garantizar que la corrección y la implementación de las mismas se hicieron de manera correcta.

¹ <https://trello.com/>

- Los *testers* se basarán completamente en los requerimientos del cliente (SPARQL 1.1, GeoSPARQL y el protocolo entregado en metodología de la investigación) para contextualizar las pruebas que llevarán a cabo.
- Cada prueba unitaria que ellos reciban, la evaluarán para comunicar sus opiniones, correcciones o aprobaciones. Esto determinará si el desarrollador debe de cambiar o corregir la prueba unitaria, o en dado caso, avanzar a la siguiente.

Cada vez que se finalicen las historias en cada fase del proyecto, el *mánager* y la *couch* serán los encargados de decidir si es necesario hacer un cambio en cuanto a la velocidad y dirección en la que va el proyecto para que en la siguiente fase sean implementados los cambios que ellos consideren pertinentes con el fin de entregar el producto a tiempo.

Bibliografía

- [1] Harvard University, «Java Coding Standards,» Octubre 2006. [En línea]. Available: https://sites.fas.harvard.edu/~cscie22/problem_sets/coding_standards.shtml. [Último acceso: 12 Octubre 2019].
- [2] Google, «Google Java Style Guide,» 28 Septiembre 2019. [En línea]. Available: <https://google.github.io/styleguide/javaguide.html>. [Último acceso: 12 Octubre 2019].