**Proposal Title:** Document Based NoSQL Support for WSO2 Identity Server Database

**Name:** Asantha Thilina

**Email:** asanthathilina@gmail.com

**College:** Sri Lanka Institute Of Information Technology

**Field of Study:** Information technology specialized in software Engineering

**Interests:** Big Data Handling, Algorithm Designing, Artificial Intelligence, Information Security

**LinkedIn:** https://lk.linkedin.com/in/asantha-thilina-344687b0

**Education and Previous work related to the Project**

I am a student of Sri Lanka Institute Of Information Technology. I expected to get my B.Sc. (Information technology) (Hons) degree on February 2017. In my 3 years of University study up to now I have worked with Java, C++, C#, PHP, Python and JavaScript programming languages and further gained knowledge and experience much more with programming in Java, C# and PHP. In my University I have followed courses like Software Engineering, Data Base Management System, Artificial Intelligence, Data Analysis and Algorithm, Data Communication and Computer Networks. I think that knowledge will be very helpful when working on this project and also currently I following this online course[1] which is conducted by mongoDB university. I have worked with Git and Maven, and I think that experience will be very helpful when working with this project.

**Background of the project**

WSO2 Identity Server is a Centralized User Identity Management server which can connect and manage different applications in multiple platforms and domains [2]. One of its main component is User Store Manager.

Currently WSO2 Identity Server supports LDAP, Microsoft Active Directory and Relational Databases as User Stores. With the fast growth of enterprise level NoSQL databases, it is important to support NoSQL user stores efficiently to improve the value that WSO2 Identity Server can deliver to enterprises in terms of user data management.

**Planned Approach**

WSO2 Identity server currently available for RDBMS through JDBC User Store. In order to get use to available database communication methods in WSO2 Identity Server, I will implement Custom User Store Manager for MySQL database by extending the available Abstract User Store Manager.

First I will be design NoSQL Database Architecture Diagram that supports WSO2 Identity Server. In the phase of designing the architecture diagram I will be focus on following areas.

* Efficient Data Read/Write.
* Non Redundancy data mapping in database.
* Document size considerations.
* Complexity of data structures.
* Data Consistency.

For the efficient data read/Write I will be create suitable Indexes inside the DB such as Single Field Indexes, Compound Indexes, Multikey Indexes, etc. [3]. At this stage I cannot predict what are the optimal indexes to be create because it depends on the DB Structure which will be going to implement.

In order to manage Document size efficiently, I will be use the MMAPv1 storage engine [4].

After Designing the DB Structure I will implement MongoDBUserStoreManager that supports storing/retrieving data from NoSQL databases. While developing this I will be consider following areas.

* Efficient data Retrieval.
* Data Integrity and Security.

Use a proper design pattern to handle multiple client request efficiently.

For improve the security of the MongoDBUserStoreManager I follow the AFAIK encryption mechanism which is currently used in WSO2 Identity Server.

**Estimated Schedule**

March 27 - April 27

Getting Familiar with how WSO2 Identity Server works and it’s functionalities in deep.

Understanding the design and source code of WSO2 Identity Server.

April 28 - May 25

 April 28 – April 30

Getting familiar with available database schema structure for JDBC user store manager.

May 01 – May 06

Getting familiar with advanced MongoDB data modeling techniques.

May 06 – May 25

Mapping Available JDBC user store table structure to MongoDB document structure.

Design and document MongoDB database architecture diagram.

**Submit database architecture diagram.**

**Publish architecture diagram in blog.**

May 25 - July 24

 May 25 – July 13

Implementation of the MongoDBUserStoreManager.

July 13 – July 24

Create test cases for unit tests in MongoDBUserStoreManager.

Document test cases and test results.

July 24

**Submit MongoDBUserStoreManager module extension and source code.**

**Submit unit test case document for MongoDBUserStoreManager.**

 **Submit test scripts for unit testing.**

July 25 - August 05

Functionality testing.

Performance testing and documenting test results.

August 05

**Submit test document for functionality testing and performance testing.**

**Submit test scripts for functionality and performance testing.**

August 05 – August 13

Do some researches on how to integrate other NoSQL databases (such as CouchDB, RavenDB, Neo4j,etc.) with WSO2 Identity Server

August 13 - August 21

Documenting the implementation details of the MongoDBUserStoreManager and the MongoDB database.

Creating blogs for the details of the project.

 **Publish article along with source code by describing my approach.**

 **Submit a developer guide for MongoDB user store manager.**

**Deliverables**

* Database Architecture Diagrams
* MongoDB User Store Manager
* Automated Tests
* Developer documentation
* Articles/blog posts on using the MongoDBUserStoreManager

**References**

**[1]** [**https://university.mongodb.com/courses/M101P/about**](https://university.mongodb.com/courses/M101P/about)

**[2]** [**http://wso2.com/products/identity-server/**](http://wso2.com/products/identity-server/)

**[3]** [**https://docs.mongodb.org/manual/core/indexes/**](https://docs.mongodb.org/manual/core/indexes/)

**[4]**[**https://docs.mongodb.org/manual/core/data-model-operations/#data-model-document-growth**](https://docs.mongodb.org/manual/core/data-model-operations/#data-model-document-growth)