OpenMRS ID Platform Improvements

Primary Mentor	Elliott Williams
Backup Mentor	Hannah Downey
Assigned to	Weihua Cheung

Abstract

Elliott built the OpenMRS ID Dashboard as a student in 2011. The Dashboard is a lightweight web application (~4000 SLOC) that facilitates user-account creation and profile management in the OpenMRS Community. We're looking to design better ways for users to "create a profile" that allows them to tie lots of information to their OpenMRS ID—such as who they are, what their needs are, what they're interested in. We will be evolving the OpenMRS ID system to be capable of holding this kind of information.

Currently, the OpenMRS ID's data model is built around an LDAP user object (we use OpenLDAP under the hood to store the basic user profile data). Since we want to store much more versatile data than LDAP allows, the Dashboard will need to be decoupled from its LDAP core, and a new data model that can store and manipulate all kinds of data on the user will be implemented. Building and implementing this new data model is the first objective of the project.

OpenMRS ID also needs better connectivity with other services that hold OpenMRS contributor data. We want to allow users to get data into their OpenMRS Profile from other networks—GitHub, Twitter, etc. We also need to interact with the data *outside* of the Dashboard—from other applications like Desk.com or HingX. To handle this, a REST interface should be implemented that has full access to the new data model.

The Dashboard is written in Node.js. **Knowledge of server-side JavaScript, web services (REST and OAuth), and LDAP user-management is recommended.** It's okay if you're new to some of these areas, but please have some knowledge of Node and Express, LDAP infrastructure, and web development best-practices by the start date of **May 19**.

Objectives

- Re-write OpenMRS ID's data model to a generic data store, probably based on MongoDB. This data model will need to sync with the current OpenLDAP server or needs to implement an LDAP protocol server using ldapjs.
- Build a REST and OAuth-backed interface to the new data model.
- Improve the signup and profile-editing interface at https://id.openmrs.org with some of the new OpenMRS ID functionality.

Extra Credit

- *Use* the REST API you built to import data from OpenMRS ID to Desk.com, our CRM/Helpdesk service.

Design Ideas

- I'm leaning toward using Mongoose on top of MongoDB as the platform behind the new data model. We'll want to create a schema that contains all the user attributes we store on LDAP currently, and that can be extended to store anything else.
  - In addition to a user-centric data model, we should move some of the data that's currently stored in MySQL via Sequelize over. These data include email verifications, google groups subscriptions, and navbar links.
- Once the Mongoose ODM is in place (and structured in a models/ directory or something similar), we can build an LDAP server using ldapjs that draws from this data model directly. ldapjs provides an Express-like interface for responding to LDAP requests. The goal will be to replace our current OpenLDAP server with this LDAP interface on top of the new data model.
- Finally, I'm interested in using Restify to serve the data model as a REST interface. Restify can be app.use()'d into the Express server and served from an /api uri on the server.
- As it currently stands, the application is a little unstructured—there's not so much MVC / separation of concerns. I'd love to better organize the application as a part of the development process. For some inspiration, an incomplete and out-of-date attempted restructuring exists on an old branch. While I'm not suggesting picking up the work on that branch, it might give some good ideas on how to go forward.
- Wanna see something pretty? next.openmrs.org currently has a new Wordpress theme we'll be redesigning www.openmrs.org for. The infrastructure team would love to use the REST api from this project to allow users to create IDs directly from the Wordpress site.

Resources

- Current OpenMRS ID source code: https://github.com/openmrs/openmrs-contrib-id
- *A Reprise of LDAP*: a good blog post giving an overview of what LDAP is, and info about ldapjs, the library we intend to incorporate in this new data model http://blog.nodejs.org/2011/09/08/ldapjs-a-reprise-of-ldap/
- *LDAP for Rocket Scientists* is a good guide to for those new to LDAP. It is not, in fact, for rocket scientists.
- *Mastering OpenLDAP* by Matt Butcher is resource for learning the fundamentals LDAP. It's available as an e-book.

Get In Touch

Contact Elliott Williams at **elliott@openmrs.org** or **elliott_w** on #openmrs to chat about the project.

Ongoing Project Details

The project is assigned to Weihua Cheung now. Below are the project plans

Abstract

The current one is built with OpenLDAP and restricted to its limit. We can't store flexible user information on the current one. Also, the current one didn't organized the code very well.

The new module will provide a more extendable solution for storing user information, with a better organized structure, like MVC. In addition, it will provide a way to integrate the data from different website, like Facebook, Google, Twitter, etc, and user might log in with these accounts. Meanwhile it should provide APIs for outside access of data, and probably a simple website for query purpose.

Goals

- Refactor the code, decouple functionality.
- Beautify the code, use better code style for it.
- Provide more extendable user data model.
- Build REST and OAuth-backend API.
- Integrate with accounts with other website.
- Maybe add some new features.

Expected Timeline

- **March~April:** Get familiar with the OpenMRS-ID Platform by solving the ticket, and continue to learn things about Node.js, express and LDAP.
- **April~May(Before official starting):** Continue to learn. Make some contributions and try to design a good architecture, maybe a data model as well. And test if they are feasible.
- **19th May:** *The official commencement.*
- **1st-3rd Week:** Refactor the project, build mock-up for database related operations first.
- **4th Week:** Fully test it, begin to build the new db operation.
- **5th-6th Week:** Build the database module then, and assembly it with main frame.
- **7th-8th Week:** Implement the RESTful APIs, build a simple viewing webpage for direct query.
- **9th-10th Week:** Try to integrate user info from other website, like, Twitter, Facebook, Google+, etc.
- **11th Week:** Add some other fancy features, and put the project under strict test.
- **18th Aug:** *Official 'pencil down date', try to complete every doc and test.*
- **22nd Aug:** *Official final evaluation.*

Real Progress

This field will be updated weekly.

- **March:** Learning stuffs about Node.js async design pattern, solving my net connection problems.
- **April:** Getting familiar with OpenMRS-ID.
  - ☑ Solve the ticket about desk.com.
  - ☑ March on, fix problems of my local site.
  - ☑ Learn stuff about refactor and architecture design.
- **May:** Start working

  - ☑ Get my blog settled http://pages.plypy.com.
  - ☑ Have a look on Node OpenIDConnect (Abandoned).

- **GSoC 1st Week**: Refactoring
  - Settled the code style, we'll use the Felix's(See resources below).
  - ☑ Create a public issue for ongoing works.

    ⚠ ID-22 - Jira project doesn't exist or you don't have permission to view it.

  - ☑ Reorganize routes.

- **2nd Week:** Continue refactoring & begin to mess around with mongoose
  - Completed the works of 'moving things to its own module'.
  - Dealing with tons of indentations :/
  - Refactoring suspended and merged works so far
  - Starting to work on Mongoose
- **3rd Week**:
  - ☑ Build the basic data model under Mongoose.

  - Consider how to do unit testing, and maybe refactor the structure again, decouple more for easily testing.
  - Created ID-24, ID-25
- **4th Week:**
  - Fixed ID-6

  - ☑ Try to integrate new data model with signup module

  - ☑ Move the back-end validation to signup module

  - ☑ Complete the signup module

- **5th Week:**
  - Updated the auth module
  - Adapted to the new ui created by Elliott
  - ☑ Replace session store

  - ☑ Replace verification module

- **6th Week:**
  - Basically finished the integration works
  - ☑ New UI for profile page

  - ☑ Perfect all validations

- **7th Week:**
  - Finished validation works
  - Worked out a prototype for edit-profile page
  - ☑ Data migration plan (Temporarily urned to a dynamical migration approach)

  - ☑ Start to working on LDAP layer with ldapjs (Abandoned, turned to sync with OpenLDAP)

- **8th Week:**
  - ☑ Finsh migration script

  - ☑ Decouple the mongoose model with other module

  - ☑ Write installation walk-through for dashboard 2.0

  - ☑ Figure out a way to manage groups in LDAP

  - ☑ Update LDAP part and try to achieve 2-directional sync   **Abandoned**

  - ☑ Write tests

  - ☑ Update dependecies and related API usages

  - ☑ Use formage to build a admin console see ID-35

- **9th Week:**
  - Try to deployment!!!
  - Solve all kinds of problems.
  - Documentations
  - **Bravo** for Dashboard 2.0!
- **10th Week:**
  - Deployed the Dashboard 2.0 on production!
  - Discovered few issues and get them settled, like ID-47, ID-48...
  - Moving things back to my home... officially having announced the ending of this big semester.
- **11th Week:**
  - Picked things up again, fixed ID-51
  - ☑ Learn passport.js, figure out a prototype for our RESTful API.

  - Built a very prototype for the REST module, published and will be completed here
- **12th Week:**
  - Fixed ID-48, ID-51
  - Working on refactoring the OAuth module first, PR for ID-52 here
  - Start the dependency upgrade (express to 4.x, node to 0.10.x...) at this branch ID-55
  - Working on the final presentation
  - Discussed the RESTful APIs here
- **13th Week:**
  - Done with the Final presentation here
  - ☑ ****, Ubuntu died after upgrade, save the data

  - Spent a few days in hospital for my throat
- **Final Week:**
  - Documentation, ID-56
    - ☑ new installation guide

    - ☑ development guide

Resources

[OpenMRS ID OpenLDAP Explained](#)

[OpenMRS ID OpenLDAP Infrastructure Picture](#)

The [Felix's](#) code convention.