GSoC 2021: System Management

<table>
<thead>
<tr>
<th>Primary mentor</th>
<th>Daniel Kayiwa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup mentor</td>
<td>Herbert Yiga</td>
</tr>
<tr>
<td>Assigned to</td>
<td>TBD</td>
</tr>
</tbody>
</table>

This project is part of GSoC 2021: Modernizing Administration functions for OpenMRS

Description

Porting administration functions for managing system settings and modules within OpenMRS to OpenMRS 3.0: A Frontend Framework. This project focuses on global properties, scheduler settings, and module management.

The primary goal of this project is to create working micro frontend module(s) for these administration functions that can plug into the Administration Dashboard, meaning these functions will be expected to be developed using the best practices defined by the OpenMRS 3.0 Frontend Framework (Carbon Design, React/TypeScript) and register with the administration dashboard (mechanism to be determined), providing a title and a React component to render the administration function(s). The dashboard project will be responsible for organizing available administration functions, overall screen layout, and navigation between administration functions.

This project may include writing Java code to extend the REST Module to create needed REST endpoints. Those functions that have been implemented as Open Web Apps (OWAs) likely have the necessary REST endpoints, but functions that exist only in the legacy UI may not have yet been exposed as REST endpoints. The REST of Administration project can help in filling these gaps.

"OWA" refers to Mozilla’s Open Web Apps (OWA), a legacy standard from 2010 that existed before frontend frameworks like React, Angular, Vue, etc. were popular. In the early 2010s, OWAs were an attempt in the OpenMRS community to use a standard to create client-side applications. OWAs are not recommended for new development; new development is encouraged to adopt conventions of the new Frontend Framework.

Project Champions

- Burke Mamlin
- Daniel Kayiwa

Skills Needed

- Familiarity with Java and JavaScript
- Familiarity with React and TypeScript
- Good understanding of REST
- HTML/CSS

Objectives

- Create a "Hello World" ESM module to plug into the Administration Dashboard following conventions of the Frontend Framework (React, TypeScript, Carbon Design). See this page for an overview of the plan. The module should register itself as an administration function and, when selected, display it's "Hello World" within the Administration Dashboard’s content pane.
- Implement the following administration functions as ESM module(s), each registering as an administration function.
- For those administration features that do not have existing REST endpoints, create (or coordinate with the GSoC 2021: The REST of Administration project to create) those endpoints by extending the REST Module.

Below are the administration functions to be recreated in the new frontend framework. Examples of existing versions of each functions (in the Reference Application, as an OWA, or in the Legacy UI of OpenMRS 1.x) are provided for reference.

Scheduler

Provides the functions to manage scheduled (background) processes within OpenMRS. Implemented in System Administration > Manage Scheduler as well as within the Advanced Administration pages.

- OWA-based implementation in Reference Application System Administration - (System Administration > Scheduler) - github
- Legacy UI implementation - github
Manage Scheduler

List/search, add, edit, start, stop, run, and delete scheduled tasks.

OWA implementation in Reference Application

Legacy UI implementation (System Administration > Advanced Administration > Scheduler)

Maintenance

Provides some key miscellaneous administration functions for OpenMRS, including access to fundamental configuration settings.

- OWA-based implementations - [github]
- Legacy UI implementations - [github]

System Information

Display system information

OWA implementation in Reference Application

Legacy UI implementation
Settings

Allow view and editing of global properties organized by module

OWA implementation in Reference Application System Administration

Legacy UI implementation in Advanced Administration: Settings (grouped by module) & Advanced Settings (all properties ungrouped)

View Server Log

Show server log entries

Legacy UI implementation in Advanced Administration

View Database Changes

Show database changes (liquibase log)

Legacy UI implementation in Advanced Administration

View Logged In Users

Show a list of logged in users

Legacy UI implementation in Advanced Administration
Search Index

Provide a button for an administrator to rebuild their search indexes

Legacy UI implementation in Advanced Administration

Modules

Provides functions to install, start & stop, configure, and remove OpenMRS modules. Implemented in System Administration > Manage Modules as well as within the Advanced Administration pages. These features should respect the runtime setting `module.allow_upload=false` to disable module management from the web interface.

- OWA-based module management in Reference Application System Administration - [github](https://github.com)
- Legacy UI implementation of module management - [github](https://github.com)

Manage Modules*

List/search, install, start, stop, and remove modules.

Legacy UI implementation in Advanced Administration

Module Properties

Location of modules and whether or not they can be uploaded.

Legacy UI implementation in Advanced Administration

Extra Credit

- TBD

Resources

- Read up on the new OpenMRS Frontend framework: [OpenMRS 3.0: A Frontend Framework that enables collaboration and better User Experience](https://openmrs.org)
- Read up on FHIR. Good introductions can be found in this post, in this slide show or this video.
- Read up on OpenMRS REST web-services: [REST Module](https://openmrs.org)