Portal_Documentation

This is an outline of information relevant to the architecture and maintenance of the portal, with links to further documentation where appropriate.

1. Portal architecture
   ♦ Essential support services / servers
     1. *Git code repository*
     2. *Sourceforge repository*
     3. *Redmine project management / wiki for the portal*
     4. *NEMO wiki (this site)*
     5. Master spreadsheet that lists all portal form fields and has development notes. Check out from sourceforge svn:
        $ svn co
        https://nemoontologies.svn.sourceforge.net/svnroot/nemoontologies/RDFdata/working/DataAcquisProv2RDF_spreadsheet
   ♦ Three application server environments
     1. *Local development on a laptop/desktop*
     2. Development server: http://dev-portal.nemo.nic.uoregon.edu
     3. Production server: http://portal.nemo.nic.uoregon.edu
   ♦ Components required in all environments
     1. Database: *PostgreSQL 8.x*
     2. Language support: *Ruby 1.9.2p180*
     3. Web application framework: *Rails 3*
     4. Version control: *Git client*
     5. Deployment: *Capistrano*
   ♦ Additional components required in dev/prod environments
     1. *nginx web server*
     ♦ Web app implementation details
        1. Authentication with [http://www.myopenid.com OpenID]
        2. Authorization with *CanCan*
        3. *Field wrappers*
        4. *Nav bar*
        5. *CSS styles*
        6. *Object hierarchy*

2. Administrative tasks in the portal
   ♦ Adding a portal user in conjunction with [new NEMO user steps].
   ♦ Managing existing users: add "/users to base server URL, whether on dev, prod, or local server.
      User roles.
   ♦ When ontology is updated: Refreshing ontology terms, db-onto-maps, and field annotations in the portal.
   ♦ Managing field annotations

3. Development tasks
   ♦ Check out code from git repository to a local dev environment and work on it locally. Includes checking back in and other git workflow.
   ♦ Deploy code to dev/prod servers with Capistrano.
   ♦ Perform database dumps and loads
   ♦ Change database schema with migrations