

Lois J. Greene-Hernandez
Sunnyside, NY 11104
loisgh@gmail.com
[linkedin.com/in/loisgh/](https://www.linkedin.com/in/loisgh/)
loisgh.info
917-549-1132

Software: ActiveRecord, Agile, Agile Methodologies, Apache, BDD, Cucumber, Fedora, Flask, Freemarker Templates, Git, HBase, Hibernate, J2EE/JDBC, JQuery, JUnit, Java6, Java7, Javascript, JQuery, Jinja, Jira, Linux, MSSql, Maven, Mockito, MySQL, Oracle, PHP4, PHP5, Pivotal Tracker, PostgreSQL, Python 2.7, Rails4, Rspec, Ruby, Ruby on Rails, Spring Dao, TDD, Tapestry, TestNG, Tomcat, Unix, Wicket, Wicket Charts/High Charts, Wordpress, XML, XSLT, foreman, github, heroku, hibernate, rvm

Bank of America thru Apex Systems

December 2016 – February 2018

Python Developer – Python 2.7, Flask, Jinja, HBase, Jira, Unittest, JQuery, Javascript

- Worked on Python and HBase copy process. Copy process used generators to copy paths from one environment to another in the case of a data outage.
- Created online page using Python, Flask and Jinja templates to update input parameters for job submittal.
- Created Data Validation Framework to validate the many data paths of market data that we archive.
- Online component lets users can set up job configurations for paths to be validated.
- Backend component does the actual validation based on schedule entered in the online configuration.
- Configurations can be customized to include:
 - exceptions based on path.
 - Exceptions based on path and column
 - Expiry date calendars
 - Holiday calendars to avoid reporting data as missing on business holidays
- The basic validations are:
 - Check that the path is not empty
 - Check that the data in the path is valid
- Other optional validations are:
 - Flat check (making sure that data changes daily and that loader API isn't getting stale data.)
 - Negative check (making sure that data isn't negative)
 - Futures contract count (ensuring that categories of futures have the correct number of contracts)
 - All job validations can be set through the configuration panel.
- Users can subscribe themselves to receive validation alerts on spreadsheets.
- Distribution email lists can also be subscribed/unsubscribed to validation alerts.
- All Validation notices are written out to an amps message queue. Amps queue errors are then written out to a Geneos ITRS dashboard. This is a dashboard used by our support staff to respond to issues.
- Four weeks worth of validation data is persisted to help us identify problems and patterns.
- Created an online application to copy job configurations from one environment to another or to delete configurations from environments.
- Removed the need to maintain a clunky copy script that was used for all sets of configurations.
- Configurations could be copied instantly without waiting for the next release cycle.
- Configuration copy was written in Python/Flask with JQuery and some angular.

Case Commons

February 2016 – April 2016

Ruby on Rails Developer – Ruby, Rails4, TDD, Rspec, ActiveRecord, Pivotal Tracker

Worked on Casebook which is an application that allows social workers to track their cases for the Indiana department of children's services.

Knoa Software

July 2014 – January 2016

Software Engineer – Java7, Wicket, TestNG, JUnit, Mockito, Spring, Hibernate, Services, MSSql, Oracle, Wicket Charts/High Charts, Maven, Freemarker Templates, Agile, Scrum, Jira, Jenkins.

Worked on enhancements to the console portion of a system monitors the user activity of SAP and Siebel software. The console configures and provides information about the knoa server and agents. Worked on agile team with weekly dev sync and scrum.

- Dictionary Mapping allows for the meaningful consolidation of agent tracking data in user reports. Data is collected from the agent in Raw Strings. Dictionary Mapping involves mapping those raw strings to rules based on star patterns. Created new Raw Strings Tab Panel. This page allows for the creation of mapping rules by raw strings. Rules can also be modified and deleted as well. There is also a search function. Data is displayed in a wicket data table complete with pagination. Created services
- Created a home page that shows a graph and table with Mapping totals sorted by dictionary and by SAP solution. The table is created by sorting and totaling all dictionary entries by various categories in to a collection which is

then loaded into a wicket data table. The graph is created with wicket charts, which is High Charts in a wicket wrapper.

- Alerts system tracks errors and issues that occur in our system through the console interface. Some alerts are generated via regularly scheduled jobs or processes. Others are generated via the server and put in a message queue, which is then consumed by the console and added to an alert database. The alert system generates emails to certain users and provides reports of all issues. Created the Alerts Reporting page, which displays all alert data in a wicket table with sorting and filtering functionality. Each alert links to some sort of detail page that provides more information. In some cases the detail links to a console log with a time parameter so that the logs pertaining to the alert can be displayed.
- Created unit tests for all code changes using the Mockito framework.
- Created services for data and functionality that comes from jmx calls.

New York Public Library

August 2012 – July 2014

Application Developer – Ruby on Rails3, Agile Methodologies, BDD, Cucumber, Git, Solr, Fedora, github, Jenkins Amazon EC2, heroku, rvm, Java6 and 7, hibernate, spring, foreman, knowledge of Ruby2 and Rails4

- Worked on API to return results from solr and fedora requests. API used a separate class for the fedora and solr calls. Gem RSOLR is used for solr calls. NET:HTTP is used for fedora.
- Worked on Admin page to allow prospective api users to sign_up for username/password and auth-key access. Users can also edit their ID's. An Admin ID can also edit and destroy Users.
- Created cucumber Tests to for login, sign_up and edit pages. Created cucumber tests for api calls as well.
- Worked on Digital Collections website. Created user front end to allow them to store images and image information for the featured images on the homepage. Digital Collections website is deployed in the cloud. (Amazon EC2)
- Worked with MSSQL to marshall data into MODS format to be ingested into our system. Initial data was pulled from a MSSQL database using Java/Spring.
- Worked on a SIPIngest process. SIP's are archives that contain data and metadata. The metadata is ingested into our system where it is stored in Oracle, fedora commons and eventually indexed in SOLR. This process is meant to handle many different data types. (e.g. OCR data, Still Images, Audio and Video). Worked on sub modules for Video and Still Images. Also worked on the hibernate piece that writes to our file_store database.
- Created rails "Linkminter" project. Our Cataloguers often need permalinks before they know the actual destination of the link. This rails project provides a front end to the java endpoint that creates links. Also, once the destination is known it allows users to update the links.
- Worked on Metadata dashboard. Dashboard shows vital statistics for all of our data and answers various questions for those that use our data and apis to power their systems. (e.g. how many items were digitized recently? Why isn't my item showing up in digital collections.) This system allows our users a more accessible view into fedora commons systems and solr which power our systems.

School of Visual Arts

March 2008 – August 2012

Webdeveloper/Webmaster – J2EE/JDBC, Ruby on Rails, Apache, Tomcat, PHP4, PHP5, Wordpress, Subversion, MySQL, PostgreSQL, hibernate, Tapestry, XML, XSLT, Linux, Unix

- Created Rails 3/AJAX app to query our employee database for faculty members and return ID information.
- Worked on a Rails 3 app to query our highly confidential student database in the event of an emergency. App is to be deployed on iphones/androids by select members of the Crisis Committee. App uses TinyTDS and FreeTDS to connect to a MSSQL database. Front end of app is jquery mobile.
- Maintain and enhanced various Java/J2EE(JDBC) Systems. These include a 1) Vacation request System, used by staff to request time off. 2) Faculty Forms System, used by faculty and staff to process and approve faculty, course and schedule changes. 3) Continuing Ed course finder, which lets prospective students find the course they are looking for. The course finder is a built using Hibernate and Tapestry.
- Created JSP shopping cart to improve the Continuing Education registration process.

Prudential Financial

February 2007 – March 2008

Developer – Java, J2EE, Blaze, Oracle, Sybase, Unix

- Created Blaze Rules to process financial feeds for the organization. Wrote java beans to read the data entities used to process the feeds. Feeds create records in an Oracle table which is imported into Oracle Financials.
- Created tables for our FTMS system in both Sybase and Oracle. FTMS is a system that provides a way for users to enter and maintain data in tables. The tables are maintained on the Sybase side and then imported into Oracle. Wrote Sybase and Oracle scripts to create tables and define their properties. Wrote Sybase triggers for the Oracle import.
- Created SQL statements and PL-SQL stored procedures to used in various ITG for or workflow validations.

TBWA/Chiat Day

June 2006 – February 2007

PHP Developer – PHP4, Ajax, XML, Mac OSX

- Created generic AJAX employee information web page for company intranet. Web page does AJAX call to dynamically create XML with employee information.
- Maintain Print Media system. This is a PHP/MySQL/Smarty system used to generate ads that will appear in regional and national newspapers. All PHP projects were developed using ZEND.

Broadview Networks

October 2005 – April 2006

Java Developer – Java, JDBC, Swing

- Designed and Implemented production fixes and major enhancements to Broadview Networks Provisioning Utilities. Provisioning Utilities are used to bring new phone customers on to Broadview Networks Switches.
- Created CNAM transaction using Java/Swing to update Verisign Calling Name database.
- Created Java modules for the Gateway to create the outgoing message sent to Verisign as well as the incoming response. The Gateway is a Java/Swing based system that processes all transactions and their responses.
- Modified E911 utilities system to handle Rhode Island Error responses. This system updates the Bell Atlantic E911 database. Automated the handling of Rhode Island E911 Errors.

DoubleClick

February 2005– July 2005

Quality Assurance Engineer – Unix, Perl

- Memory Server/Ad Server multicast Optimization project. Verified that jump (clickthrough) information could be passed by Memory Server as well as the Ad Server. Helped reduce load on the Ad Server.
- User Information Server Multiple Ad Family project. Verified that UIS which provides targeting and user data to the Ad server could receive and store more information. Ad Families allows advertisers to control the serving of large classes of ads.

Unicast/Viewpoint

June 2004 – February 2005

Quality Assurance Engineer – Unix, Perl, HTML

- Responsible for quality assurance of web-based ad play and ad operations applications. Performed functionality, white box, compatibility, usability and regression tests on both applications. Ensured referential integrity between ad operations and ad play applications.
- Projects worked on include optimization of Ad Play .

EDUCATION

Queens College

Bachelor of Arts Degree

Sun Microsystems

Sun Certified Java Programmer

Agile Training