**Building DICOM Cleaner with Google Integration Demo**

1. Clone repository that contains updated DICOM Cleaner project : <https://github.com/codeminders/dicomcleaner.git>
2. Build this project using ant tool. There is build.xml file for Ant.

Execute: “*ant* ” or “*ant –f build.xml*”

1. After execution there is dist folder with dicomCleaner.jar in it. It is ready for execution.

**Changed/added components overview**

1. Added classes:

Package com.codeminders.demo.model: Location and ProjectDescriptor - classes for location and project descriptors;

Package com.codeminders.demo:

* GoogleAPIClient – class that makes communication with Google cloud;
* DICOMGoogleClientHttpRequest – class that writes DICOM data to Google cloud;
* DICOMStoreDescriptor – DTO with information about used DICOMstore;
* GoogleDICOMExport – class that implements export DICOM files to Google cloud;
* GoogleDICOMImport – class that imports DICOM data from Google cloud;
* GoogleDicomstoreSelector – UI JPanel that allows user to choose Project, Location, Dataset and DicomStore during import/export procedure;
* ReportPanel – UI JPanel that renders report to user;
* ReportService – class that accumulates report data from all components during import, clean, blackout and export procedures;

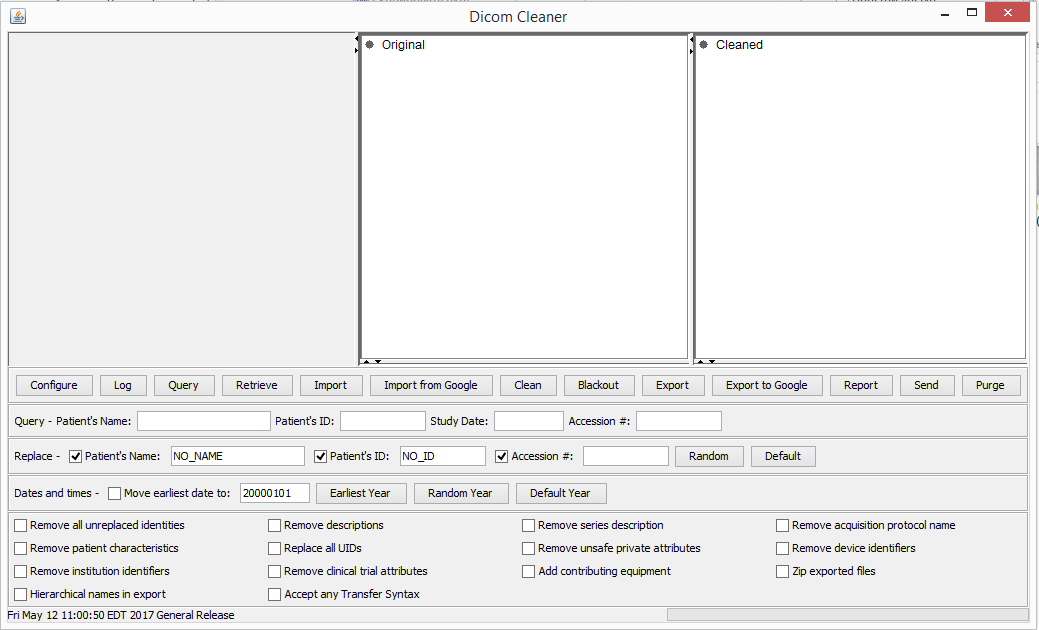
1. Changed:

* com.pixelmed.display. DicomCleaner – UI frame class that implements all UI for application. We have added new buttons in this UI:
* “Import from Google” – imports data from Google. If there is no active token then user should authorize on Google cloud and provide needed access;
* “Export to Google” – exports data to Google. User chooses any DICOMStore for export. If there is no suxh DICOMStore it will be created on Google cloud.
* “Report” – shows rendered report;
* com.pixelmed.dicom.MediaImporter – integration with ReportService added, so we can track data;
* com.pixelmed.display.DicomImageBlackout – integration with ReportService added, so we can track data about images that were blackout;
* com.pixelmed.dicom.AttributeList = one of validation check skipped, this allows to demonstrate for DICOM images;

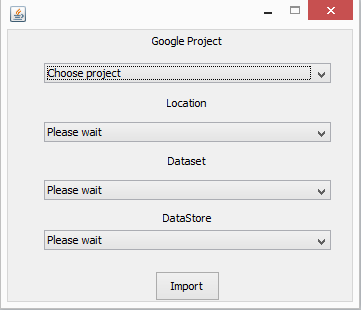
**Executing DICOM Cleaner with Google Integration**

1. There is “dist” folder with exported jar file dicomCleaner.jar. To run DICOM Cleaner execute:

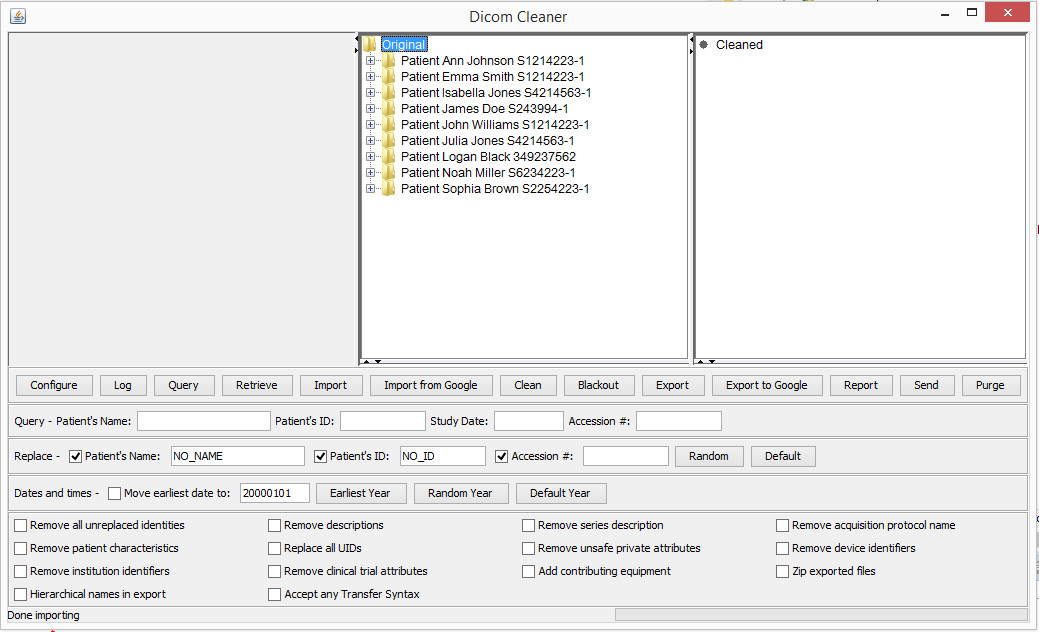
java –jar dicomCleaner.jar



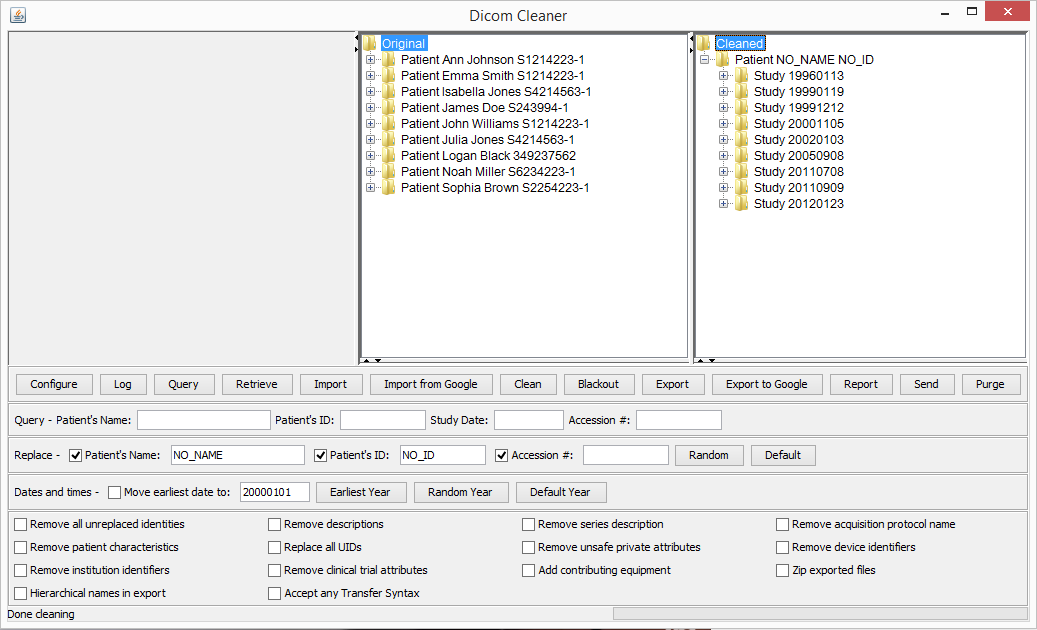
1. Press “Import from Google”.
2. Sign in to Google with your account connected to GHC.
3. Choose source for import service by choosing project, location, dataset and dicomStore:



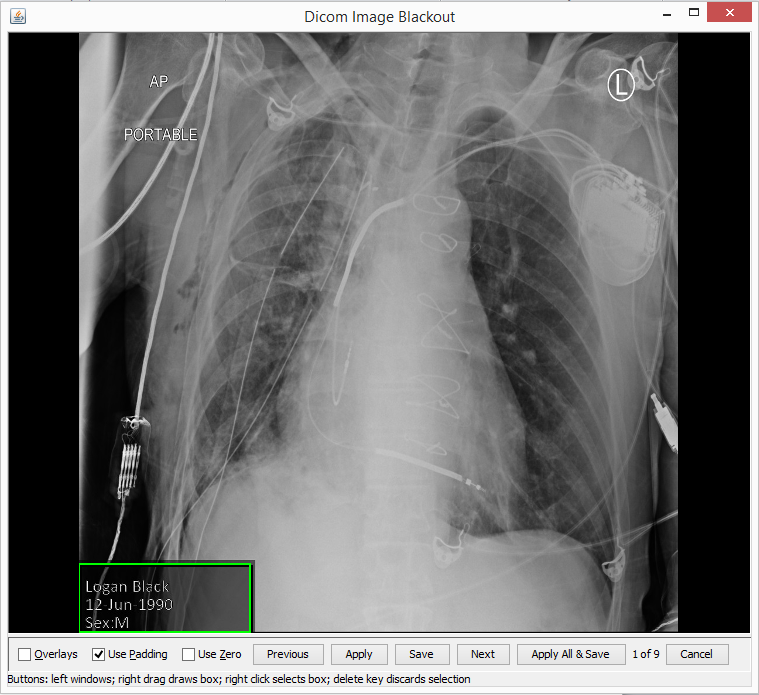
1. Press “Import”. Wait till data will be imported from Google.
2. Choose dicom files for anonymization, by clicking on “Original”



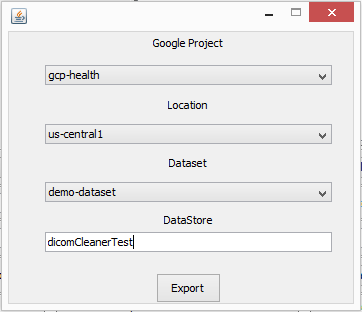
1. Press “Clean”, wait till the end of this process.
2. Choose files for pixel anonymization.



1. Press “Blackout”. Tool for pixel anonymization should appear with first image in it.



1. With right mouse button choose all places where image need pixel anonymization. Then press “Apply”, “Save” and “Next”. Repeat this procedure with all images.
2. Press “Export To Google”. Choose Project, Location, Dataset and enter DataStore (new or old)



1. Wait till data will be exported.
2. Press “Report” to see the result report.

