# HP Data Quality ETL App Fronted Assembly Specification

# Project Overview

## System Description

Client wants to build an Extract Transform and Load (ETL) application that will allow user to move data between a source file or database and a destination directory or database. The goal of this challenge is to create the underlying architecture for this Data Quality app. The tool will allow moving file and relational data on a regular basis and will display the status of the currently scheduled and recently completed jobs.

This assembly is responsible for converting prototype into fully functional web app that accesses backend via REST API.

## Competition Task Overview

A complete list of deliverables can be found in the TopCoder Assembly competition Tutorial at:

<http://apps.topcoder.com/wiki/display/tc/Assembly+Competition+Tutorials>

*Note: Extensive implementation notes are provided at method documentation on UML Class Diagrams. Please follow them for implementation.*

*Note: Please read the whole Application Design Specification first. All the details not mentioned in this specification are provided in that document.*

### 00-1-Login.png

/login endpoint will be used.

### 01-1-Dashboard\_Grid.png

/projects and /jobs endpoints will be used to get user projects and jobs and all related information like executions.

### 02-1-Settings.png

/projects endpoint with POST method will be used to create the project.

### 05-1-Job-Details.png

/jobs endpoint with GET method will be used to get all the information related to job. /jobs with PUT method will be used to update the job including rescheduling. jobExecutionStatuses/{jobExecutionId}/pause will be used pause the job execution. In the same manner other operations like resume or start will be performed.

### 06-3-2-Create-Job\_Mapping\_filled\_out.png

/jobs endpoint with POST method will be used to create the job.

## Technology overview

## Java 8

## REST API

## JSON

## Spring Framework 4.x <http://projects.spring.io/spring-framework/>

## Spring Boot 1.3.x <http://projects.spring.io/spring-boot/>

## Log4j 1.2 <http://logging.apache.org/log4j/1.2/>

* Apache POI 3.15 <https://poi.apache.org/download.html>

## Existing Documents

* Class Diagrams
* Sequence Diagrams
* Application Design Specification
* Assembly Specifications
* ERD
* Swagger API definition