

AQUENT Sample Development Project Specification

The following details are the high level project specifications for the interview sample project.

Platform and framework requirements:

- Apache Tomcat 7
- Maven 2
- Spring MVC
- Spring JDBC
- SQL database: in memory, local file, or optionally PostgreSQL
- JSP views
- HTML5, CSS, and jQuery

General Notes

- Pretend your code will be part of a large production codebase. Use proper layering, data models, javadocs, and general best practices.
- Maven 2 should be used to build the application and create a deployable WAR file. This WAR file must be deployable to Apache Tomcat 7.
- Any questions should be directed to your interviewer(s). Questions should be limited to discrepancies in this document.
- Upon completion, please send the project to your interviewer(s) within the allotted time frame.

UI Overview

This application is a simple CRUD contact manager web interface providing the following functionality:

- Listing of people
- Creation of a new person record
- Editing of an existing person record
- Deletion of an existing person record

The layout / design of the pages may be kept to simple links, buttons, etc. A full featured graphical design is not required.

Person Listing: On the main page of the application, a list of people found in the system should be displayed. The following summary data should be shown for each person record:

- First name
- Last name
- Email address

There should also be a mechanism to edit (initiating the workflow defined in the “Edit Person” section) and delete (initiating the workflow defined in the “Delete Person” section) each record.

If no results are found, “No results found” should be displayed.

A mechanism to create a new record should also be provided, initiating the “Create Person” workflow below.

Create Person: The workflow should capture the following information about a person:

- First name
- Last name
- Email address
- Street address
- City
- State (abbreviation)
- Zip code

Validation should occur for each of these fields. This may be performed on the client, server, or both. Validation messages should indicate the field that failed and why. The following validation rules should be applied:

- First name: non empty, max 30 characters
- Last name: non empty, max 30 characters
- Email address: non empty, max 30 characters
- Street address: non empty, max 60 characters
- City: non empty, max 30 characters
- State: exactly 2 letters
- Zip code: exactly 5 digits

At the end of the workflow, the person data should be saved and the user should be returned to the main page (Person Listing).

Edit Person: The workflow should capture the same information and steps about a person as in “Create Person”. Validation should also occur for this workflow, using the same rules as “Create Person”. At the end of the workflow, the person data should be saved and the user should be returned to the main page (Person Listing).

Delete Person: A confirmation page or dialog should be shown, stating “You are about to delete the person: \$FIRST_NAME \$LAST_NAME, are you sure?”, followed by “Delete” and “Cancel” links or buttons.

Upon clicking Delete, the person should be deleted from the system, and the user should be returned to the main page (Person Listing).

Upon clicking Cancel, the user should be returned to the main page (Person Listing).