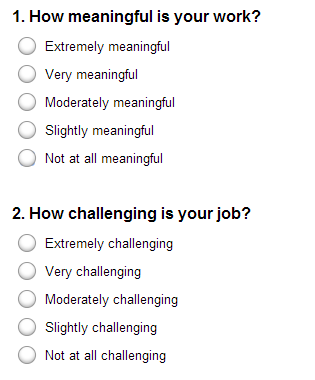
**Capabilities covered:**

* Understand the web standards, patterns & protocols
* Define client & server architecture
* Implement Web UI that meets the functional specifications
* Add actions, interactions, business logic using JavaScript (ES6 & TS)
* Use Spring MVC framework to implement presentation layer of an application
* Use Spring Security framework to set up basic security controls for an application - including authentication, authorization, and data transport security (SSL)
* Use Hibernate framework to implement persistence layer of an application
* Analyse complex business scenarios and create a data model
* Use MySQL to build SQL engine for an application
* Use Java to implement business layer of your application
* Use Eclipse IDE for programming Java application
* Test, and Debug your application to make application ready for deployment
* Follow coding standard and maintain high quality by analysing various reports generated through code quality tools
* Automate the build using Maven

# **Problem Statement: Employee Satisfaction Survey**

Create a happier workplace using employee satisfaction survey. Happy employees are more likely to delight co-workers, customers, and partners and build the good reputation of your brand. Use employee surveys to check in with employees about their satisfaction with their roles and responsibilities, the work environment, and their experiences with management, etc.

The survey will have questions like:

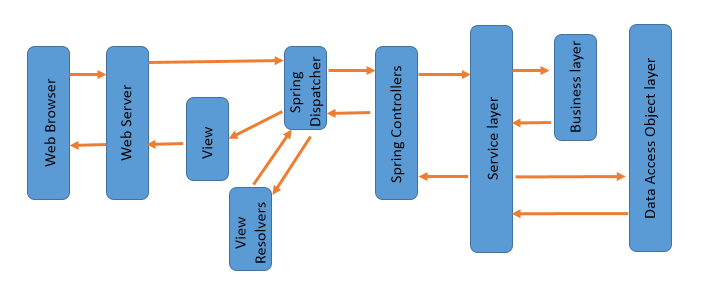


Pre-condition for the application is that tables should be created for employee with columns (employee ID, name, and email) in the backed and data has to be inserted. **Provide all the DDL and DML database scripts used for creating tables and records inserted. Insert at least 3 employees.**

**Note:** One question can have many options.

Instructions:

1. **READ THE QUESTION PAPER COMPLETELY BEFORE YOU START CODING.**
2. Use appropriate data types and precisions for the variables
3. Handle all exception and alternate flows
4. Validation must be implemented both at client side and service side
   1. Client side using Java script
   2. Server side using Spring validator framework (only for first use case, screen-1)
5. Apply CSS styles wherever applicable
6. Solution must follow below architecture



Import **“EmployeeSurvey.zip”** dynamic web project. The project should be completed using Spring MVC for web layer, JPA/Hibernate for persistence layer, MySQL for database, Use AOP/annotations for database transactions

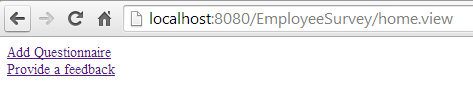
**The project contains:**

1. Exception classes
2. Completed deployment descriptor web.xml file
3. Spring business tier xml file with DataSource, SessionFactory, Hibernate Template and HibernateTransactionManager configuration xml schema declarations
4. Spring web tier xml file with View Resolver configuration
5. Employee entity class and Question entity without ORM mapping
6. EmployeeDao.java DAO interface
7. QuestionDao.java DAO interface
8. EmployeeSurveyService.java service interface
9. HomeController.java Controller class which will redirect to “home.jsp” from “index.jsp”
10. The database.properties file with Driver, URL, username, password configurations
11. The landing page “index.jsp” which forwards the request to “home.jsp” through the FeedbackController
12. The home.jsp page

Home page contains two hyperlinks:

**a) To add questionnaire**

**b) List questionnaire for feedback**



Things to implement:

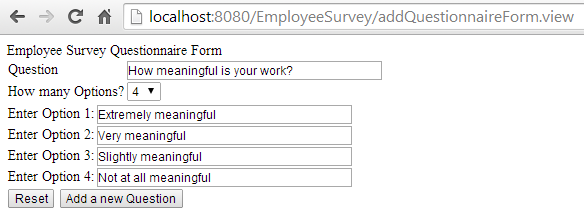
## **Screen 1**

**Adding Questionnaire for Employee Survey**

On clicking “Add Questionnaire” hyperlink on home page, user is redirected to Question Form.

Each questionnaire can contain 2 to 6 options. When user selects the number of options from “How many Options?” dynamically Text boxes for entering options has to be displayed. For example if user selects “4” as shown below, four labels and four text boxes are displayed.

Example screen for 4 options:



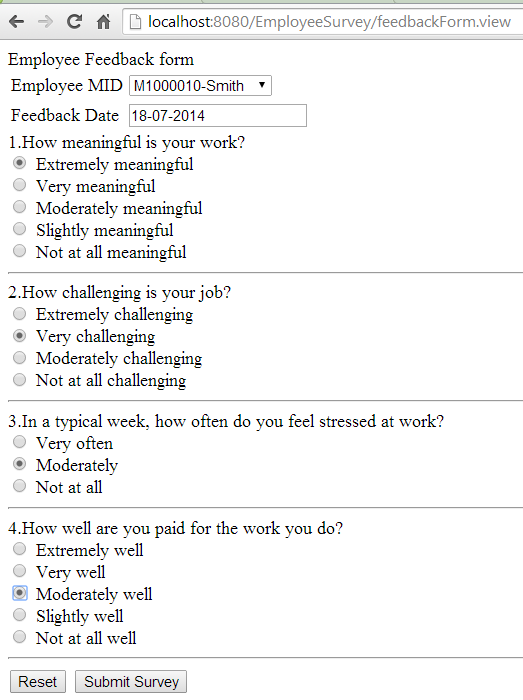
User enters the data and submits the form by clicking on “Add a new Question”. On successful submit Questions and options should be stored in the database tables. User should be redirected back to home page with message “Survey Question and options added successfully”.

**Validation has to be done using Java script and Spring Validation framework.**

1. All fields are mandatory.
2. Each option has to be unique.

## **Screen 2**

**Display Survey Form**



1. All employees present in the database should be displayed with MID-NAME.
2. Feedback Date should be pre populated with system data.
3. Questions and Options should be displayed
4. Use JavaScript to validate and no need to use spring validator framework for this use case, screen2

**Note:** No need to write code to capture the form data on click of “Submit Survey”

**XXXXXXXXXXXXXXXXXXXXXX END XXXXXXXXXXXXXXXXXXXXXXXXX**