Feedback Management System

Table of Contents

1 INTRODUCTION

This document outlines a mini project for the J2EE LOT. The project is to develop Feedback Management System for Training programs (FBS). This document contains the work flow of the system and gives guidelines on how to build the functionality gradually in each of the course modules of the J2EE LOT.

1.1 SETUP CHECKLIST FOR MINI PROJECT

Minimum System Requirements

- Intel Pentium 90 or higher (P166 recommended)
- Microsoft Windows 95, 98, or NT 4.0, 2k, XP, Windows 7
- Memory: 32MB of RAM (64MB or more recommended)
- Internet Explorer 6.0 or higher
- Oracle 10g
- JDK 8
- Eclipse Photon
- Junit 4.0 ,log4j

1.2 INSTRUCTIONS

- The code modules in the mini project should follow all the coding standards.
- Create a directory by your name in drive <drive>. In this directory, create a subdirectory Mini Project. Store your Project here.
- You can refer to your course material.
- You may also look up the help provided in the java docs.
- The total time required to complete this mini project is 50 hrs.
- Since this project work will span over couple of months, you will need to take care of maintaining the code

2.1 OBJECTIVE

Development of Feedback Management System (FMS)

2.2 ABSTRACT OF THE PROJECT

This project is aimed at developing a desktop feedback management system for Data global solutions . The Training department of Data global solutions constantly conducts several training programs for the employees .Collection of feedback and analysis of feedback is a very critical part of any training program. This desktop application helps the training department to improve the training execution based on the analysis of feedback reports.

2.3 FUNCTIONAL COMPONENTS OF THE PROJECT

Following is a list of functionalities of the system. Wherever, the description of functionality is not adequate; you can make appropriate assumptions and proceed .There are 3 types of users. All users are the employees of the Data global solutions. Assume that credential details of all the employees are existing in the database. The functionalities to be performed by the different users are as follows.

- Training Admin

    1 Maintain the skill set of faculties

    2. Maintain the Training courses

    3. View the feedback reports and feedback defaulters report

- Training C-coordinators

    1. Creation/modification/deletion of the Training Programs

    2. View the feedback reports and feedback defaulters report

- Participants

    1. Enter feedback for the training programs attended

Feedback data is collected from the participants to understand the positives and improvement areas of the training. Following parameters should be rated by the participants

1. Presentation and communication skills of faculty
2. Ability to clarify doubts and explain difficult points
3. Time management in completing the contents
4. Handout provided(Student Guide)
5. Hardware, software and network availability

Rating Terminology

5-Excellent: “Ideal way of doing it”

4-Good: “No pain areas or concern but could have been better”

3-Average: “There are concerns but not significant”

2-Below Average: “Needs improvement and is salvageable”

1-Poor: “This way of doing things must change”

2.4 TECHNOLOGY USED

- Front End :– 1. Class with main() function
- Business Logic Components and Services :- 1. Java Beans
- Databases:- 1. Oracle 10g

3 IMPLEMENTATION IN J2EE LOT

3.1 SUMMARY OF THE FUNCTIONALITY TO BE BUILT

The participants need to develop the desktop FMS by building the functionality incrementally in each of the course modules of J2EE LOT.

Sr. No	Course	Duration (in PDs)	No. of Saturdays	Functionality to be built
1	Programming Foundation with Pseudo code	3	1	Analyze the given case study
2	Introduction to Software Engineering	0.5		Analyze the Case study using SDLC phases.
3	Web Basics (HTML 5,CSS 3, JavaScript, XML)	4.5	1	Developing prototype i.e. developing screens/web pages in HTML and client side validation in JavaScript.
4	Oracle Basics	4	1	Creating relevant database tables
5	OOP & UML	1.5	1	Creating relevant Use case and class diagrams
	Programming Foundation with Pseudo code + Web Basics +Oracle Basics +OOP & UML Test	1		
	Core Java 8 & Development Tools (Junit, Log4j)	10	2	Developing Business components (java classes). Coding for test classes & testing the functionality using JUnit
	Core Java 8 + Dev Tools + OOP/UML Test	1		
8	Servlets	3.5	2	.Project specific implementation is not needed as mini project is in Core Java
9	JSP	2		
10	Developer Workbench (PMD, MAVEN)	1		
11	Servlets + JSP + Dev Workbench Test	1		
12	Basic Spring 4.0	5	1	Prepare document for presentation.
13	Basic Spring Test	1		
14	Mini Project presentation	1		

3.2 GUIDELINES ON THE FUNCTIONALITY TO BE BUILT

The functionality and components to be built in each of the course modules of J2EE LOT is as follows: Convert your HTML Pages into Console Screen using Java class with below options.

1. Course: HTML, JavaScript

a. Develop the following screens:

i. Login: All employees are authenticated in this. If the supplied user credentials are valid, then details is displayed according to the user type, and otherwise error message is displayed.

ii. Home: On successful user authentication, the user details is displayed with appropriate options according to the type of user.

Training Admin Role:

- Faculty skill Maintenance
- Course Maintenance
- View Feedback Report

Co-coordinator Role:

- Training program Maintenance
- Participant Enrolment
- View Feedback Report

Participant Role :

- Feedback Entry

iii. Faculty Skill Maintenance page:

This should allow the mapping of course to the faculties based on the skills. This displays the faculty list and the course list

iv. Course Maintenance page:

This should allow the maintenance of course details like course name, duration

v. Training Program Maintenance page:

This should allow the maintenance of Training program details like TrainingID, Trainingcourse, faculty scheduled for the course, start date and end date. Course list and the faculty list mapped to that particular course must be displayed on console.

vi. Participant Enrolment Screen:

This should allow enrolling the participants to the different training programs. Participant id can be either selected or typed to enrol to the Training programs. Training programs must be listed on console.

vii. Participant Feedback Screen:

This should allow the participants to enter the feedback for his/her completed training program

viii. Feedback Report:

This should be available to Training admin and coordinators Shows the options for the following reports

Report 1: All Training programs Report with average feedback for the selected month

S.no	Date	Training	Faculty Name	Feedback Scores				
				Pres&comm	Clarify dbts	TM	Hand out	Hw/sw/ ntwrk
1								
2								
3								

Average Scores:

Report 2:Faculty wise report with average feedback for the selected month

S.No	Date	Training	Feedback Scores				
			Pres&comm	Clarify doubts	Time Mngmt	Handout	Hw&Sw
1							
2							
3							

Average Scores:

Report 3:FeedbackDefaulters report for the selected month

From the compiled results, a report is generated as shown below .This lists the participants who are yet to provide feedback or have provided incomplete feedback (indicated by X)

S.No	Date	Trng	Participant Name	Faculty Name	Feedback Scores not Provided				
					Pre &Com	Clarf doubt	Time Mngmt	Hand out	Hw/Sw
1	03/03	XYZ	Student #1	Faculty #1	X				X
2									
3									

b. In this course you need to develop the user interface using java classes and document the flow of your application including the screen shots in a word document. The screens/web pages should include the fields as per the functionality mentioned above. Also, include client-side validations using Regular Expression in each of these screens

2. Course: Oracle(Duration: 11 hours)

Create the following database tables:

TRAINING_PROGRAM

Field Name	Description	Datatype	Size
Training Code	Running Sequence Number	Number	5
Course code	Course id foreign key to course table	Varchar	50
Faculty Code	Faculty scheduled for training foreign key to employee table	Varchar	50
Start Date	Starting date of the training	Date	
End Date	End date of the training	Date	

COURSE_MASTER

Field Name	Description	Datatype	Size
Course ID	Running Sequence Number	Number	5
Course Name	Name of the Course	Varchar	50
No of Days	No of days	Number	5

FACULTY_SKILL

Field Name	Description	Datatype	Size
Faculty Id	Foreign key to employee	Number	5
Skill Set	Skills separated by comma	Varchar	200

TRAINING-PARTICIPANT_ENROLLMENT

Field Name	Description	Datatype	Size
Training_code	Foreign Key to program master	Number	5
Participant Id	Foreign key to employee	Number	5

FEEDBACK_MASTER

Field Name	Description	Data type	Size
Training Code	Foreign Key to Training _Master	Number	5
Participant Id	Foreign Key to Employee	Number	5
FB_Prs_comm	Number	Number	1
FB_Clrfy_dbts	Number	Number	1
FB_TM	Number	Number	1
FB_Hnd_out	Number	Number	1
FB_Hw_Sw_Ntwrk	Number	Number	1
Comments	Comments field	Varchar	200
Suggestions	Suggestions field	Varchar	200

EMPLOYEE_MASTER

Field Name	Description	Datatype	Size
Employee_ID	5 digit unique number	Number	5
EmployeeName	Name of the employee	Varchar	50
Password	To authenticate	Varchar	20
Role	For authorization	Varchar	20

3. **Course: OOP & UML(Duration: 5 hours)**
    a. Develop relevant Use case and Class diagrams for the **FMS**

4. **Course: Core Java 8+ Developer Tools (Duration: 14 hours)**

5. **Documentation(Duration: 2 hours)**
    a. Project Documentation: Document your project details (Duration: 1 hour 30 mins).
    b. Project submission: Submit your project with all the artifacts including the test cases & documentation (Duration: 30 mins).

3.3 EVALUATION AND ASSESSMENT PARAMETERS

This miniproject will be done in groups of five to six. Each group will identify a Team Lead who will decide which team member will code for which functionality. This project shall be evaluated at the end of spring module.

Evaluation Criteria (out of 100):

Look of console for all the screens	**05**
Client-side validation of inputs	**10**
Code Documentation and using coding standards	**10**
Overall Business logic. This includes: • Usage of Logging API (log4j)	**30**
Good amount of appropriate dataset to showcase project completely	**5**
Appropriate test cases using JUnit 4.0	**5**
Using MVC architecture and clean encapsulation of business logic in appropriate components. Judicious use of java beans.	**35**