JEE

Mini Project

Feedback Management System

(FMS)

Document Control

|  |  |  |  |
| --- | --- | --- | --- |
| ***Reference No:*** |  | | |
| Security Classification: | Internal | | |
| Issue Date: |  | | |
| Author(s): | **Name** | | **Title** |
| Uma Maheswari P | | Manager – T & D |
|  | |  |
| Reviewer(s): | **Name** | | **Title** |
| Hema G  Mahima Sharma | | Asst. Manager – T & D  Group Manager – T & D |
| Issuer(s): | **Name** | | **Title** |
|  | |  |
| Issuer Signature(s): |  | | |
| Distribution: | **Name** | | **Title** |
| Team | | All Designations in Training |
|  | |  |
|  | |  |
| Document History: | **Date** | **Revision** | **Change** |
| Oct 24 13 | 0.01D | Initial draft |
| June 29,15 | 0.02D | Changes in Miniproject document content as per the upgraded courses |
|  |  |  |

Table of Contents

[1 Introduction 4](#_Toc360528891)

[1.1 Setup Checklist for Mini Project 4](#_Toc360528892)

[1.2 Instructions 4](#_Toc360528893)

[2 Problem Statement 5](#_Toc360528894)

[2.1 Objective 5](#_Toc360528895)

[2.2 Abstract of the project 5](#_Toc360528896)

[2.3 Functional components of the project 5](#_Toc360528897)

[2.4 Technology used 6](#_Toc360528898)

[3 Implementation in J2EE LOT 7](#_Toc360528899)

[3.1 Summary of the functionality to be built 7](#_Toc360528900)

[3.2 Guidelines on the functionality to be built 8](#_Toc360528901)

[3.3 Evaluation and assessment parameters 11](#_Toc360528902)

# 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.

## 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 9i client and access to oracle 9i server
* JDK 8
* Eclipse Luna
* Junit 4.0 ,log4j

## 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 **MiniProject**. 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

# Problem Statement

## Objective

Development of Feedback Management System (FMS)

## 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.

## 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** 
  + Maintain the skill set of faculties
  + Maintain the Training courses
  + View the feedback reports and feedback defaulters report
* **Training C-ordinators**
  + Creation/modification/deletion of the Training Programs
  + View the feedback reports and feedback defaulters report
* **Participants**
  + 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

* + Presentation and communication skills of faculty
  + Ability to clarify doubts and explain difficult points
  + Time management in completing the contents
  + Handout provided(Student Guide)
  + 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”

## Technology used

* + - *Front End :–* 
      1. Class with main() function
    - *Business Logic Components and Services :-* 
      1. Java Beans
    - *Databases:-*
      1. Oracle 9i

# Implementation in J2EE LOT

## 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 |  |  |

## 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 **(Duration: 10 hours)**
   1. Develop the following screens:
   2. **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.
   3. **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 Enrollment
* View Feedback Report

Participant Role :

* Feedback Entry
  1. **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

* 1. **Course Maintenance page:**

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

* 1. **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.

* 1. **Participant Enrollment Screen:**

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

* 1. **Participant Feedback Screen:**

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

* 1. **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 | Handout | 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 | Faculty | Feedback Scores not Provided | | | | |
|  |  |  | Name | Name | Pre &Com | Clarf doubt | Time Mngmt | Handout | Hw/Sw |
| 1 | 03/03 | XYZ | Student #1 | Faculty #1 | X |  |  |  | X |
| 2 |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |

* 1. 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

1. 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 |

1. Course: OOP & UML**(Duration: 5 hours)**
   1. Develop relevant Use case and Class diagrams for the **FMS**
2. Course: Core Java 8+ Developer Tools **(Duration: 14 hours)**
3. Documentation**(Duration: 2 hours)**
   1. Project Documentation: Document your project details (Duration: 1 hour 30 mins).
   2. Project submission: Submit your project with all the artifacts including the test cases & documentation (Duration: 30 mins).

## 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** |