JEE
Mini Project
University Admission System
(UAS)

Document Control

Reference No:			
Security Classification:	Internal		
Issue Date:			
Author(s):	**Name**		**Title**
	Shrilata Tavargeri		Manager – T & D
Reviewer(s):	**Name**		**Title**
	Training Team		Executive, Managers – T&D
Issuer(s):	**Name**		**Title**
	Dinesh Anantwar		Sr. Manager
Issuer Signature(s):			
Distribution:	**Name**		**Title**
	Team		All Designations in Training
Document History:	**Date**	**Revision**	**Change**
	Nov 24, 08	0.01D	Initial draft
	Nov 29, 08	0.02D	Review comments incorporated
	July 03,13	0.03D	Changes in Miniproject document content to standardize for 60 days training.
	June 29,15	0.04D	Changes in Miniproject document content as per the upgraded courses
	July 28,16	0.05D	Changes in Miniproject document content as per the upgraded courses

Table of Contents

1 INTRODUCTION

This document outlines a mini project for the JEE LOT. The project is to develop an online University Admission System (UAS). 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 JEE 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)
- Oracle 9i client and access to oracle 9i server
- JDK 8
- Eclipse Luna
- JUnit 4.0

1.2 INSTRUCTIONS

- The code modules in the mini project should follow all the coding standards.
- Create a directory by your name in drive **D**. In this directory, create a subdirectory **MiniProject<Groupcode>**. 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 PROBLEM STATEMENT

2.1 OBJECTIVE

Development of a University Admission System (UAS)

2.2 ABSTRACT OF THE PROJECT

This project is aimed at developing a University Admission System (UAS) for applicants. This system can be used to search a university program for which an applicant wishes to take admission; apply for a selected university program without any login and fill up the form, administration staff can add/update/delete any program that university offers. Members of admission committee have the responsibility to filter the candidates for a specific program on the basis of application data and interview. There are features like report generators etc in this system.

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 three categories of people who would access the system viz. applicants, **m**embers of **a**dmission **c**ommittee (mac) and administrators. Each one of them would have some exclusive privileges (for e.g. Applicants can apply for a program by filling up the form without any login process, members of admission committee alone will be able to view applicant details and filter the applicants for a specific program they have applied for and only the administrator has the right to keep track of the university's program details.)

1. Applicant should be able to
   - View all programs scheduled by the university
   - Apply online for a scheduled program of the university, by filling up the application form that auto generates the application ID
   - View the application status, based on the application ID

2. The **m**ember of **a**dmission **c**ommittee (mac) should be able to :
   - Login into the system using his/her credentials.
   - View applications for a specific program.

- Accept/Reject an application on the basis of the details of the applicant. If accepted, fill in the scheduled date for an interview of the applicant before confirming the applicant to take the program.
- After the interview, update the status of the application to Confirmed/Rejected

3. The administration should be able to

   - login to the system using his/her credentials
   - Update and manage (add or delete) information of the programs offer by the university
   - Manage (add or delete) schedules of the programs offered by the university
   - Generate various reports like:
     - View List of applicants confirmed/ accepted (waiting for interview)/rejected for a scheduled program.
     - View list of programs scheduled to commence in a give time period

Transition of Status: Applied ->Accepted/Rejected->Confirmed/Rejected

2.4 TECHNOLOGY TO USE

- *Front End & Web Components:–*
  1. Java Classes would be designed for this Purpose.
- *Business Logic Components and Services :-*
  1. Java Beans
- *Databases:-*
  1. Oracle 9i

3 IMPLEMENTATION IN JEE LOT

3.1 SUMMARY OF THE FUNCTIONALITY TO BE BUILT:

The participants need to develop the **UAS** by building the functionality incrementally in each of the course modules of JEE 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	Web Basics (HTML, JavaScript, XML)	4.5	1	
3	Oracle Basics	4	1	Creating relevant database tables
4	OOP & UML	1.5	1	Creating relevant Use case and class diagrams
5	Programming Foundation with Pseudo code+ Web Basics +Oracle Basics	1		
6	Core Java 8 & Development Tools(Junit, Log4j))	10	2	Developing Business components (java classes). Coding for test classes & testing the functionality using JUnit
7	Core Java 8 & Development Tools+ Dev Tools + OOP/UML Test	1		
8	Servlets	3.5	2	Developing the application using the prototypes. Converting the HTML web pages to java classes
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 JEE LOT is as follows:

: Here screen refers to the Console Screen

1. **Course: Core Java (Duration: 10 hours)**
   a. Develop the following screens:
      i. View Scheduled Programs screen: This page provides the applicant with information related to programs scheduled by the university, this screen has a link to 'Apply Online'
      ii. Apply For a Course : Allows an applicant to fill in and submit application form for a scheduled program, an applicant ID is auto generated at DB level.

iii. View Application Status: An applicant should be able to view the status of application by entering the Applicantion_id

iv. Maintain Information on Programs Offered and Scheduled: This functionality should be available only for authorized administrators. Administrations maintain valid information about programs offer and scheduled by the university (add new / delete existing). Update only programs offered information and updatable data includes Program description, applicant eligibility.

v. View Applications for Scheduled Program Screen: This screen should only be available to authorized members of admission committee (mac). She/he must be able to see the report based on id of a scheduled program.

vi. Filter Applicants Screen: This screen should only be available to authorized members of admission committee (mac). She/he must be able to view the participants for a specific scheduled program and Accept/Reject the application, update the application status accordingly. If the applicant is accepted for the program, an interview date should be entered. If the applicant's interview is conducted, the status should be changed to either Confirmed or Rejected. Any change in the status of an application should result in an email being sent to the applicant notifying him/her the details of change in status.

vii. View information of Programs Scheduled to Commence in a given time period: Provide details of programs scheduled to commence within the given time period, start and end dates should be accepted, such that start date is less that end date.

viii. View list of applications Accepted/Rejected/Confirmed for a particular Scheduled Program: This should display details of applicants Accepted/Rejected/Confirmed for a particular scheduled program.

2. Course: Oracle (Duration: 5 hours)

a. Create the following database tables:

i. Application: This will contain the list of valid applications (Applicants fill in the application form)

ii. Programs Offered: This will contain details of programs offered by the university

iii. Programs Scheduled: This will contain details of a programs scheduled by the university, among the offered programs

iv. Participant : This will contain details of confirmed participants for a scheduled program

v. Users : This will contain all the valid logins and passwords, there is no screen to enter values in this table, assume that this table is present with valid data, that will be used by the application

b. The structure of the above listed tables is as follows:

i. Application : Application_id (auto generated serial no.),full_name (varchar2(20)), date_of_birth (date), highest_qualification (varchar2(10)), marks_obtained (number), goals(varchar2(20), email_ id(varchar2(20)), Scheduled_program_id(varchar2(5)), status(varchar2(10)),Date_Of_Interview(date)

Note: Possible values of status are 'applied' by default, 'accepted' ,'rejected' and 'confirmed'

ii. Programs_Offered: ProgramName (varchar2(5)), description (varchar2(20)), applicant_eligibility(varchar2(40)) , duration(number), degree_certificate_offered(varchar2(10))

iii. Programs_Scheduled : Scheduled_program_id (varchar2(5)), ProgramName (varchar2(5)), Location (varchar2(10)), start_date (date), end_date (date), sessions_per_week(number)

Notes:

- Location can be normalized since it would contain further information like city, state, zipcode etc
- No. of seats for a program are not limited

iv. Participant : Roll_no (varchar2(5)), email_id (varchar2(20)), Application_id (FK) , Scheduled_program_id(varchar2(5))

v. Users : login_id(varchar2(5), password(varchar2(10)), role(varchar2(5))

Notes: Possible values of role are 'admin' and 'mac'

3. Course: OOP & UML (Duration: 5 hours)

a. Develop relevant Use case and Class diagrams for the **UAS** application.

4. Course: Core Java 8 & Development Tools (Duration: 14 hours)

a. Develop business components (java classes) for the following functionality:

i. Login verification: This component will authenticate the user. Verify if the user who is trying to access the system is a valid user. This verification is as against the valid users listed in the users table, which has valid logins for administrators and members of admission committee. There is no login required for an applicant.

ii. Retrieve Programs Scheduled: This component provides the applicant with information on all programs scheduled by university.

iii. Apply Online: This component should allow an applicant to submit his application for a scheduled program only after performing obvious validations on the entered data. An application ID is auto generated at DB level and shown to the applicant.

iv. Retrieve Application Status: An applicant should be able to view the status of application by entering the Application id

v. Maintain Program Information: This is a typical CRUD (Create/Read/Update/Delete) operation. This functionality should be available only for authorized administrators. Administrations maintain information includes Programs Offered (add/update/delete) and Programs Scheduled (add/delete), only valid operations should be performed on the data. For Ex. A program scheduled cannot be deleted /updated if there are applicants for it.

vi. Retrieve Applications for Scheduled Program: This should only be available to authorized members of admission committee. He/She must be able to view all the applicant data based on Scheduled program id

vii. Filter Applicants: This component updates status of an application and enter the date of interview only upon accepting an application. This should only be available to authorized members of admission committee. He/She must be able to view the participants for a specific scheduled program and Accept/Reject the application, update the application status accordingly. Status can be changed to 'Confirmed' only after, the previous status was 'accepted' and the date of interview is present, dated previous to or same as today's date. If the applicant is accepted for the program, an interview date should be entered. If the applicant's interview is conducted, the status should be changed to either Confirmed or Rejected.

viii. View information of Programs Scheduled to Commence in a given time period: Available to administrator retrieve details of programs scheduled to commence within the given time period.

ix. View list of applications Accepted/Rejected/Confirmed for a particular Scheduled Program : Available to administrator retrieve details of applicants Accepted/Rejected/Confirmed for a particular scheduled program

b. Develop test classes for testing the following functionality

i. Login

ii. Apply for a program

        iii. View list of applications

    c. Test the application using JUnit.
    d. Configure Logger to log the status of an application

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. 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 and Feel	**10**
Client-side validation	**15**
Code Documentation and using coding standards	**15**
Functionality	**40**
Dataset	**10**
Appropriate test cases using JUnit 4.0; ;logging application using Log4j	**10**