Case Study – Banking Application System

1. Business Scenario

A Prestige Banking System, offers a wide variety of loans and services for their customers which has a requirement of having an online net banking, where customers can see their account activities online instead of visiting the bank branch, do online transactions instead of using cheque’s to save the customers time.

Moreover customers would come to know about the new changes and offers the bank make that could be the rate of interest and charges apply for receiving message or having debit card and credit card. As the details of the customers and their accounts has to be maintained in the database and they have to access their accounts through the GUI the team should use Oracle Database and Java/JEE.

The Application will have web pages that would be developed using HTML/JSP, the team has to create a home page so that as soon as the customer accesses the application the home page must be displayed through that home page he can visit other pages that could be Investors page, contact us page and etc. As the application interacts with the database the data should be maintained not only in the database it should also be maintained in a reusable component java beans, the team could create the java beans with respect to the table they have in the database. For Instance, If customer wants to login he needs to enter his id and password so that these values should be checked in the database and also the customer names would be maintained throughout the page so that a java bean can be created that will have customer name as the property which will be set as soon as the customer log’s in and that name will be retrieved at every page where the name should be displayed till he logs out, once he log’s out this java bean can be removed from the session.

Oracle Internal & Oracle University Use Only

The application should allow customer to access various information’s like what all the transaction’s he has done with the date. The customer could able to change his log in password and transaction password, If he wishes to change the password the application should allow him to change passwords which are not used previously i.e., older passwords must not be accepted by the application. Customer should able to transfer the amount from one account to another account while transferring the amount the application should ask all the required information of the destination account which are mentioned in the Student Activities. When user tries transferring the amount the application should ask him to enter his transaction password, failing to which the amount should be prevented to credit to the destination account.

The application must have other pages that can be accessed without logging in like contact us if the user goes to this page he should see the details and address of the bank, an investor page should have the investors list on the bank, the application should also have section on the pages which would have set of images randomly changing. Student has to create all the required tables that are mentioned in the problem statement and implement the application according to the business scenarios, the implementation details are mentioned in the Activities to be done by the student section.

Problem Statement

**2.1 Assumption**

We assume that there is right now only one bank and customer’s details is already stored with some amount of money in their account. Firstly design a database table based on the following information’s. Create primary key, foreign key references and other constraints wherever necessary.

**2.2 Points to focus**

**Bank:** Contain details about the bank id, bank name, IFSC, branch and number of customers.

Oracle Internal & Oracle University Use Only

**Customer:** Contains details about the customer id, name, age, login password and bankid (customers would use customer id and login password for login, whereas bank id is used at the time transactions)

**Account:** Contain details about the account number, customer id, account type (savings/current / salary), IFSC, transaction password, account creation date and time, transfer limit and minimum balance.

**Address :** Contains customers address details about address id, street name, state, city, pin and customer id.

**Transaction Details:** Contain transaction details about transaction id, reference number, account number of the customer, transaction type (credit/debit), transaction date and time, bank id and IFSC. (Reference number will be duplicated as it will be shared by sender and receiver both).

**Password Details:** Contain the password information about password id, customer id, account number, old login password, new login password, login password creation date and time, new transaction password, old transaction password and transaction password creation date and time.

**Create a Trigger** : Create a trigger with a name bank\_trigger on bank table and write **PLSQL** which should update the number of customers in the bank table once the customer details are added to the customer table.

**2.3 Business process in the application and use cases**

**Note:** Team would implement using Java/JEE, HTML, CSS and JavaScript.

After entering the **URL** the application should take the user to the home page which would have some **links** like **home, investors, about us and contact us** these links should be common for all the pages to a user who can visit after he clicks on any of the link (ex. If he clicks on investors it should show number of investors and their details as well as the page should also show the links as explained above). Here the user can be guest or customer they could be able to see the pages without **logging in.** Team has to provide information on all the pages which are linked.

Oracle Internal & Oracle University Use Only

1. The home page should also have a separate section for **displaying images** regarding some advertisements and offers which should be keep changing after some seconds these must be common in all other linking page from home page but after **customer logs** in these links and images should not appear.

Oracle Internal & Oracle University Use Only

1. If a Customer **visits the home page** he should have a **login section** where he should be able to login by entering **valid customer id and login password**, failing to which he should be redirected to the same login page with a proper error message. After **successful authentication** he should be able to see an authenticated page which will have his **name, account number, account type and available balance.** On the same page he should also have **links for Account Activities, Transfers, Change Password and logout.**

Oracle Internal & Oracle University Use Only

1. On clicking on Account Activities he should get a page which should have all the transactions he has done in a table like transaction id, account number, amount, transaction type, reference number and date and time of the transaction. Team has to fetch these details from the corresponding tables and their columns of database table.
2. On clicking on **Transfers** he should get a page which should display a form that should accept user input like **destination account number, name , IFSC code and amount,**  once you click on transfer button it should **ask customer** to enter **transaction password** if its incorrect then error must be displayed and allowed to enter the transaction password again, if password entered is correct then it should be credited to the destination account and also application/database should be able **to generate a sequence number for transaction id and reference number** whose value should be stored in a transaction table’s transaction id will be unique but **reference number will be common for source and destination account.** These details present in the **transaction table** are useful when customer wants to see his account activities. Once the amount is transferred to anyone’s account the amount in the **destination account** table should also be updated as a total amount the account has. Store the date; bankid and IFSC when the transaction happens accordingly (make use of the other table information’s). The transaction table just maintains the **transactions** done by all the customers this table is used by the application properly when customer wants to check his transaction details.
3. On clicking change password application should allow customer to change either login/transaction password or both

Oracle Internal & Oracle University Use Only

**Note:** Application should not allow customer to store old passwords, make use of the password table to track the old and new one

1. On clicking on logout customer should be logged out
2. If customer is inactive for 2 minutes session should be automatically terminated

**Note: make sure that links like account activities, transfers, change password and logout are visible in all the pages you visit after the log in.**

Tools to be used:

* Oracle Database 10g
* JDK 1.7
* Eclipse
* Apache Tomcat
* Angular 4

Activities to be done by the students

Students should first create all the tables like Bank, Customer, Address, Account, Transaction\_Details, Password\_Details and apply the relationships from one table to another. E.g., account number column of Transaction\_Details table would reference to the account number column of the Account table so that when customer needs to find out the account activities he should get all the transaction done.

Note: Ensure that primary key constraint is given for all the tables

After creating the tables and their relationships they have to store some records to each table (at least 5 records), Students should also create a trigger with a name bank\_trigger on bank table and write **PLSQL** which should update the number of customers in the bank table once the customer details are added to the customer table. Once all the database tasks are over students can concentrate on the next module.

Student are going to create lot of web pages here, firstly they have to configure an home page for the application so that as soon as they access the application through an URL it should show this home page, the home page will have following components

An heading at the top which displays the bank name and footer at the bottom of the page, heading and footer should appear on all other pages the user visits.

Below the heading a common navigation links like **Home, Login, Investors, About Us, Contact Us.** These navigation links should appear on all other pages the user visits, ex if he visits Investors page user would be directed to Investors page there user should able to see the heading, footer and navigation links, same is applied for Login About Us and Contact Us pages

A section that will show some images(width and height of the images should be small) which would be randomly changing these images can be for advertisements like some offers the bank provides, interest rates, its success rate, their new products and etc.

When user clicks on the login link a login page should be displayed which should take customer id and password

Oracle Internal & Oracle University Use Only

If the user clicks on the Login link then the application should render a login page and should do the following

The login page should ask a user to enter customer id and login password, If customer enters valid customer id and login password then the application should direct him to the success page by checking id and password in the database table **Customer**, else an error message should be displayed saying invalid credentials and should allow him to login again by asking id and password below the error message. To check customer id and password use the DAO

**Note: Students should use DAO whenever they have to execute any business logics like checking user credentials, checking customer activities, transactions, change password etc. and also java beans whenever they have to use the values entered from the client side with the values fetched from the database table.**

While reading customer id and password these values should be stored in a java bean so that you can reuse these values across the multiple pages through, but only if a valid id and password is entered a java bean should be stored in the session so that across the pages you can use this java bean which will have id and password in it.

If the user tries to login with empty customer id and login password a warning should be displayed saying the particular field can’t be empty, the input validation should be done using client side technique

Upon successful login a success page should contain the following

Customer details at the middle of the page displaying his/her Name, Account Number, Account type and Balance in a proper table format.

Customer details and Account details are retrieved by using customer id and account number from Customer and Account tables respectively, as the session contains a java bean having customer id and password student can use this id and access the corresponding account number in account table and retrieve the details present in it.

A common navigation link above the customer details in the success page which will have links such as Account Activities, Transfers, Change Password and Logout

These common links are displayed across multiple pages like Account Activity, Transfers and Change Password

If the user clicks on **Account Activities** he should get a page which **should have all the transactions** he has done in a table like **transaction id, account number, amount, transaction type, reference number and date and time of the transaction.** Team has to **fetch these details from the tables** Account and Transaction\_Details, Students have to use a java bean if required to store account details and transaction details which are fetched from the database table to the respective attributes through the setters and getters methods. for e.g., a java bean called Account and TransactionDetails would have attributes matching to the Account and Transaction table columns and both the java beans will have setters and getters methods so that it would be reused while displaying these values through JSP/JSTL/JSF

Oracle Internal & Oracle University Use Only

Upon clicking on **Transfers** he should get a page which should display a form that should accept user input like **destination account number, account holders name , IFSC code and amount,**  once you click on a transfer button store these values in a java bean (student has to create a separate java bean for storing the above information’s) and it should **ask the source customer** to enter **his customer id and transaction password** which should also be stored in a separate java bean so that these values can be reused wherever necessary, if its id and passwords are incorrect then error must be displayed and allow customer to enter the transaction password again, check for input validation and display appropriate warning message when customer tries to transfer amount with empty customer id and password, if customer id and password entered are correct then it should be credited to the destination account and also application/database should be able **to generate a sequence number for transaction id and reference number** whose values should be stored in a transaction table’s transaction id will be unique but **reference number will be common for source and destination account (**e.g., if a customer having account number 123 credits 10000Rs to 456 account number then the transaction\_details table will have 2 record entries having unique transaction id for 2 records but common reference id for the 2 records)**.** These details present in the **transaction table** are useful when customer wants to see his account activities. Once the amount is transferred to anyone’s account the amount in the **destination account** table should also be updated as a total amount the account has. Store the date; bankid and IFSC when the transaction happens accordingly (make use of the other table information’s).

If customer clicks on change password then a page with two links to be rendered which must have **Change Login Password** and **Change Transaction Password** link, make sure the common navigation links are still present in these pages as well, if customer clicks on either of the link then 3 text fields should be rendered asking him to enter old password in the first text field, second text field should take enter new password and third text field should take confirm new password. Following requirements to be met while changing either of the passwords i.e., for Login and Transaction passwords : -

* + - A warning message if the text fields are empty submitted
    - **Old passwords should be first checked in the database table whether its matching to logged in customer id or not,** if not an error message should be displayed saying old password is not matching
    - **New password should match the following criteria like minimum 8 characters in length, in that at least one special character like @, \*, $, one number, one upper case letter,** failing to which an error message should be popped to client asking him to enter the password again

Oracle Internal & Oracle University Use Only

* + - **New password’s and confirm passwords should match** failing to which an error message should be popped
    - **Students should also prevent customers entering any of his previous passwords for both login and transaction passwords;** use the Password\_Details table to check all his previous passwords as it has details like customer id, account number, old password, new password, date and etc.
    - Students have to implement these business rules for login password and transaction password both
    - Once the change password has met all the above conditions then old password and new password to be stored in the Password\_Details
    - **Students should also allow customer to change either of the password** i.e., if customer wants to change only login password but not transaction’s then care should be taken to update only login password by creating a new record in the Password\_Details table by updating only login password and retaining the current transaction’s password as it is same is applied if customer wants to change only transaction password.
    - **Password\_Details table** will have columns like old login password and old transaction password which would be initially empty for the new customers since when the account is created newly he can have only one password which itself will be new.

If customer clicks on logout then session should be terminated and all the object/data/JavaBeans stored in a session should be removed

**If customer is inactive for 1 min then session should be automatically terminated** this should be done once the customer is logged in.

**Once the session is terminated and user tries to access any of the URL related to account details, transfer or change password** a error page asking him/her to login first to access this page should be rendered along with customer id and login password form so that user can enter id and password to login.

**Note: Students has to use java beans and DAO whenever they found some values/object properties should be checked in the database and pass the values/objects to another page or when the values has to be reused, store the records to the proper columns and tables.**

Guidelines to the instructors:

**Step 1**

After teaching them the HTML, CSS, JavaScript, SQL/PLSQL, Java, JDBC and JEE, let student create all the required tables for the project. It might go in cycle and the facilitator helps the participants to design tables by giving the comments and feedbacks.

Oracle Internal & Oracle University Use Only

**Step 2**

Once the basic design is approved by the facilitator, the team can go ahead and create a web project in IDE, components to be monitored

Creation of Tables and constraints

Inserting the records to the table

Designing View components

Navigating from one view to another

Creating Java Beans and DAO for the database tables

Creation of Servlet for reading the user data and rendering the proper View

Input Validation

**Step 3 – Role Play**

Let the team present their project.

Note Down comments/ideas by other team and facilitator

Keeping these ideas as base, team has to list down the queries that can facilitate the departmental store.

**Step 4 – Validate**

All learning objectives defined above for the project should have been met. If not revisit and ensure the learning objectives are met for the project

**Step 5 - Extension of Project work**

If student completes the above task well in advance trainer has flexibility to give additional work to give the existing components and line. Create a separate link called Add Beneficiary in the authenticated page for a customer where he can store the beneficiary details like Account Number, IFSC, Bank name, Branch name, Phone number and Address. Populate an unique beneficiary id for every beneficiary customer adds and maintain these details in a separate table called Beneficiary so that whenever customer needs to transfer the amount he can directly enter the beneficiary id and click on submit so that all the required details for transferring amount should be automatically filled in the transaction page like destination account number, name, IFSC and etc, except the amount to be transferred since you are allowing customer to decide how much amount to be transferred.

Do a validation on this activity as well so that if a beneficiary id is not found appropriate error message should be popped to the customer.

**Step 6** – Pre Assessment and Post Assessment during the workshop

Oracle Internal & Oracle University Use Only

Reference Material:

Java Documentation :

<http://docs.oracle.com/javase/tutorial/>

Java EE Documentation :

<http://docs.oracle.com/javaee/7/tutorial/>

Estimated duration for completing the project:

12 hrs