**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
* Analyze 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 analyzing various reports generated through code quality tools
* Automate the build using Maven

**Note:**

1. Do not change the table structures.
2. Use appropriate data types and precisions for the variables
3. Handle all exception and alternate flows
4. Use generics wherever applicable for parameters and return type
5. Write at least three test cases (both positive and negative/exceptional) for service class
6. **Loose coupling:** Create a separate Dao interfaces and implementation classes. Database interaction code should be only in DAO classes. Dao classes should not have UI code.

# Problem statement: Ticket Tracking System

Design and implement an internal ticket tracking application for a software company. It will be used to log a ticket, to close a ticket and to view tickets with the turnaround time taken to resolve a ticket.

Database design with sample data is listed below. Do not add/ remove columns to this table, create the tables “EMPLOYEE” and “TICKETS” as listed.

**EMPLOYEE table: MID is Primary Key**

|  |  |  |  |
| --- | --- | --- | --- |
| **MID** | **EMPLOYEE\_NAME** | **HIRE\_DATE** | **DEPT** |
| M100100 | Karthik | 2004-1-10 | BIZC |
| M100101 | Swetha | 2011-12-12 | HiTech |
| M100102 | James | 2010-12-12 | ISSD |
| M100103 | Rahul | 2012-2-22 | ISSD |
| M100104 | Zaheer | 2012-3-19 | T&T |

**Note:** Create a Stored Procedure to insert records into **‘EMPLOYEE’** table. [Provide database scripts file for creating stored procedure and inserting records].

**TICKETS table: Ticket\_ID is Primary key, LOGGED\_BY and RESOLVED\_BY are foreign keys referencing MID of Employee table**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TICKET\_ID** | **LOGGED\_BY** | **RAISED\_DATE** | **SEVERITY** | **TICKET\_DESC** | **RESOLVED\_BY** | **RESOLUTION** | **RESOLVED\_DATE** | **STATUS** |
| 1 | M100101 | 2012-10-3 | Major | AppV not working | M100103 | Need to restart with LAN cable | 2012-10-4 | CLOSED |
| 2 | M100100 | 2013-7-10 | Critical | Laptop restart problem | NULL | NULL | NULL | OPEN |

The home page contains the following links:

* **“Log a Ticket”**
* **“Close a Ticket”**
* **“View Turnaround time”**

|  |  |  |
| --- | --- | --- |
| **Use case #1** | | **Log a ticket** |
| Trigger | | User clicks on “Log a Ticket” hyper link on the home page. |
| Pre-Conditions | | |  | | --- | | The table containing the details of “EMPLOYEES” should be created and pre-populated with a set of values manually from the back-end. | |
| Post-Conditions | | |  | | --- | | The complete information about employee’s ticket is stored in the application database. | |
| UI Screen details | | The input form contains:   * Employee: dynamic dropdown combo box [ select] displaying list of employees which are present in the database * Ticket Date-Time: Text box to accept Date-Time in the format [DD-MM-YYYY HH:MM] * Severity: dropdown combo box [select] displaying “Critical”, “Major” and “Normal” * Ticket Description: Multi line text box. * Submit button * Cancel button |
| Main flow | | |  | | --- | | User provides the required details in the form and clicks the “Submit” button. On successfully saving the details, A unique ticket identifier is generated. User is redirected to the home page and the following message is displayed on the Home Page: “Ticket: <Ticket-No> is submitted successfully”. [ Ex: Ticket 5 is submitted successfully] | |
| Data validations | | All the form fields are mandatory  Ticket Date-Time should be in the specified format  Ticket Date-Time should be earlier than the current date-time |
| Business rules | | |  | | --- | | An employee can have more than one ticket logged in the system. | | |  | | --- | | The list of employees does not include employees from the ISSD department. | | |
| Alternate flow | | User chooses to abort the “Log a Ticket” operation by clicking on the “Cancel” button, he should be redirected to the home page |
| **Use case #2** | | **Close a Ticket** |
| Trigger | | User clicks on “Close a Ticket” hyper link on the home page. |
| Pre-Conditions | | |  | | --- | | A ticket has been logged in the application database. | |
| Post-Conditions | | The status of the ticket is changed from “Open” to “Closed”. |
| UI Screen details | | The input form contains:   * Ticket ID: dynamic dropdown combo box [select] displaying list of tickets which are not closed [present in the database]. * Resolved by: dynamic dropdown combo box [ select] displaying list of employees from “ISSD” team which are present in the database * Resolution: Multi line text box. * Submit button * Cancel button |
| Main flow | | |  | | --- | | User provides the required details in the form and clicks the “Submit” button. On successfully updating the details, User is redirected to the home page and the following message is displayed: “Ticket: <Ticket-No> is closed”. [ Ex: Ticket 5 is closed] | |
| Business rules | | |  | | --- | | The current system timestamp is recorded as the time of closing the ticket | |
| Alternate flow | | User chooses to abort the “Close a Ticket” operation by clicking on the “Cancel” button, he should be redirected to the home page |
| **Use case #3** | **View Tickets with the Turnaround Time** | |
| Trigger | User clicks on “Close a Ticket” hyper link on the home page | |
| Pre-Conditions | |  | | --- | | A ticket has been closed in the application database. | | |
| UI Screen details | The details all tickets, with the turnaround time [total time taken between the submission and resolving] are displayed in the exact format given below [Given below is just a sample data]:    **Note:** The above report displayed should also be saved into a file for future reference | |