ASP.NET MVC

Duration: 6 hours

Ticket Tracking System

Design and implement an internal ticket tracking application fora 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: EID is Primary Key

EID	Employee_Name	Hire_Date	Dept
E100100	Venkat	2004-1-10	MGM
E100101	Krishna	2004-1-10	MGM
E100102	Chandrashekhar	2005-3-11	DEV
E100103	Saheer Ali Khan	2008-10-13	DEV
E100104	Shashikanth	2007-2-17	DEV
M100103	Avinash	2007-3-10	DEVOPS
M100105	Ashok	2008-6-18	DEVOPS

TICKET table: TICKET_ID is Primary Key

TICKET_ID	LOGGED_BY	RAISED_DATE	SEVERITY	TICKET_DESC	RESOLVED_BY	RESOLUTION	RESOLVED_DATE	STATUS
1	E100101	2012-10-3	Major	App server not working	M100103	Need to restart with LAN cable	2012-10-4	CLOSED
2	E100104	2013-7-10	Critical	Laptop restart problem	NULL	NULL	2013-7-11	OPEN

Implement Use case #1 and Use case #2 using ASP.NET MVC and Entity Framework.

The home page contains the following links:

"Log a Ticket"

"Close a Ticket"

Use case #1	Log a ticket
Trigger	User clicks on “Log a Ticket” hyperlink 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 redirected to the home page and the following message is displayed: “Ticket ticketNo 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 DEVOPS 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” hyperlink 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 “DEVOPS” team which arepresent 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 homepage and the following message is displayed: “Ticket ticketNo 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/she should be redirected to the home page

Use case #3 View Tickets with the Turnaround Time

Execute this as a Standalone application

A ticket has been closed in the application database table “tickets” with the status “CLOSED”.

Create and host a WCF service which exposes a function “**ViewTickets”** to get the details of all the “CLOSED” tickets. Create a client of your choice either web client or console client to display the closed tickets as below:

The details of 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]:

Employee Name	Ticket	Severity	Turnaround Time (in hours)	Description	Resolved By
Krishna	**1**	Major	**6.5**	App server not working	Avinash
Shashikanth	**2**	Critical	**22.5**	Laptop restart problem	Avinash

Marks allocation:

#	Description	Marks (50)
1	**Log a ticket:** on submitting the form, data is inserted into "tickets" table	6
2	**Log a ticket: Message "**Ticket ticketNo is submitted successfully" is displayed on home page after successfully inserting a ticket	2
3	**Log a ticket: Validation** a) All the form fields are mandatory. If form is submitted with incomplete values, the operation is aborted, and the following message is displayed on the screen. Example: "The value is not provided for the field <ticket description| ...>".) b) Ticket Date-Time should be in the specified format, else display error message "Invalid Ticket Date-Time" c) Ticket Date-Time should be earlier than the current date-time, else display error message "Ticket Date-Time should be earlier than current date-time"	2 2 3
4	**Log a ticket: Alternate flow: Clicking on cancel button redirects the user to home page**	1
5	**Log a ticket: Business rules->** The list of employees does not include employees from the DEVOPS department.	3
6	**Close a ticket: on submitting the form;** The status of the ticket is changed from "Open" to "Closed" and the current system timestamp is recorded as the time of closing the ticket.	6
7	**Close a ticket: Business rules->** The list of employees includes employees only from the DEVOPS department.	3
8	**Close a ticket: Fetching ticketid's from database and** displaying list of tickets which are not closed.	3
9	**Close a ticket: Alternate flow: Clicking on cancel button redirects the user to home page**	1
10	**Creating WCF service to view tickets with the operation name "ViewTickets"**	7
11	**Hosting the WCF service**	4
12	**Creating Client to View Tickets with the Turnaround Time**	3
13	**Configuring the WCF service for fault exception**	4