Project Report

|  |  |
| --- | --- |
| Project Title | Problem Manage a Server Outage Scenario & an Issue & Change Request Management System |
| Qualification Name (NICF) | Advanced Certificate in Software Applications (Development and Deployment) |
| Product Name |  |
| Module Name (NICF) |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Student name | | Assessor name | |
| Muhammad Kemal | |  | |
| Date issued | Completion date | | Submitted on |
|  |  | |  |
|  | |  | |
| Project title | Student Registration Form Development | | |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature: Date: 18 August 2023 |

Table of Contents

[1. Project Background 2](#_Toc143257926)

[2. Project Objective 3](#_Toc143257927)

[3. Project Requirements Specifications 4](#_Toc143257928)

[3.1 Functional requirements 4](#_Toc143257929)

[3.2 Non-functional requirements 5](#_Toc143257930)

[4. Task 1: Principles of Problem Management 6](#_Toc143257931)

[4.1 What is Problem Management? 6](#_Toc143257932)

[4.2 The Three Phases of Problem Management 6](#_Toc143257933)

[4.3 Principles of Problem Management 7](#_Toc143257934)

[4.4 Problem Management Example 8](#_Toc143257935)

[5. Task 2: Tools, Process & Technologies 9](#_Toc143257936)

[1. Problem Logging 9](#_Toc143257937)

[6. Task 3: Investigate & Diagnose 10](#_Toc143257938)

[7. Task 4: Explain Prioritize 11](#_Toc143257939)

[Priorities and categories problems: 11](#_Toc143257940)

[8. Task 5: Problem Management Solution 13](#_Toc143257941)

[9. Task 6: Systems will be implemented 13](#_Toc143257942)

[10. Task 7: Best Practices in Problem Management 14](#_Toc143257943)

# Project Background

Developing a Software Developer's Community Portal

ABC Jobs Pte Ltd is embarking on a transformative project to establish a dedicated community portal for software developers akin to LinkedIn. This initiative aims to create a hub where programmers and software administrators can seamlessly connect, exhibit their skills, explore job prospects, and collaborate on projects. This project aligns with the framework of the NICF-Capstone Project using Java, leveraging knowledge from previous modules, including UI Framework, Database Design, and Web Development Foundation.

The project's overarching objectives are twofold: firstly, it requires participants to design, plan, and rigorously test the community portal developed during Module 5, showcasing their ability to synthesize skills acquired throughout the course. Secondly, the project addresses the recurrent downtime experienced by ABC Jobs Pte Ltd's existing community website. By implementing problem management principles and utilizing cutting-edge tools and technologies, the project aims to resolve these issues and establish a sustainable solution. The project endeavors to enhance incident and enhancement management practices through meticulous documentation, adherence to best practices, and nurture a dynamic software developer community.

The development of this community portal underscores the intersection of technological proficiency and real-world problem-solving. It provides a tangible platform for software professionals to collaborate and flourish and exemplifies the participants' capacity to translate theoretical knowledge into practical solutions that have a lasting impact on a vital industry.

**Scope of the project**

* Explain the principles of problem management.
* Use a variety of tools, processes and techniques to identify problems.
* Conduct investigation, analysis and resolution.
* Explain the steps to investigate and diagnose problems.
* Priorities and classify change requests.
* Prepare a solution to address the root cause of the problem.
* Document and monitor problems
* Explain best practice for documenting problems.

# Project Objective

The goal of the project is to use a problem-management framework to efficiently manage and address concerns relating to the community site. To improve the website's stability and performance, this entails locating, analyzing, and fixing issues. The objective is to develop a strong problem documentation process, prioritize issues, offer solutions, and streamline incident and change request management.

**Tools & Platforms Used**

a. **IntelliJ IDEA**

* Purpose: Used as the primary integrated development environment (IDE) for developing Java-based software tools.
* Function: Provides a comprehensive platform for coding, debugging, and testing Java programs, ensuring efficient development and maintenance of the community portal.

b. **Excel Sheet**

* Purpose: Utilized for issue tracking and maintaining a known error database.
* Function: Enables the team to log and manage issues, track their status, and establish a repository of common errors, facilitating efficient troubleshooting and resolution.

c. **Microsoft Word**

* Purpose: Employed to create the Problem Management Report.
* Function: Allows the creation of detailed and structured reports that document the problem management process, including the identification, analysis, and resolution of issues related to the community portal.

d. **Figma**

* Purpose: Utilized to create visual diagrams, For Brainstorming, for root cause analysis.
* Function: Enables the creation of clear and organized diagrams that help analyze the root causes of problems. Specifically, fishbone diagrams aid in identifying underlying factors contributing to issues in the community portal.

Utilizing these tools and platforms, the project aims to efficiently develop, manage, and document the problem management process, leading to an improved and more reliable community portal for software developers.

# Project Requirements Specifications

## ****3.1 Functional requirements****

**Users (Software Programmer)**

**Register:** New Software Programmers can create an account and register on the community portal.

**Login:** Software Programmers can log in to their accounts using their credentials

**Logout:** Software Programmers can log out of their accounts to end their session.

**Posting Thread: Software Programmers should be able to post on their dashboard**

**Comment on Thread: Software Programmers should be able to comment on threads.**

**Search users:** The Software programmers can search for other users by parameters like First Name, Last Name, City.

**View Other Users: Software Programmers can view other profiles after searching it.**

**Update profile:** Software Programmers can edit and update their profile information.

**Apply Jobs: Software Programmers can apply job that already post by Admin.**

**Administrator**

**Login:** Administrators can access their accounts with their login information.

**Add user:** To end their session, administrators can log out of their accounts.

**Delete user: admin can delete user data that has registered on the portal.**

**Modify user:** The admin can update the data of registered users on the portal.

**Send Bulk Email: The admin can send Bulk Emails to the users in the community portal.**

**Post Jobs: The admin can post the jobs in the community portal.**

## ****3.2 Non-functional requirements****

**Safety Requirements:**

The community portal project has specific safety requirements due to the sensitive nature of user data:

1. **Data Privacy and Security:** Users' personal information is stored in the database, hence strong data privacy and security safeguards are required. To secure user data from unwanted access or breaches, ensure compliance with pertinent data protection legislation and put encryption and access controls into place.
2. **Audit Trails and Logging:** Implement comprehensive audit trails and logging mechanisms to track user interactions and system activities. This helps monitor and investigate potential security breaches or unauthorized access attempts.

**Performance Attributes:**

1. **Availability:** The community portal aims to achieve high availability to ensure uninterrupted user access. Implement load balancing, redundancy, and failover mechanisms to minimize downtime in case of server failures. Regularly conduct disaster recovery drills to ensure swift recovery during unexpected incidents.
2. **Maintainability:** The administrator should regularly monitor and maintain the community portal to ensure optimal performance. Schedule routine maintenance tasks, such as database backups, software updates, and performance optimization, to enhance the portal's stability and longevity.
3. **Usability:** The ABC community portal is designed to accommodate many users and a substantial amount of user information. To ensure usability:
   * **Scalability:** Design the system to handle increased user load and data growth. Implement horizontal scaling by adding more servers or resources as needed.
   * **User-Friendly Interface:** Develop intuitive and user-friendly interfaces that enable users to navigate portals, find information quickly, and perform tasks efficiently.
   * **Responsive Design:** Ensure the portal design is responsive and adaptable to various devices and screen sizes, enhancing the user experience across multiple platforms.
   * **Efficient Search and Retrieval:** Implement efficient search and retrieval mechanisms to enable users to access their desired information quickly, even as the database grows.
   * **Optimized Performance:** Utilize caching mechanisms, minimize database queries, and employ content delivery networks (CDNs) to optimize page load times and overall performance.

By addressing these non-functional requirements, the project ensures the community portal's safety, performance, availability, maintainability, and usability. These requirements contribute to delivering a reliable and user-friendly experience for administrators and users of the ABC community portal.

# Task 1: Principles of Problem Management

## 4.1 What is Problem Management?

## In IT service management, problem management is a proactive procedure with the goal of locating, examining, analyzing, and resolving the underlying causes of incidents and problems. The emphasis is on dealing with the underlying causes of persistent problems, reducing their negative effects on services, and averting recurrences of them.

## 4.2 The Three Phases of Problem Management

1. Problem Identification

This stage denotes the early identification of potential problems that could have an impact on IT services. Utilizing sources like incident reports, trend analysis, and user feedback is one way to proactively identify potential issues and stop them from escalating.

* **Incident Reports:** Monitoring and analyzing incident reports helps in identifying recurring patterns or issues that may indicate an underlying problem.
* **Trend Analysis:** Examining historical data and trends can reveal anomalies or deviations that can lead to potential problems.
* **User Feedback:** Gathering feedback from users can provide insight into their experience, uncovering issues before they become a major concern.

1. Problem Control

Immediate steps are taken at this stage to temporarily reduce the impact of the known problem or prevent its damage. Quick solutions are implemented, such as isolating the problematic component or implementing temporary fixes, until a permanent solution is established.

* **Component Isolation:** If a particular component is causing problems, isolating it can prevent its negative impact from spreading to other parts of the system.
* **Temporary Fix:** Implementing a workaround or workaround can quickly restore functionality while a more permanent resolution is being developed.
* **Service Continuity:** The focus is on maintaining service availability and functionality while minimizing disruption caused by issues.

1. Error Control

Effective problem management also includes managing errors in their workflows. Proper tracking, reporting, and correcting errors during the problem management process is critical to ensuring process integrity.

* **Error Tracking:** Logging errors and problems encountered during problem management allows for a systematic understanding of where problems may occur in the process.
* **Error Reporting:** Transparently communicating errors to the relevant team ensures that issues are recognized and can be dealt with efficiently.
* **Error Correction:** Correcting errors in a timely manner prevents them from spreading further and maintains the accuracy and reliability of the entire problem management process.

## 4.3 Principles of Problem Management

1. Problem Identification

• Sources of Identification: Problem identification can come from user reporting, system monitoring, and analysis of operational data.

• Initial Categorization: The identified problems are given an initial category based on their type or impact, assisting in the shifting of treatment priorities.

1. Problem Investigation

High priority or high-risk problems should be resolved first, as their impact on the service is highest. The speed at which problems are investigated and diagnosed depends on the priority assigned. Correctly categorising problems makes it easier to identify trends.

* **Investigation Priority:** Issues with high impact or high risk receive higher priority in the investigation process.
* **Data Analysis:** Incident data and other information is analyzed to identify patterns or trends that can assist in investigations.
* **Source Identification:** The main aim of an investigation is to identify the underlying source of the problem.

1. Problem Analysis

* **RCA (Root Cause Analysis):** Involves an in-depth analysis method to identify the root cause of the problem, not just the surface symptoms.
* **Recovery Measures:** More effective and permanent recovery measures are planned based on the analysis.

1. Problem Resolution

• Solution Development: The problem team designs a solution that focuses on addressing the root cause of the problem.

• Change Deployment: The change is implemented, and the service is returned to its normal state with the new solution being implemented.

1. Review

• Effectiveness Evaluation: Evaluation is conducted to ensure that the implemented solution actually addresses the problem.

• Lessons Learned: Lessons from this issue are documented to improve the problem management process in the future.

## 4.4 Problem Management Example

Problem Identification:

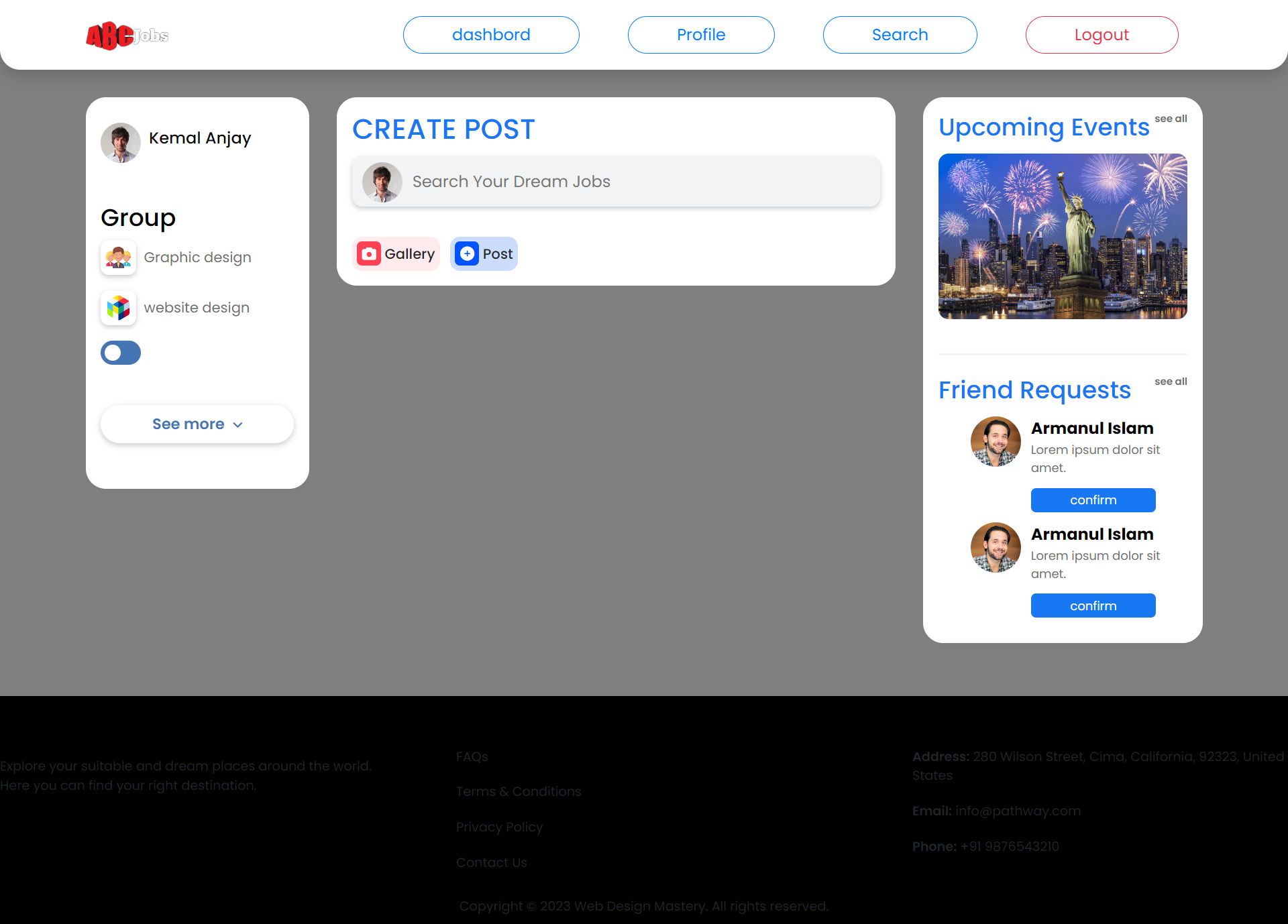
1. User Thread Not Showing:

* Description: While logged in, users were unable to browse the Thread Section.
* Impact: User not satisfied, potentially affecting service satisfaction.

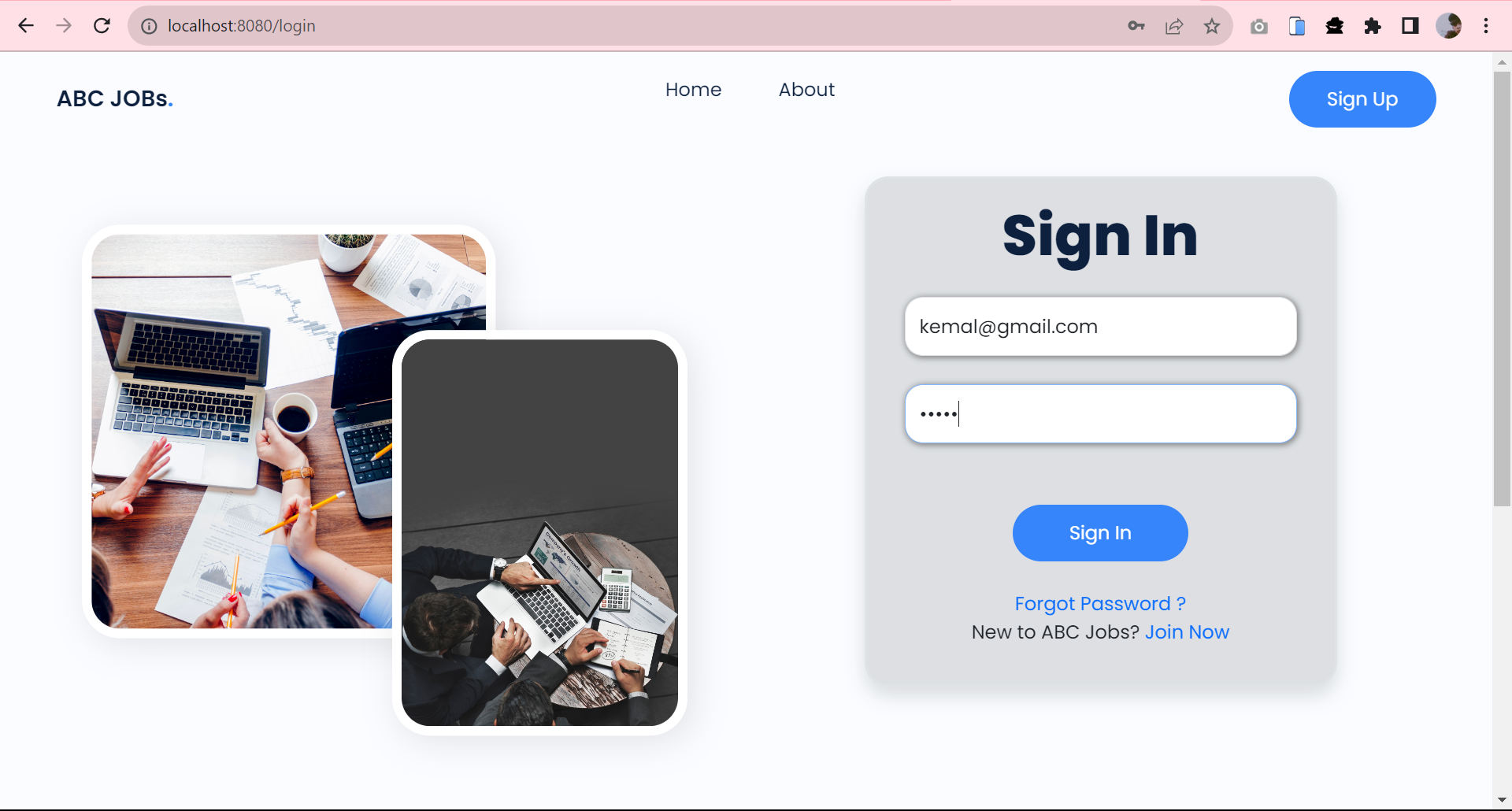
1. Error Viewing Invalid Credentials:

* Description: An error was not displayed when the user entered incorrect login credentials.
* Impact: The user may not understand why the login attempt failed.

Evidence No. 1



Evidence No. 2



# Task 2: Tools, Process & Technologies

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Issue Identification and Tracking Document | | | | | | | | | | |
| Created By: | | | Muhammad kemal | | Last Update By: | | |  | | |
| Date Created: | | |  | | Last Revision Date: | | |  | | |
| Issue No. | Issue Description | Issue Type | Identified By | Date Identified | Issue Assigned To | Target Resolution Date | Priority | Status | Date Resolved | Resolution Description |
| 1 | User Thread Does Not Display on Dashboard | Technical Issues | Customer | 14-8-2023 | Software Development  Team | 15-8-2023 | High | Resolved | 15-8-2023 | Items and Var in JSP Page Correction |
| 2 | Not displaying the invalid login details error | Technical Issue | Software Tester Team | 07-8-2023 | Software Development  Team | 08-8-2023 | High | Resolved | 08-8-2023 | Adding the code of message |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

## Problem Logging

Techniques = Issue Tracking System

Issue Tracking Techniques in Word and Figma

An essential tool for identifying issues is creating a tracking document template in an Excel sheet.

Process:

1. Utilize Document Templates: Start by leveraging the capabilities of both Microsoft Word and Figma to establish a comprehensive tracking document template. This template enables the efficient recording and vigilant monitoring of identified issues.
2. Thorough Documentation: Document each issue with meticulous attention to detail. Include essential particulars such as a comprehensive description of the issue, its type, the individual who identified it, the date of identification, the assigned team responsible for its resolution, the targeted date of resolution, its priority level, the current status, the date it was resolved, and a comprehensive description of its resolution.
3. Continuous Updates: Ensure consistent and timely updates to the tracking document as issues traverse the intricate path toward resolution. Regularly monitoring and logging the progress of each issue within the document is essential.
4. Use the tracking document to communicate issues’ status and resolutions to relevant stakeholders.
5. Leverage the Excel sheet's sorting, filtering, and data visualization features to enhance issue management.
6. **Problem Investigation**

Techniques = Root Cause Analysis

Tool = Brainstorming

Process:

1. Utilize a Fishbone Diagram to systematically analyze the root causes of identified issues.
2. Identify major categories (bones) that could contribute to the issue, such as people, process, equipment, environment, and materials.
3. Drill down into each category to identify potential root causes that might lead to the observed problem.
4. Encourage cross-functional discussions to gather insights from various team members.
5. Prioritize root causes based on their significance and relevance to the issue, enabling targeted corrective actions. Problem Resolution.

# Task 3: Investigate & Diagnose

1. **Investigate the problem**

Issue 1: User Thread Not Showing in the Dashboard

* Issue Description: User thread is not displaying on the dashboard.
* Issue Type: Technical Issues
* Identified By: Customer
* Date Identified: 14-8-2023
* Issue Assigned To: Software Development Team
* Target Resolution Date: 15-8-2023

**Investigation:** The software development team must evaluate the dashboard's user thread display capabilities in order to look into this problem. They should examine the pertinent code, database queries, and any most recent system modifications. Knowing if this is a data-related issue, a programming error, or a configuration fault that stops user threads from appearing on the dashboard is crucial.Issue 2: Invalid Login Details Error Not Displayed

* Issue Description: The invalid login details error message is not displayed.
* Issue Type: Technical Issue
* Identified By: Software Tester Team
* Date Identified: 07-8-2023
* Issue Assigned To: Software Development Team
* Target Resolution Date: 08-8-2023

Investigation: To explore this problem, the software development team needs look at the login feature and error handling procedure. They ought to check the code in charge of checking login information and showing error notifications. The goal of the inquiry is to ascertain why the system does not display the error message when incorrect login information is given.

**Diagnose the problem**

Issue 1: User Thread Not Showing in the Dashboard

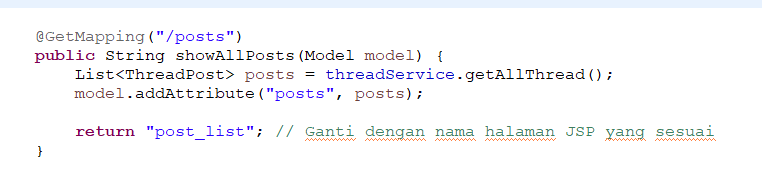
Diagnosis: After further investigation, it was discovered that the code in charge of retrieving and showing user threads on the dashboard contained a flaw that had been caused by a recent code change. User threads were missing from the dashboard as a result of the bug that prevented the system from retrieving them from the database.

Diagnosis Steps: The development team identified the specific code segment responsible for querying and displaying user threads. Through code review and debugging, they traced the issue to an incorrect SQL query that failed to retrieve the necessary data. This led to an empty result set, resulting in the absence of user threads on the dashboard.

Diagnosing the Root Cause of the Problem through Debugging:

Debugging is a crucial step in diagnosing software-related issues. In this case, the development team utilized debugging techniques to uncover the root cause of the problem:

* Code Inspection: Developers reviewed the code responsible for fetching user threads and analyzed the logic behind the data retrieval process.
* Print Statements: The team strategically inserted print statements within the code to track the data flow and identify where the process was breaking.
* Variable Inspection: Debugging tools were used to inspect the values of variables and database queries during runtime to identify discrepancies or errors.
* Step-by-Step Execution: Developers executed the code step by step, monitoring variable values and code behavior at each stage to pinpoint the moment when the data retrieval failed.



Issue 2: Invalid Login Details Error Not Displayed

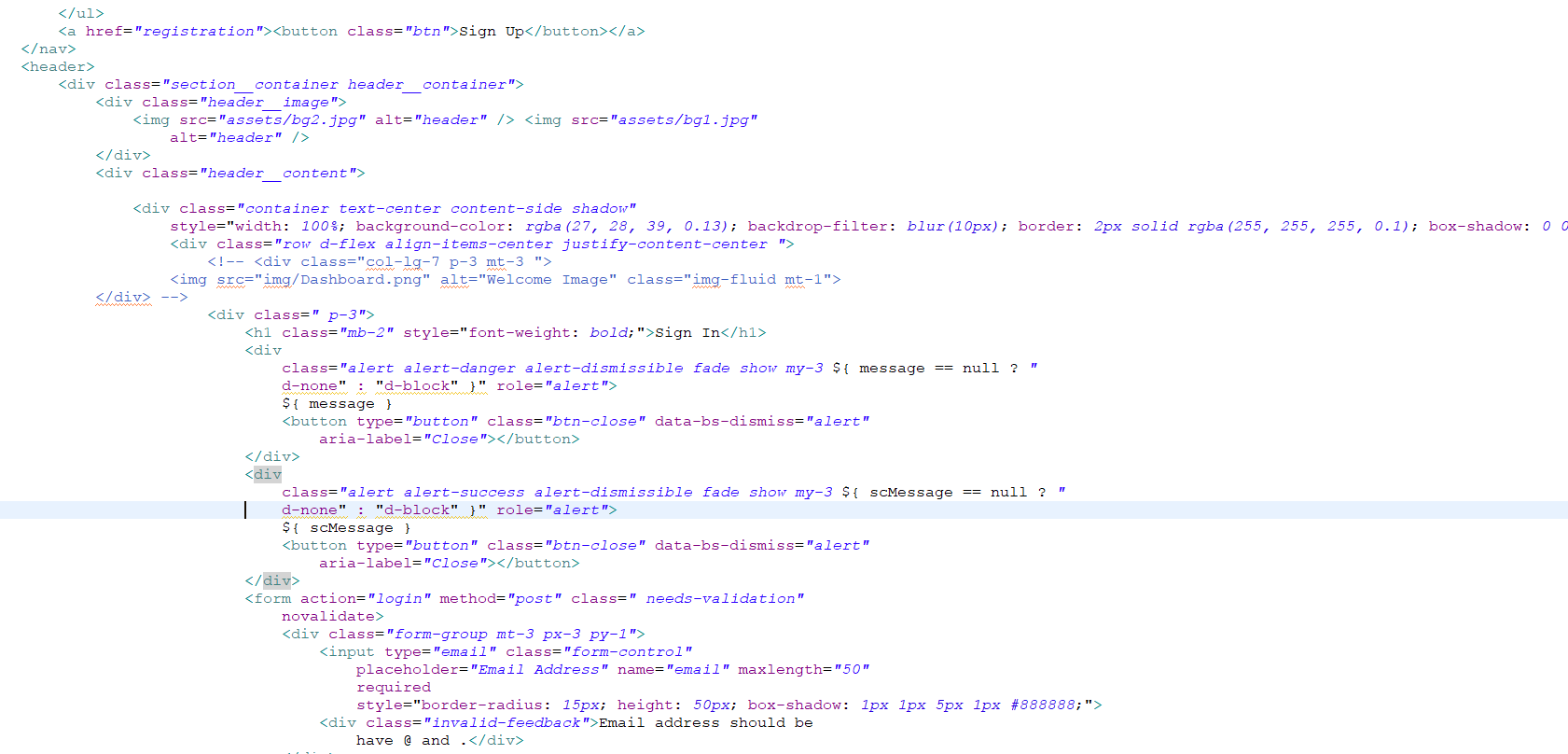
Diagnosis: After investigating the issue, it was identified that the error message display code is not being executed when invalid login details are entered. As a result, users need to receive feedback when their login attempts fail.

Diagnosis Steps: The development team traced the issue to an oversight in the login validation process:

* Code Review: The developers reviewed the code responsible for validating login details and identified that the error message display function was not being called within the code block handling invalid login attempts.
* Conditional Logic Issue: Upon further examination, it was discovered that a conditional statement intended to trigger the error message display was not evaluated correctly due to a typo in the code.
* Test Scenarios: The development team executed test scenarios with deliberately incorrect login details to replicate the problem and observe the system's behavior. They confirmed that the error message was not appearing as expected.

Debugging was essential in uncovering the root cause of this issue:

* Code Inspection: Developers reviewed the login validation code and identified the conditional logic governing the error message display.
* Debugging Statements: Print statements were inserted to confirm the flow of execution and identify whether the code block for the error message display was being entered.
* Condition Evaluation: Debugging tools were used to inspect the values of variables and expressions during runtime to pinpoint the moment when the conditional statement failed to evaluate correctly.
* By leveraging debugging techniques, the development team identified the root cause—conditional statement error—and rectified the issue. The error message for invalid login details was reinstated, providing users with the necessary feedback.

****

# Task 4: Explain Prioritize

## Priorities and categories problems:

**Tier 1: Low priority issues**

* The first tier is for basic or minor problems.
* Handle low-priority incidents that do not impact the business. It is easy to solve and repeats a lot. Usually, incidents here are converted into incident models.

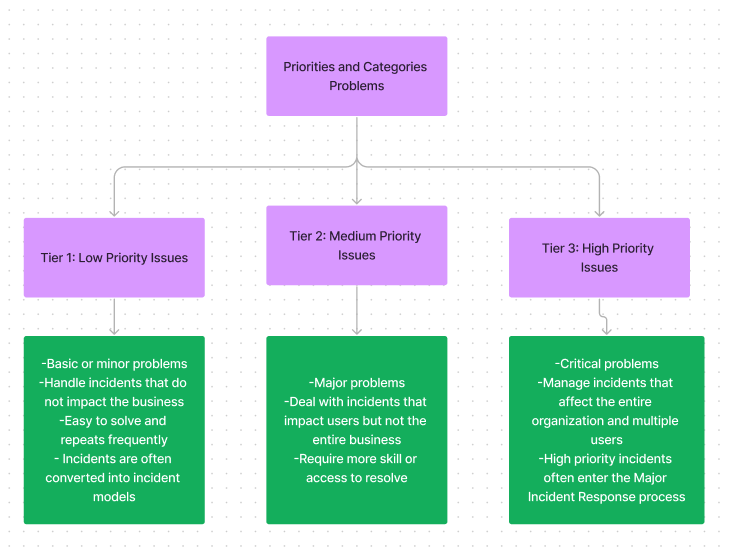
**Tier 2:** **medium priority issues**

* The second tier is the major problems.
* Dealing with incidents that impact users but not the business as a whole. These incidents require more skill or access to resolve.

**Tier 3:** **High Priority Issues**

* The third tier is the critical problem
* Manage incidents that affect the entire organization and multiple users. These incidents are high priority and often enter the Major Incident Response process.

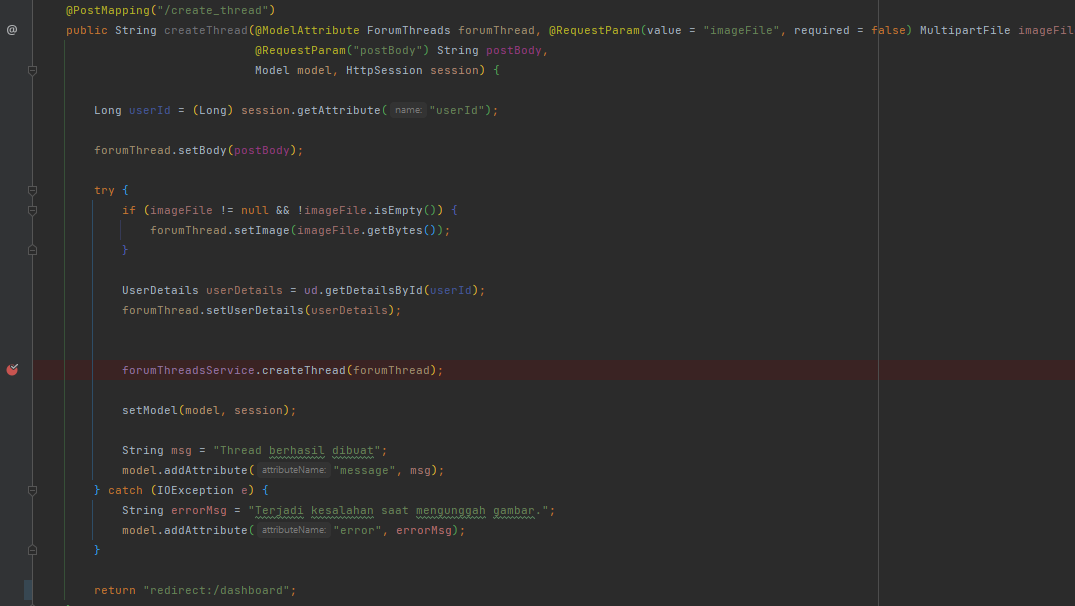
**Problems Categories Diagram:**

****

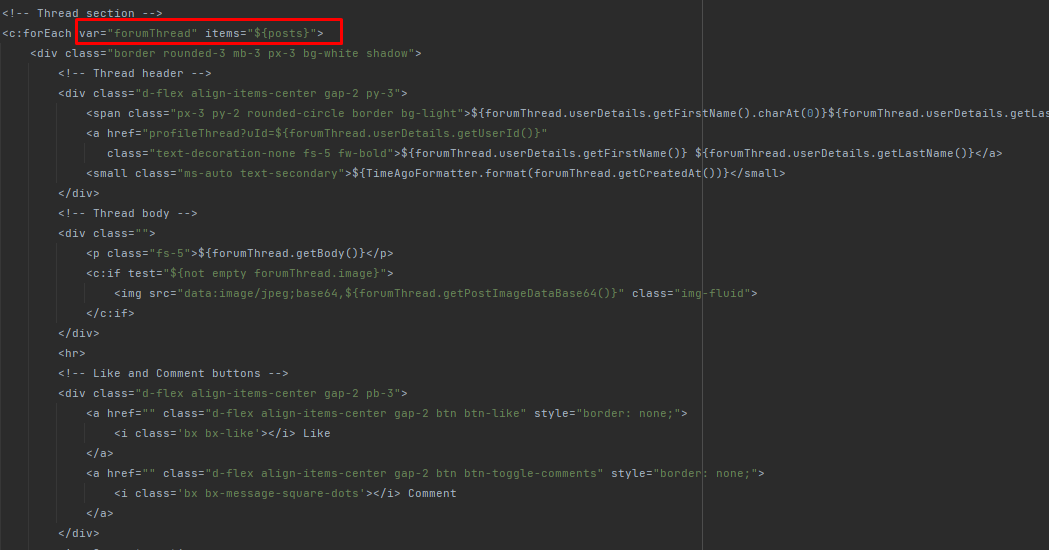
# Task 5: Problem Management Solution

**Solution for Thread not showing in The Dashboard:**

1. Debug create\_thread endpoint
2. Check the logic in create\_thread endpoint
3. Ensure in the JSP have a correct var and items it will sync with backend and database.
4. Ensure have implement endpoint to call all thread



Debug is success and not have a problem with endpoint



Identify the var and items. And check the attribute in the thread

And this is the endpoint for call the thread

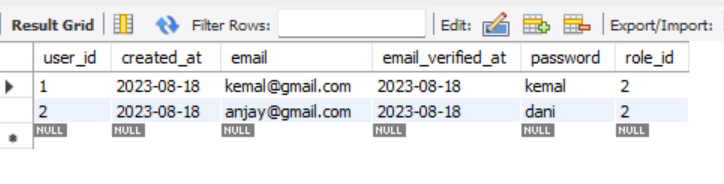


**Solution for Invalid login details error not displayed:**

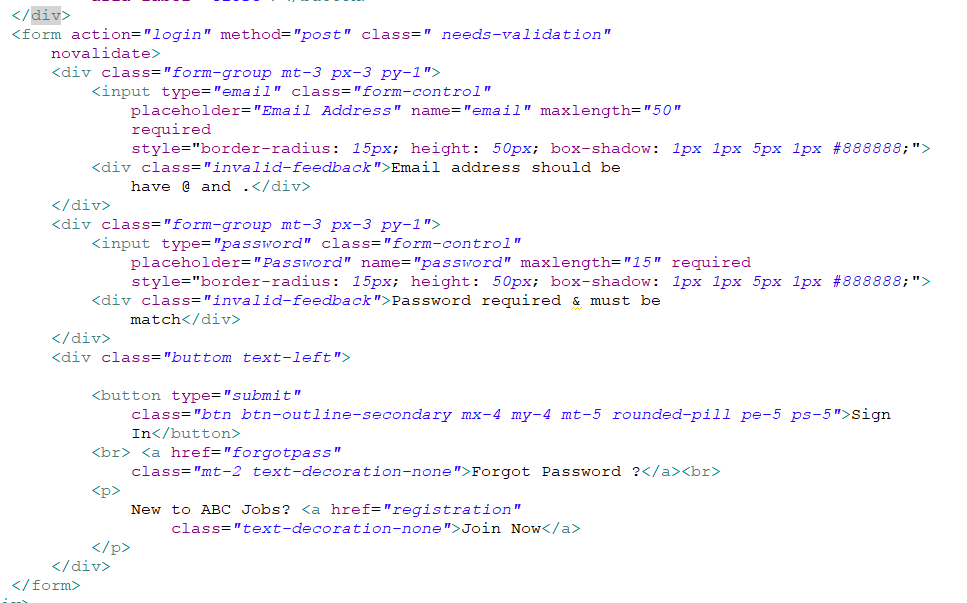
1. Debug Login Endpoint
2. Ensure the data user has saved in database
3. Identify the login.jsp have a correct variable to synchronize with backend
4. Ensure the login.jsp have a code for show the message



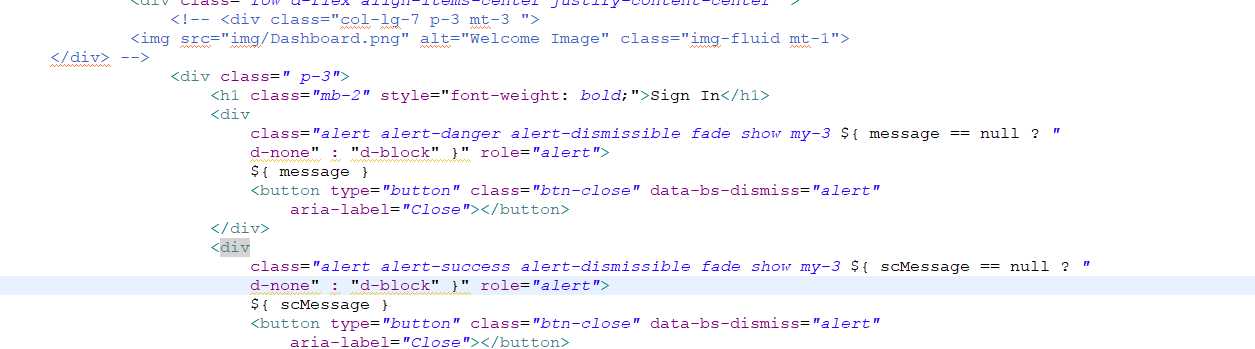
Debug Login Endpoint. Ensure we have a correct logic



The Data of User already registered has saved to database



Ensure All the variables is correct



Add the message code if error login

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Known Error Database | | | | | | |
| **Ticket no.** | Issue Description | Issue Type | Root Cause | Workaround | Status | Date Resolve |
| 1. | User Thread Not Showing in the Dashboard | Technical Issues | There is mistake items and var in JSP Page | Correcting Items and Var in JSP Page | Resolved | 15-8-2023 |
| 2 | Invalid login details error not displayed | Technical Issue | Code for Error message is empty | Adding the code of message | Resolved | 08-8-2023 |

# Task 6: Systems will be implemented

**Brainstorming Method:**

The brainstorming technique will be used to speed up efficient idea development and problem solutions. The collaborative and innovative process of brainstorming invites team members to offer ideas, recommendations, and potential fixes for problems that have been recognized. The brainstorming process can provide original and workable solutions by utilizing the group's aggregate knowledge and varied viewpoints.

**Steps of Brainstorming:**

1. **Problem Statement:** Identify the problem or challenge you want to address through the brainstorming session. Ensure that the problem is clearly defined.
2. **Team Collaboration:** Invite team members involved in solving the problem to participate in the brainstorming session. Ensure that each member has a shared understanding of the problem.
3. **Idea Generation:** Ask each team member to generate ideas related to the identified problem. These ideas could be solutions, concepts, or other creative thoughts.
4. **Quantity Over Quality:** Encourage team members to produce as many ideas as possible without considering the quality or feasibility at this stage.
5. **No Criticism:** Avoid criticizing or judging ideas during the initial phase. Focus on freely gathering ideas without the fear of rejection.
6. **Build on Ideas:** Encourage participants to build on the ideas presented by others. This can lead to the developing of more refined and innovative solutions through collaboration.
7. **Divergence and Convergence:** Begin with a divergent phase where various ideas are generated. Then, move into a convergent phase where the team narrows the list and identifies the most promising ideas.
8. **Capture Ideas:** Record all the ideas generated during the brainstorming session. This could be in the form of notes, sketches, or other media.

**Benefits of Brainstorming:**

* **Diverse Solutions:** Brainstorming encourages the generation of various solutions by involving different perspectives and backgrounds.
* **Innovation:** Creative thinking during brainstorming can lead to innovative approaches that might have yet to be considered initially.
* **Team Collaboration:** Brainstorming promotes teamwork and collaboration, fostering a sense of shared responsibility for problem-solving.
* **Engagement:** Team members become engaged and invested in the solution-finding process, enhancing their commitment to implementing the chosen solutions.

**Implementation Plan:**

* Identify the problem to be solved, determine the team involved, and schedule the brainstorming session.
* Conduct the brainstorming session following the steps described above. Ensure each team member has an opportunity to speak and contribute.
* Assign a facilitator to guide the brainstorming process, ensuring everyone has a chance to contribute.
* Document all ideas for later evaluation and selection of viable solutions.

# Task 7: Best Practices in Problem Management

Maintaining operational efficiency and providing outstanding service require managing problems while adhering to best practices and industry standards. Five key best practices for efficient issue solving are listed below:

1. **Clear Separation of Clients and Incident Handlers:** Establishing a distinct barrier between the client and the person in charge of addressing incidents is crucial for enforcing an open and unbiased problem-solving procedure. Potential conflicts of interest are avoided by making sure the incident controllers are separate from the affected clients. This division makes it easier to do unbiased research, communicate clearly, and look for solutions. It is a fundamental procedure that promotes trust and gives clients the assurance that their issues are dealt with properly and impartially.
2. **Establishment of a Known Errors Database:** Continuous progress and well-informed decision-making depend on a robust Known Errors database. Every incident that is resolved should be thoroughly recorded in this database. By doing this, a database of past incidents and their fixes is produced, serving as a useful tool for troubleshooting in the future. The Known Errors database facilitates proactive problem management and speeds up the resolution of reoccurring problems. The effect of possible issues can be reduced by preventive steps and systemic improvements based on patterns and trends found in this database.
3. **Root Cause Analysis and Resolution:** It's important to deal with a problem's symptoms, but it's just as important to find and fix the problem's core cause. To understand the underlying causes of occurrences, perform a thorough root cause analysis. Finding the underlying reason paves the way for long-lasting solutions and stops similar problems from recurring. This procedure raises the standard of service while demonstrating a dedication to producing long-lasting effects.
4. **Continuous Communication and Feedback Loop:** Consistent client and stakeholder communication is necessary for effective problem management. To keep clients updated on the progress of incident response and any preventive steps taken, keep the feedback loop open. This open line of contact gives your problem-solving procedure credibility and demonstrates to clients that their problems are being treated seriously. Additionally, client feedback offers priceless insights for optimizing service delivery and problem-management tactics..
5. **Regular Review and Improvement:** Problem solving is a dynamic process that needs constant review and development. Review the effectiveness of your problem-solving procedures on a regular basis, looking at statistics like client satisfaction and incident resolution time. Utilize these insights to pinpoint areas that could be improved, to streamline procedures, and to put appropriate remedial measures in place. A continuous improvement culture makes sure that problem management is flexible and sensitive to new problems.

Organizations may improve their problem-management skills, raise customer satisfaction, and foster a resilient and proactive problem-solving culture by adopting these best practices.