International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Page Mapping and Page Rights Implementation

Naveen. G

Sysarc Infomatix Private Limited, Chennai, India

Abstract

I have worked for page rights and page mapping implementation. We have logical unit rights, core logical unit mapping, core page category and core logical unit table. We have read, write, none access for each page. When we have 'read' access page will be displayed but all the fields will be disabled and button will be hidden. When we have 'write' access page will be displayed, fields will be editable and button will be displayed. We can perform save/edit and delete operation.

Keywords: Page rights, Page mapping, Read/write access.

INTRODUCTION

I have implemented page mapping and page rights implementation. For each page implementation, we have core page category table. We have sponsor, member bank and banker user. In sponsor login, In process application will be editable. In member bank login, pages will be disabled and member documents will be enabled. So, member can attach documents. Member will submit the request to banker. Banker can approve/reject the request.

LITERATURE SURVEY

I have implemented page mapping and page rights for Progress report page. When we retrieve the data from database it will be 'read' operation. When we implement save/update data from database it will be 'write' operation. It is implemented so that unauthorised users cannot access the page. We will display invalid access alert. When a user has read only access, page will be displayed and fields will be disabled. We had existing JSP and EJB source. We implemented in Angular and Restful webservices.

PAGE MAPPING AND PAGE RIGHTS IMPLEMENTATION

I have added entry in core logical units, core logical unit mapping, logical unit rights, core page category table. We have setup, application, administrative options menu. For setup level pages, we have parent id 'set up'. For application level page, we have parent id 'application'. For administrative level page, we have parent id 'administrative options'. Each url entry will be added in table. In backend, we have annotations get mapping, post mapping, put mapping.

During server startup, we will hit the database and retrieve the data. Angular is a single page Application. So, we set the data in service by using map. And when we navigate to each page, data will be retrieved from map. We implemented page mapping and page rights in UAT and deployed patch in production.

RESULTS AND DISCUSSION

For each round, we will implement scoring to calculate rank to eligible applications. After scoring, each application will be disabled in sponsor login. We implemented session storage and added disabled

properties to each key value. The output was successful and when the user tried to access a page in which he/she does not have access 'invalid access' alert will be displayed. url tampering and unauthorized access is restricted.

CONCLUSION

I have implemented page rights for progress report page. I tested in local, testing server and it worked. I faced issues while development and fixed the issues by debugging the code. It passed User Acceptance Testing(UAT) and moved to production. I want to thank my team leader and collegues for the support.

FUTURE SCOPE

I have implemented the logic in Angular for front end and Restful websevices for backend. We used Node js framework for front end. We used Hibernate, Spring boot framework for backend. We used MS SQL for database. We can try implementation using React js for front end and python for backend.