

Cognizant Academy

truYum

FSE Spring Data JPA & Hibernate Specification Document

Version 1.0

	Prepared By / Last Updated By	Reviewed By	Approved By
Name	Chandrasekaran Janardhanan	Vimalathithan Krishnan	Ramadevanahalli Lingachar, Shashidhara Murthy
Role	Learning Solution Designer		
Signature			
Date			

Table of Contents

1.1	1 Purpose of this document			
1.2	1.2 Definitions & Acronyms			
1.3	.3 Project Overview			
1.4	Scope	2		
1.5	Intended Audience	3		
1.6	Hardware and Software Requirement	3		
2.0	Project Setup in Eclipse	3		
3.0	Database and Bean Mapping configuration			
4.0	TYUC001 & TYUC002 - View Menu Item List (Admin & Customer)	4		
4.1	Populate form fields in Edit Menu Item component	4		
4.2	Save Menu Item	4		
5.0	Coding Standards and Guidelines	4		
6.0	Submission	4		
6.1	Code submission instructions	4		
7.0	Change Log	5		

Introduction

1.1 Purpose of this document

The purpose of this document is to define the user interface specification for truYum project.

1.2 Definitions & Acronyms

Definition / Acronym	Description
JPA	Java Persistence API

1.3 Project Overview

Refer truyum-user-stores.xlsx for understanding the functionality and features.

1.4 Scope

Develop Spring Data JPA implementation for truYum application.

1.5 Intended Audience

- Product Owner
- Scrum Master
- Application Architect
- Project Manager
- Test Manager
- Development Team
- Testing Team

1.6 Hardware and Software Requirement

- 1. Hardware Requirement:
 - a. Developer Desktop PC with 8GB RAM
- 2. Software Requirement
 - a. Eclipse

2.0 Project Setup in Eclipse

Use the same Eclipse project created for spring core specification and implement spring data jpa specification on top of it.

3.0 Database and Bean Mapping configuration

- Modify pom.xml to support Spring Data JPA and Hibernate dependencies.
- Modify all classes in model package to refer to the respective tables and columns.
 - IMPORTANT NOTE
 - Change all List to Set
- Based on the relationship type define necessary many-to-one, one-to-many and many-to-many relationships.
- Create necessary Repository classes for each model with empty implementation.

4.0 TYUC001 & TYUC002 - View Menu Item List (Admin & Customer)

- Create a method in MenultemRepository to filter out the active items and launch date is not in future
- Modify MenultemService method to use the repository method.
- Modify getMenuItemListAdmin() to invoke repository's findAll() method.
- Modify getMenultemListCustomer() to invoke the method defined in MenultemRepository.

5.0 TYUC003 - Edit Menu Item

5.1 Populate form fields in Edit Menu Item component

Modify MenultemService.getMenultem() to use getOne() method of MenultemRepository.

5.2 Save Menu Item

Modify MenultemService.modifyMenultem() to use save() method of MenultemRepository.

6.0 Coding Standards and Guidelines

- Database script files should be placed in the relevant folders
- Define @Transactional annotation in all service methods
- Never use System.out.println(), use logger.debug() instead
- Include start and end logs in each method
- Include debug logs for data retrieval and flow
- All standards defined in Java module needs to be followed.

7.0 Submission

7.1 Code submission instructions

Use git add, commit and push commands to upload your code into remote GltLab repository.

During commit, give the commit message as "spring-data-jpa".

8.0 Change Log

	Changes Made				
V1.0.0	Initial baseline created on <dd-mon-yy> by <name author="" of=""></name></dd-mon-yy>				
Vx.y.z	<please are="" be="" below="" change="" changes="" configuration="" control="" details="" followed="" form="" given="" if="" item="" maintained="" needs="" not,="" of="" refer="" separately.="" status="" template="" the="" to="" tool=""></please>				
	Section No.	Changed By	Effective Date	Changes Effected	