



Nagarro Software Pvt.
Ltd.

Case Study

Retail Inventory Management Software System (RIMSS)

Version: 1.3



Table of Contents

1	INTRODUCTION	3
2	PROBLEM STATEMENT	3
3	KEY REQUIREMENT CONSIDERATIONS OF EXPECTED SOLUTION	4
3.1	Vision.....	4
3.2	Key Features	4
3.2.1	Functional Requirements	4
3.2.2	Non-Functional Requirements.....	6
4	DESIGN REQUIREMENTS	7
5	QUESTIONS	7

1 Introduction

This document describes a problem that “YCompany” is currently facing. The company is planning to build an efficient “Retail Inventory Management Software System - (RIMSS)” application for this problem.

This case study expects you to go through the problem and based on your understanding propose solutions to various questions at the end of this document.

2 Problem Statement

“YCompany” is a leading luxury fashion brand that design, manufactures and markets variety of clothes, shoes and accessories for men, women and children. They are famous for sweaters, moleskin clothing, corduroy clothing, and tattersall shirts. In recent years, the company has successfully modernized its collections to appeal to a younger clientele while maintaining credibility with the traditional countryside customer. One of the challenges that they are facing is an unavailability of an efficient online shopping medium, to cater online customers. This is impacting their overall revenue and sales growth.

As part of this initiative, they would like to have a software/application, which provides an efficient and effective online experience to customers. They currently have a website which simply showcase all the products and provide purchasing services. But the website has following key issues:

- It does not provide cross-platform support
- Initial load time of website is more than a minute
- UI is not smooth and even freezes sometimes
- Product search takes unreasonable time to display results

They would like to have a new retail management inventory software system, which provide a smooth user experience and address all the above issues.

3 Key Requirement Considerations of expected solution

3.1 Vision

Vision of Retail Inventory Management Software System (RIMSS) is to provide:

- Best online shopping experience
- Seamless user interface
- Highly configuration and scalable system
- Cross platform support

3.2 Key Features

RIMSS will be an environment, which consist of a web application and a backend system. The feature set can be categorized into following two sections:

3.2.1 Functional Requirements

This section caters the following key functional requirements.

- **Home page** – Home page is the very first screen of the application. The homepage will contain the featured products, latest offers, navigation, menus, and some basic information.

- **Product search** – Product search is the key component of the application. It's must be fast, convenient and efficient. Product can be searched and filtered by following
 - Product category
 - Discounted products
 - Product cost
 - Product name
 - Product can also be filtered by color
- **Product showcase screen** – Clicking on a product (thumbnail) will open a product showcase screen. Product showcase screen will represent information about the product, product images, product cost, any discount available for the product, add to cart button and other relevant information.
- **Shopping cart** – Application must be having a shopping cart functionality, which allow users to manage the products they are interested in. Product items can be added or removed by a user in the shopping cart. During any visit to the website, shopping cart must show all the visitor's products (with total cost) available in the cart/basket. Later on, items in the "Shopping cart" can be purchased via a payment gateway.
- **Payment gateway** – Application must offer a payment gateway to provide smooth online shopping experience. Payment gateway is an online analogue of a physical credit card processing terminal that we can locate in retail shops. Its function is to process credit card information and return the results back to the online store system.
- **Latest offers (pluggable) on products** – Latest offers on products can be shown in the application home page via banners, carousel, etc. They can be easily pluggable into the system. They must be customizable in nature such as show/hide feature, configurable theme, etc.

3.2.2 Non-Functional Requirements

This section caters the following key NFRs.

- **Cross platform support** – Application must work on multiple platforms such as Desktops, Tablets and Mobiles. YCompany has conducted a market research which shows that there is almost 300-600% increase in revenue by targeting multiple devices. So, the idea is to run newly envisioned application on desktops, smartphones and tablets.
- **Cross browser compatibility** – Application must support following browsers
 - Chrome (Desktop and Android)
 - Edge
 - Firefox
 - Safari (macOS and iOS)
- **Smooth UI and rich user experience** – A response to a user interaction must be generated within 100 milliseconds after an interaction. Studies by various researchers found, that the maximum time tolerated for smooth interaction is 100ms. After that, an interface experienced as irresponsive and slow. This means that either the desired action has to be processed within the timeframe or that an adequate reaction shall be shown to the user. This can for example be a spinning circle symbol, showing that input is being processed.
- **In-built improved SEO** – Application must adhere to the SEO best practices. The application must be built with SEO in mind to help make online shop on products more visible and findable through search engines.
- **Timely updates to keep the website in trend** – Application must be designed in such a way that it should be scalable enough to adapt the latest website trends.

- **Use of latest technology for best results** – Latest and standard frameworks and libraries should be used to architect and design the application. Application overall purposed solution and codebase must adhere to the industry best practices.

4 Design Requirements


Below are some design requirements for RIMSS:

- System should have a scalable and flexible design
- System should be modular in nature and supports pluggable architecture
- System should have enough logging to help in debug any error condition
- Continuous integration support
- Unit testability

5 Questions

Below exercises are available for TL's to complete, as per the topic covered in TL Training. In order to complete this exercise, following templates & samples should be used.

Estimation (Sample)	
Solution Approach (Sample)	 XYZ Corp - Solution Approach - 1.0.docx

General Template	 General Template v1.0.docx
------------------	--

S.No	Question	Technology	Details
1.	<p>Create a Solution Approach Document containing high-level solution design.</p> <p>Note: Use Solution Approach template for this.</p> <p>Pull out NFR in a single page as you think appropriate.</p>	a) Any framework and library of your choice	<p>Your solution approach must have following concepts:</p> <ul style="list-style-type: none"> a) Technical Diagram (1 Page) b) Solution Architecture (1 Page) c) Non-Functional Requirements Coverage (1 Page) d) Performance (1 Page) e) Assumptions & Scope (In Scope/Out Scope) (1/2 Page)
2.	<p>Create an estimation sheet for the complete solution based on the selected technology stack.</p> <p>Note: Use Estimation Sheet Template. While doing estimations, please take care of NFR, In Scope, Out Scope and Assumptions.</p>		<p>Cover all stages of below phases in excel and give final numbers</p> <ul style="list-style-type: none"> a) Requirement Phase b) Design Phase c) Development Phase d) Testing & Bug Fixing Phase e) Project Management
3.	Create a working sample (for detail check <i>Technology</i> and	Max of 1 product showcase screen, 1 product search screen	Your working sample solution must have following concepts properly developed.

	Details sections) for the given problem depicting n-tier architecture. You can use any Framework of Choice.	and 1 sample task running should be coded. You can create mock data services using NodeJS or static JSON files.	<ul style="list-style-type: none"> a) Plugin based architecture. The solution should be generic enough so that any functional module could be plugged into the application. b) Create and implement at least one functional module. c) Application should be responsive in nature. d) Unit testing for business layer covering at least 2 unit tests
4.	Create a single page strategy document covering the build process for your solution. Note: You can use the General Template attached with the document.		We are expecting one pager only. This can be a single page presentation also.

Total Number of Deliverables : 3 Doc/PPT, 1 estimation sheet, 1 working sample

Total weeks available to execute : 4 Weeks (Calendar Days)

Note: Weightage will be provided to those covering all necessary design principles in solution approach and diagrams.

You are not required to provide backend solution for the given problem statement. You will be evaluated for Frontend solutioning only. You can assume that client will provide the APIs.

But, if you have expertise in backend, then you can compile the "Full Stack" solution, and you will be evaluated for the complete solution accordingly.