



Test Assignment

Prepared for: Fullstack Web Developer (Front-end and Back-end .NET)

Prepared by: Retail Exchange Network, Inc.

USA, 2017.

RETAIL EXCHANGE NETWORK, INC.

This assignment is designed to get a better overview of your skills, experience and creativity in programming as a full-stack web developer.

The assignment is a basic 1-page-website which allow user to create new or load existing product information.

This assignment contains 3 parts, communicate with each others as following:

Font-end Web Page ↔ Back-end .NET Web ↔ Service SQL Server Database

PART 1: FONT-END WEB PAGE

DESCRIPTION

Implement this 1-page-website with following user case:

1. Page header and basic information tab

- When user enter existing product number into product number field, load product information and allow user to edit and save basic product information.
- When user enter none-exist product number into product number field, prepare web page for entering and saving new product basic information.

2. Inventory tab

- When user switch to inventory tab on existing product, load product inventory quantity information on multiple inventory site.

REQUIREMENTS

1. Generic requirements

- Implement this page using basic HTML, CSS/BOOSTRAP, JAVASCRIPT/JQUERY for render, processing, transfer data from/to server.
 - Layout using HTML, CSS/BOOTSTRAP
 - JAVASCRIPT/JQUERY/JQUERY UI for UI Controls, making API requests, processing responses etc with back-end API services...
- Do not use ASP.NET or fully server-side processing framework to implement this web page.

2. Product Master header & Tab basic information requirements

- Product Number: Auto-complete suggestion based on what user type in.
- Product Image: Allow user to update/save/delete product image

RETAIL EXCHANGE NETWORK, INC.

- Cost Center Code: format ####-##-## (# is 0-9)
- Unit Price, Width, Length, Height: Format 2 decimal places
- Cube: Format 3 decimal places
- Cube value = Automatically calculated base on Width x Length x Height
- Prepack: Dynamic disable select pre-pack type on pre-pack checking condition Page header and basic information tab requirements
- Cost Center Code: format ####-##-## (# is 0-9)
- All UI Controls (button, date, autocomplete) should be implemented by using UI Front-end library/framework (for example: JQuery UI, Bootstrap Components ...)

PRODUCT MASTER SHEET

Product Master

Product Number

91250


Pack Size

12

Description

Handi-Foam 75 Spray Polyurethane ...

Product Image



Browse Logo

Discard Logo

Basic Information

☒ Hazardous Material?

Cost Center Code

9400-14-16

Expiration Date

05/20/2019

Unit Price

220.95

Inventory

Width

3.00

Length

2.00

Height


2.00

Cube

12.000

☒ Pre-Pack

Solid Styles

 Save

3. Tab inventory requirements

- When switch from Basic Information tab to Inventory tab, show inventory information of current existing item on this table
- One item can be stored in multiple Inventory Site (Example: CET, SEA) with Quantity On-Hand, Allocation information
- Other quantities are calculated on On-Hand Qty and Allocation Qty as follow:
 - Pieces (Pcs) column = Quantity x Item Pack Size
 - Exampe: On-Hand Pcs = On-Hand x Pack = 10 x 12 = 120
 - Available = On-Hand - Allocated
- Footer row to summary quantity on ALL SITE
- Data Table should be implemented by using jqGrid: http://www.trirand.com/blog/?page_id=5

PRODUCT MASTER SHEET


Product Master

Product Number Pack Size

Description

Handi-Foam 75 Spray Polyurethane ...

Product Image




Browse Logo

Discard Logo

Basic Information

Inventory

Location	On-Hand	On-Hand Pcs	Allocated	Allocation Pcs	Available	Available Pcs
CET	10	120	3	36	7	84
SEA	20	240	10	120	10	120
ALL SITE	30	360	13	156	17	224

 Save

PART 2: BACK-END WEB SERVICE

DESCRIPTION

Process API for front-end website to connect and load/create/save product data.

REQUIREMENTS

- Use .NET to implement and host web services (for example [ASP.NET](#), [ASP.NET MVC](#) web service)
- Database processing using Entity Framework and LinQ

PART 3: SQL DATABASE

DESCRIPTION

- Store demo data for all example product master and each product example inventory.

REQUIREMENTS

- Use SQL Server to create **2 tables** with following schema: 1 Item Master has multiple Inventory Sites.

1: ItemMaster Table	2: ItemMasterInventory Table
ItemMasterID_PK <i>(This primary key is also product number)</i>	ItemMasterInventoryID_PK
IMPack	IMItemMasterID_FK
IMDescription	IMSiteID
IMImageData	IMQtyOnHand
IMIsHazardousMaterial	IMQtyAllocated
IMExpirationDate	
IMUnitPrice	
IMWidth	
IMLength	
IMHeight	
IMIsPrePack	
IMPrePackStyle	
IMCostCenterCode	