WORLDSKILLS SINGAPORE QUALIFYING ROUND TEST PROJECT (TEMASEK POLYTECHNIC)

IT SOFTWARE SOLUTIONS FOR BUSINESS

SESSION 1

WSS2020_QR_TP_S1

Written by:
WorldSkills Singapore Expert
(Temasek Polytechnic)

CONTENTS

Session 1 of this Test Project consists of the following additional files:

1. WSS2020_QR_TP_S1.pdf (Session 1 instructions)
2. WSS2020_QR_TP_S1_MSSQL.sql (SQL Script to create tables with data for MSSQL)

INTRODUCTION

In this session, you will start developing the ASEAN Skills 2020 application. Due to the complexity and unique characteristics of hosting this event in Singapore, the WorldSkills Singapore Council has decided to hire you to develop an in-house customized system to support their various business processes during the preparation for the event and the actual event. The designer has provided you with some system documentation so that you can build it according to the client's needs. Take some time to carefully look through what has been provided and what is required. Prepare your own test data to help you to test the application. Document your test data, assumptions and any other key information in a Readme.txt file and save it with the other files that you are submitting.

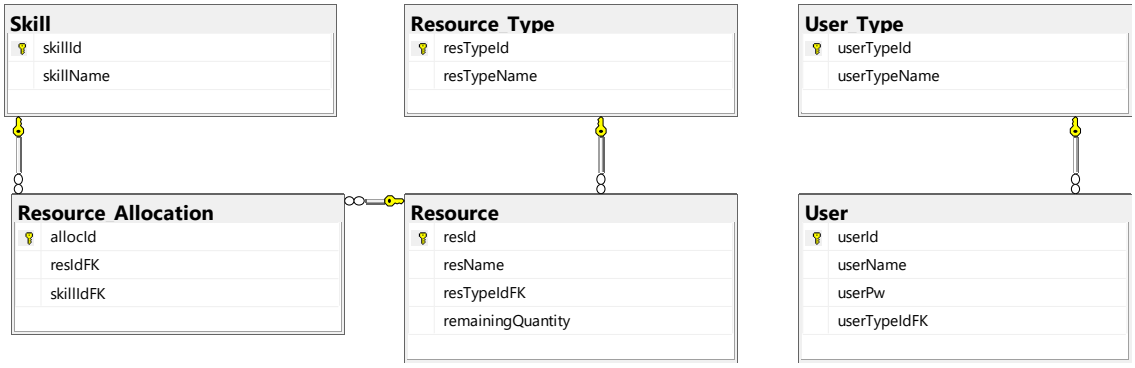
INSTRUCTIONS FOR THE COMPETITOR

In this session, you will be developing a desktop/web application (you decide which is more appropriate to meet the client's needs for this session). While developing this application, please ensure that you confirm to the following basic instructions:

- You should consistently follow the provided style guide throughout the application.
- Time management is critical to the success of any project so by the end of this session, you should submit the deliverables listed in next section so that the ASEAN Skills 2020 application will be finished on time. Any deliverables that are not submitted on time will not be evaluated.
- Make sure that you follow the provided style guide throughout all parts of the system.
- Make sure that you follow the general layout and flow of the screens as outlined in this document and the storyboard.
- Make sure that you provide appropriate validation and error messages throughout all parts of the system.
- Make sure that all relevant buttons/links are working at the end of the session.
- Make sure that you use appropriate naming conventions for all parts of the system as needed.
- Where applicable, include comments in your code so that it is easier for evaluators and the client to understand your code.
- Do note that you are building the entire system progressively so some functions may only be added in subsequent sessions.

WORKING WITH THE DATABASE

Create a database by the name of “Session1”. This will be the only database that you use in this session. Save this database in your main project folder. An SQL Script is provided for you. This script consists of the database structure and data required to complete the tasks in this session. As instructed by the designers, the database structure for this session cannot be altered (i.e. no adding or removal of tables, fields in the tables or data types). To help you understand the database structure, the database designers provided an Entity-Relationship Diagram (ERD), which explains the conceptual and representational model of the data used in this database.



DELIVERABLES

1.1 Create "1. Main menu"

Create the main menu of the application, as outlined in "1. Main menu" in the wireframe.

This screen can be accessed by all users.

1.2 Create "2. RM account creation"

Create the screen for the system administrator to create new accounts for new resource managers who need to manage the resources used in the competition, as outlined in "2. RM account creation" in the wireframe.

This screen can be accessed by all users. Include appropriate errors checks for data entry. The user ID should consist of a minimum of 8 characters. No two users can share the same user ID. The list of valid user types should be dynamically pulled from the database and users cannot select or input a type that that is not valid. After successfully registering as a resource manager, the user is automatically re-directed to the main menu but is not yet logged in to the application (i.e. the resource manager must still login to access the resource management screen).

1.3 Create "3. RM login"

Create the screen for resource managers to login to the application, as outlined in "3. RM login" in the wireframe.

This screen can be accessed by all users. Only resource managers can successfully login via this screen. Include appropriate error checks and messages for data entry and unsuccessful login attempts. When a resource manager successfully logs in, they will be automatically re-directed to the resource management screen.

1.4 Create "4. Resource management"

Create the screen for resource managers to view all their resources and delete unwanted resources, as outlined in "4. Resource management" in the wireframe.

This screen can only be accessed by resource managers who have successfully logged in.

They can see a list of the resources available in the database. This list should be sorted based on the number of available quantity for the resource. Instead of displaying the exact quantity of resources that are available, the screen should only display either "Sufficient" (if resource has more than 5 items) or "Low stock" (if resource has between 1 to 5 items left) or "Not available" (if resource has no items left). If an item has 0 available quantity, then the background for this resource row in the table should be highlighted in red. The user can filter the list to show a subset of the resources based on either the type of resource or skill that the resource is assigned to. Both the list of valid resource types and skills should be

dynamically pulled from the database and users cannot select or input a value that that is not valid. The user can click on a row to select that particular resource, before clicking on the "Update" button to go to the screen to update the information about that particular resource, or clicking on the "Delete" button to remove this resource from the database. The user can also click on the "Add" button to open the screen to add a new resource to the database.

1.5 Create "5. Add resource"

Create the screen for the resource manager to add a new resource to the database, as outlined in "5. Add resource" in the wireframe.

This screen can only be accessed by resource managers who have successfully logged in.

Include appropriate errors checks for data entry. No two resources can share the same name. The list of valid resource types should be dynamically pulled from the database and users cannot select or input a type that that is not valid. If initial value for the quantity is set to 0, then the user cannot allocate this resource to any of the skills yet. Otherwise, the user can allocate this resource to none, one or more skill areas. Quantity cannot be set less than 0. After successfully adding a resource, the user remains on this screen, in case they wish to add another resource. The user can go back to the resource management page via the "Back" button.

1.6 Create "6. Update resource"

Create the screen for the resource manager to update the details for a specific resource, as outlined in "6. Update resource" in the wireframe.

This screen can only be accessed by resource managers who have successfully logged in.

Include appropriate errors checks for data entry. The resource name and resource type cannot be changed. If the quantity is changed to 0, then the user cannot allocate this resource to any of the skills. Otherwise, the user can allocate this resource to none, one or more skill areas. Quantity cannot be changed to a value less than 0. After successfully updating a resource, the user will be automatically re-directed to the resource management screen.