Cognizant Academy

Shipping Management

C#, ADO .Net Knock Out Challenge

Version 1.0

	Prepared By / Last Updated By	Reviewed By	Approved By
Name	Gnanasekar R		
Role			
Signature			
Date			

Table of Contents

1.0 Introduction

1.1 Purpose of this document

Shipping Management is used to insert shipping details to database, get shipping details by shipping id from database and calculate the shipping charges of products based on the weight, distance and priority. They get shipping details initially and calculate the shipping charges based on the logic.

Shipping Management current focus is to automate the below mentioned requirements.

1. Insert the Shipping Details.
2. Display the shipping details on the shipping id from the database.
3. Calculate shipping charges based on weight,distance and priority.

1.2 Definitions & Acronyms

Definition / Acronym	Description
Req	Requirement

1.3 Project Overview

This project captures the various concepts, techniques and skills learnt and helps to put them into practice using C# with ADO.NET. Admittedly, this would be at a scaled-down level since the purpose is to let the associate experience the various concepts learned in C# as an individual. The individual associate is expected to create a console-based application within the specified time.

1.4 Scope

The scope of the system is explained through its following modules

1. Insert the Shipping Details.
2. Display the shipping details on the shipping id from the database.
3. Calculate shipping charges based on weight,distance and priority.

1.5 Target Audience

Learner Level

1.6 Software Requirement

1.6.1 Software Requirements

#	Item	Specification/Version
1.	C#	6
2.	ADO.NET	4.5
3	SQLSERVER	2017

Note: All the required hardware and software will be provided in the Tekstac platform.

2.0 Functional Requirements

2.1 Functional Requirements

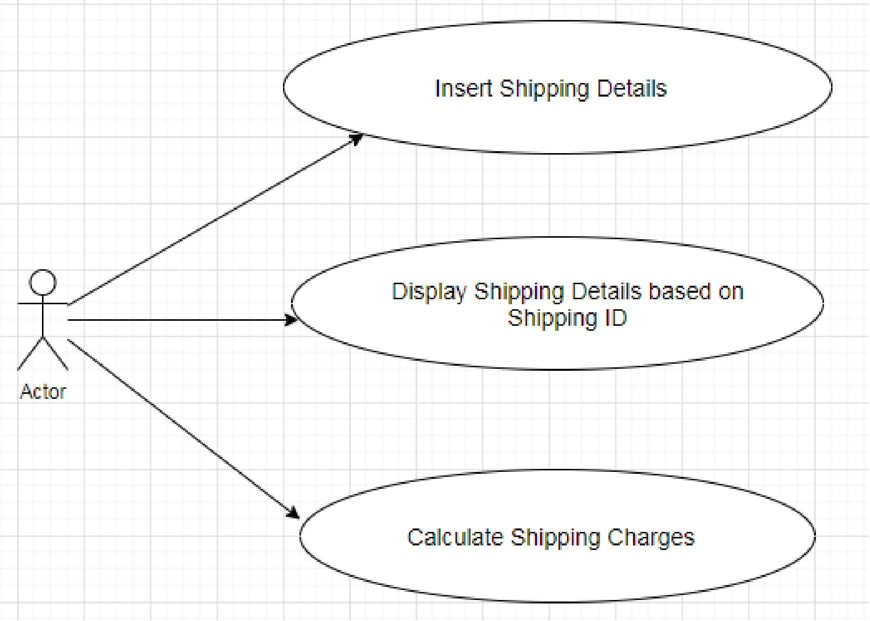
Req. #	1
Req. Name	Get the shipping details from the user . Insert the details to the database.
Req. Description	The shipping details are entered through the Main method. Assign the values to the ShippingDetails

	object. Insert the details to the Shipping table in the database.
Actors/ Users	Admin
Comments	The admin is responsible for inserting the data .

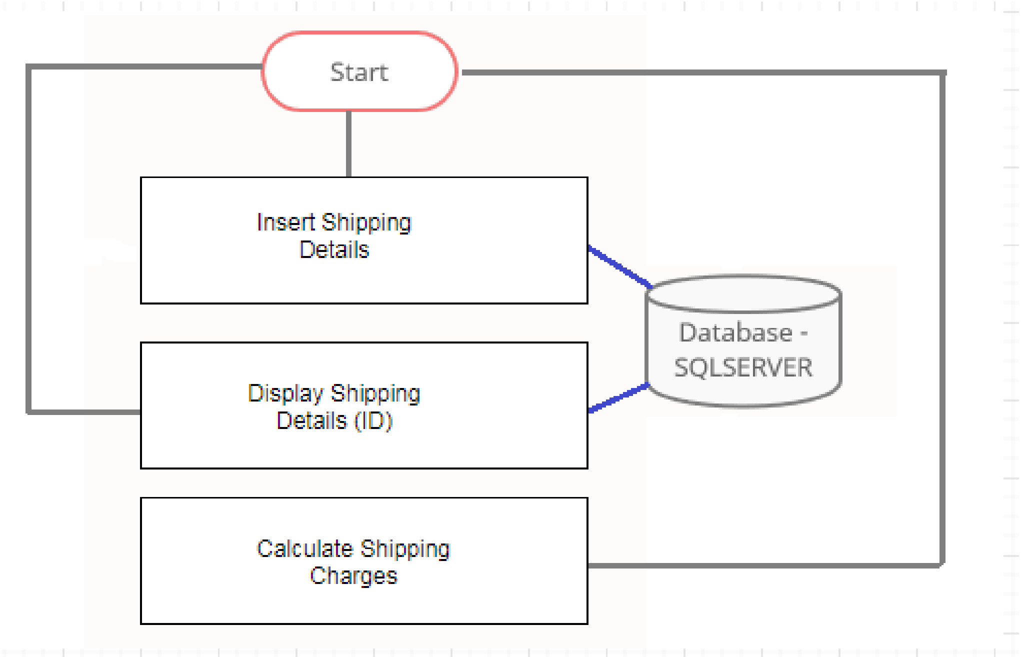
Req. #	2
Req. Name	Get the shipping id from the user and display all the details of that shipping id.
Req. Description	The shipping id is entered through main method. Pass the value to the method.
Actors/ Users	Admin
Comments	The admin is responsible for fetching thedetails from the database.

Req. #	3
Req. Name	Calculate shipping charges based on weight,distance and priority.
Req. Description	The user enter the weight,distance and priority. It will calculate the total charges.
Actors/ Users	User
Comments	The user is responsible for calculating the shipping charges.

2.2 Use case Diagram


Insert Shipping Details
Display Shipping Details based on Shipping ID
Calculate Shipping Charges
Actor

2.3 System Architecture Diagram



2.4 Sample Input/Output

```
1.Add shipping details
2.Get shipping details by shipping id
3.Calculate shipping charges
4.Exit
Enter your choice
1
Enter shipping id
1
Enter start location
Delhi
Enter end location
Chennai
Enter distance
5000
Enter start date
30/01/2021
Enter delivery date
02/02/2021
Enter priority
High
Details Added Successfully
```

```
1.Add shipping details

2.Get shipping details by shipping id

3.Calculate shipping charges

4.Exit

Enter your choice

1

Enter shipping id

2

Enter start location

Lucknow

Enter end location

Bangalore

Enter distance

6000

Enter start date

12/01/2021

Enter delivery date

16/01/2021

Enter priority

Medium

Details Added Successfully

1.Add shipping details

2.Get shipping details by shipping id

3.Calculate shipping charges
```

```
4.Exit

Enter your choice

2

Enter shipping id

2

ShippingId  StartLocation EndLocation Distance StartDate      DeliveryDate Priority

2                 Lucknow       Bangalore     6000       12/01/2021 16/01/2021 Medium

1.Add shipping details

2.Get shipping details by shipping id

3.Calculate shipping charges

4.Exit

Enter your choice

3

Enter weight

120

Enter distance

250

Enter priority

High

Shipping charges : 1200
```

3.0 Design Specification

3.1 Data Design

Table Structure:

Table name: Shipping	
Column Name	**Data type**
ShippingId	int(Primary Key)
StartLocation	varchar
EndLocation	varchar
Distance	int
StartDate	Date
DeliveryDate	Date
Priority	varchar

Design Constraints:

- Use **SQLSERVER** database to store the data. The database name is “dbTraineeHandsOn”. **This is already created for you in Tekstac.**
- The table names and the column names should be the same as specified in the table structure.
- The database connection information is specified in the “App.config” file, which is also provided as part of code skeleton. THIS IS GIVEN ONLY FOR YOUR REFERENCE. You need NOT change this.
- *Use the connection string name “SqlCon” for creating database connection(refer App.config).*

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
   <startup>
      <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6.1" />
   </startup>
  <connectionStrings>
<add name=" SqlCon"
connectionString="server=localhost;database=FlightDB;uid=XXXXX;password=XXXXXX;"/>
</connectionStrings>
</configuration>
```

Sample Data is Already inserted in the Shipping table.

ShippingId	StartLocation	EndLocation	Distance	StartDate	DeliveryDate	Priority
3	Bangalore	Kolkata	5500	15/02/2021	20/02/2021	High
4	Chennai	Mumbai	1500	19/02/2021	25/02/2021	High
5	Punjab	Harayana	200	19/02/2021	20/02/2021	Low

Note: The code skeleton will be available in the Tekstac platform

3.2 Component Details for identified Use Cases

3.2.1 Get data and insert into the database.

In the 'Main' get the following values,

ShippingId

StartLocation

EndLocation

Distance

StartDate

DeliveryDate

Priority

3.2.2 Get Shipping Id and Retrieve Details from database.

The User enters the Shipping id then it should get details of the Shipping product that matches the shipping id and return shipping details as an object

Example: If shipping Id is 1 then it should retrieve,

ShippingId	StartLocation	EndLocation	Distance	StartDate	DeliveryDate	Priority
1	Delhi	Chennai	5000	30/01/2021	02/02/2021	High

3.2.3 Calculate Shipping Charges

The User enters the weight,distance and priority and the method should calculate the shipping charges and return it.

Weight	Distance	Charges
Weight <=50	Distance <=100	200
	Distance >100 and Distance <=350	400
	Distance >350	700
Weight > 50 and Weight <=100	Distance <=100	300
	Distance >100 and Distance <=350	600
	Distance >350	1050
Weight > 100 and Weight <=500	Distance <=100	500
	Distance >100 and Distance <=350	1000
	Distance >350	1750
Weight > 500	Distance <=100	600
	Distance >100 and Distance <=350	1200
	Distance >350	2100

If Priority is High then add 20% to the charges

If Priority is Medium then add 10% to the charges

If Priority is Low there is no extra charges

Example: Weight = 50,Distance = 100 and

Priority = Medium

Shipping Charges = 200 + 10% extra charges

Shipping Charges = 220

Example: Weight = 500,Distance = 1000 and

Priority = Low

Shipping Charges = 1750 + No extra charges

Shipping Charges = 1750

3.3 Component Specification

Class Name : ShippingDetails(model class)

Responsibility:

This model object holds the state of the Shipping detail at all point-in-time.

Type(Class)	**Properties**
ShippingDetails	**int** ShippingId **String** StartLocation **String** EndLocation **int** Distance **DateTime** StartDate

	DateTime DeliveryDate **String** Priority

Note: Keep all the properties 'public'.

Class Name : ShippingUtility (utility class)

3.3.1 InsertShippingDetails Method

Type(Class)	Method	Responsibilities
ShippingUtility	public void InsertShippingDetails (ShippingDetails shippingdetailsobj)	This method should accept the ShippingDetails object and execute an sql query to insert the details into the database.

3.3.2 GetShippingDetailsByShippingId Method

Type(Class)	Method	Responsibilities
ShippingUtility	public ShippingDetails GetShippingDetailsByShippingId(int shippingId)	This method should get the shipping details of the shipping id passed as an argument that is in the database. Then return the details as ShippingDetails object.

3.3.3 CalculateShippingCharges Method

Type(Class)	Method	Responsibilities
ShippingUtility	public double CalculateShippingCharges (int weight,int	This method is used to calculate the shipping charges based on the weight,distance and priority.

	distance,String priority)	

Class Name : DBConnection (DAO class)

3.3.4 GetConnection Method

Responsibility:

This method should connect to the database by reading the database details from the *App.config* file and it should return the connection object.

Type(Class)	Method	Resources
DBConnection	public SqlConnection GetConnection()	App.config file contains the database connection details.

3.4 General Design Constraints

1. The properties/method/class name should be correctly specified as given in the document.
2. Keep all the classes as ‘**public**’
3. Do not change the App.config file.
4. Do not change the namespace name.

4.0 Submission

4.1 Code submission instructions

1. Do not change the code skeleton given as your code will be auto evaluated.
2. You can validate your solution against sample test cases during the assessment duration.
3. Your last submitted solution will be considered for detailed

evaluation.

4. Make sure to submit the solution before the time limit. After the assessment duration you will not be allowed to submit the solution.

5.0 Change Log

<table>
<tr><th></th><th colspan="4">Changes Made</th></tr>
<tr><td>V1.0.0</td><td colspan="4">Initial baseline created on <dd-Mon-yy> by <Name of Author></td></tr>
<tr><td rowspan="3">Vx.y.z</td><td colspan="4"><Please refer the configuration control tool / change item status form if the details of changes are maintained separately. If not, the template given below needs to be followed></td></tr>
<tr><th>Section No.</th><th>Changed By</th><th>Effective Date</th><th>Changes Effected</th></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>

6.0 Evaluation Areas

S.No	Description
1.	Declaration of properties in the class ShippingDetails
2.	Declaration of methods in the class DBConnection and ShippingUtility
3.	Implementation to create a valid database connection object
4.	Implementation to insert Shipping details to database
5.	Implementation to retrieve a specific shipping details from database
6.	Calculate the shipping charges based on the given business rules.