Web Development Technologies Assignment 2 specification
Deadlines: Multiple deadlines
To be attempted via: Portedu Marks allocated to
***You must attempt this assignment individually* or in group of 2-3**

If you decide to work in a group, please create a group on PORTEDU- this is very important. Failing to create a group on Portedu will lead to an award of ZERO Marks to your group members.

Note: If you are working individually, please skip to page 2

Group Owner is the student who creates the group. Only group owner CAN submit the assignment on the behalf of the group for marking.

Group Member is a student who joins an existing group.

Group Formation Closing Time - Students may create/join/leave/delete group before the group formation closing time. System will not allow these operations after group formation closing time. Students must contact the course instructor if they need to change their group memberships after group formation time.

Create New Group - Student can create a new group if not a member of any group. Some students may want to attempt group assessments as individual. These students don't have to create a group in the system.

1. Select Course
2. Click Assessment (Assignment 2 - Milestone 1) and click on My Group menu item.
3. System will present the list of all group assessments for the selected course.
4. Click on Create link next to the assessment for which you wish to create a group.
5. Confirm your request to create a new group and submit.
6. System will present operation result message.
7. Click on View link to view your group id and group secret code. You will have to securely pass on your group id and group secret code to students who you wish to take members in your group.

Join Group - To join an existing group you need the group id and group secret code of that group. The group owner or an existing group member should securely pass on this information to you.

1. Select Course.
2. Click Assessment and click on My Group menu item.
3. Click Join Group link next the assessment for which you wish to join an existing group.
4. Enter Group Id and Group Secret Code given to you by the group owner or existing group member. Click on submit button.
5. Confirm the group details and submit your request.
6. System will present operation result message.

Now repeat all of the above steps for Assignment 2 - Milestone 2.

2.1: Introduction

"The aim of this assignment is to develop a website with .NET Framework 4.5.1 using Visual Studio 2013 and SQL Server 2012/14 as the backend.

You can use either the Web form model or ASP.NET MVC model!

NWBA (National Wealth Bank of Australasia) has hired you to design their Internet Banking website. It is a simulated banking application. When complete, one should be able to:

- Check balances
- Modify a personal profile
- Simulate transactions such as deposits and withdrawals
- Schedule payments and,
- Perform some administrative tasks (*Only the authorised user*).

The following diagram depicts the logical architecture:

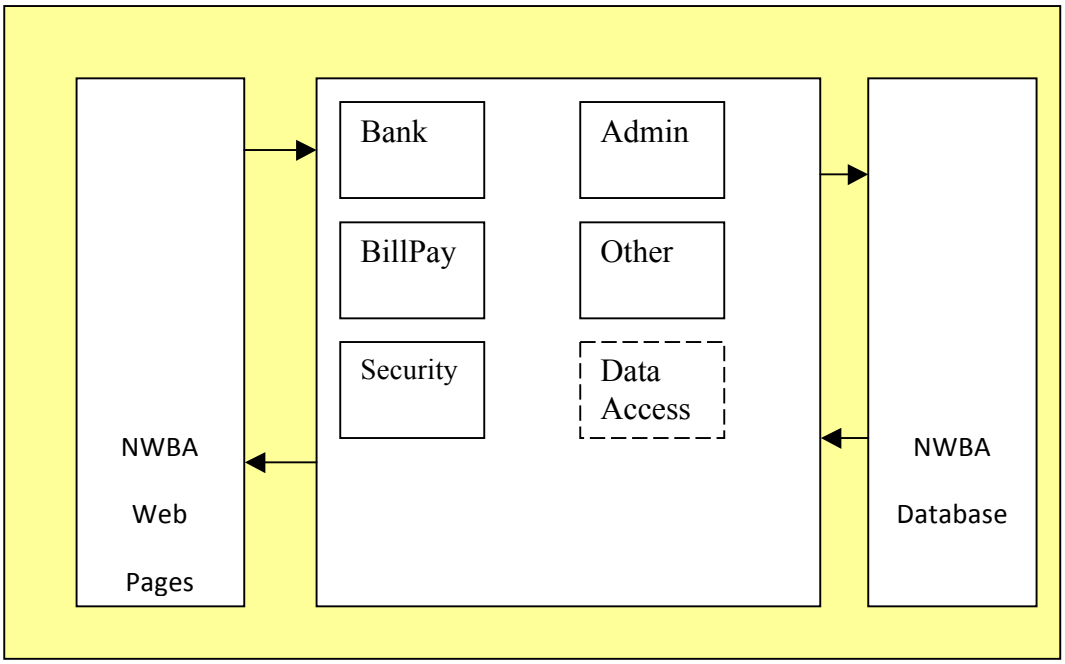

Figure 1

The architecture is a typical three-tiered, distributed application.

Layer I: User Interface (Presentation Layer) - will contain all the "aspx" or appropriate view pages in case if you are using MVC (and css, in case you decide to use style sheets) pages. No functionality is to be written in these pages.

Layer II: Business logic/ middle tier/object layer: All the functionality must be present in this layer. You will use code-behind approach (".aspx.cs" files) or Control pages if you are

using MVC. It is up-to-you how you want to write these business objects. You can either use plain code behind approach or you can use component-oriented approach where each of the business objects is a component (Code Library- when you use Visual Studio). The description of the business objects is as follows:

`Bank`- this will handle all the customer and transaction information.

`BillPay`- this will handle the scheduled payment functionality.

`Security`- this will handle the user authentication and provide accounts with suitable capabilities (details later on)

`Admin`- this will handle all the administrative functionalities

`Other`- any extra functionality that you may want to add (Optional)

`Data Access`- This will provide all the functionality one needs to connect to database and retrieve information.

Layer III: Database/Data tier- this comprises of your SQL Server 2012/14 database (tables, etc.)

2.2: General overview of the database involved

This is only a suggestion, feel free to modify the database structure, however a database with too few tables will attract penalty.

You may use ADO.NET or Entity Framework to fetch the data!

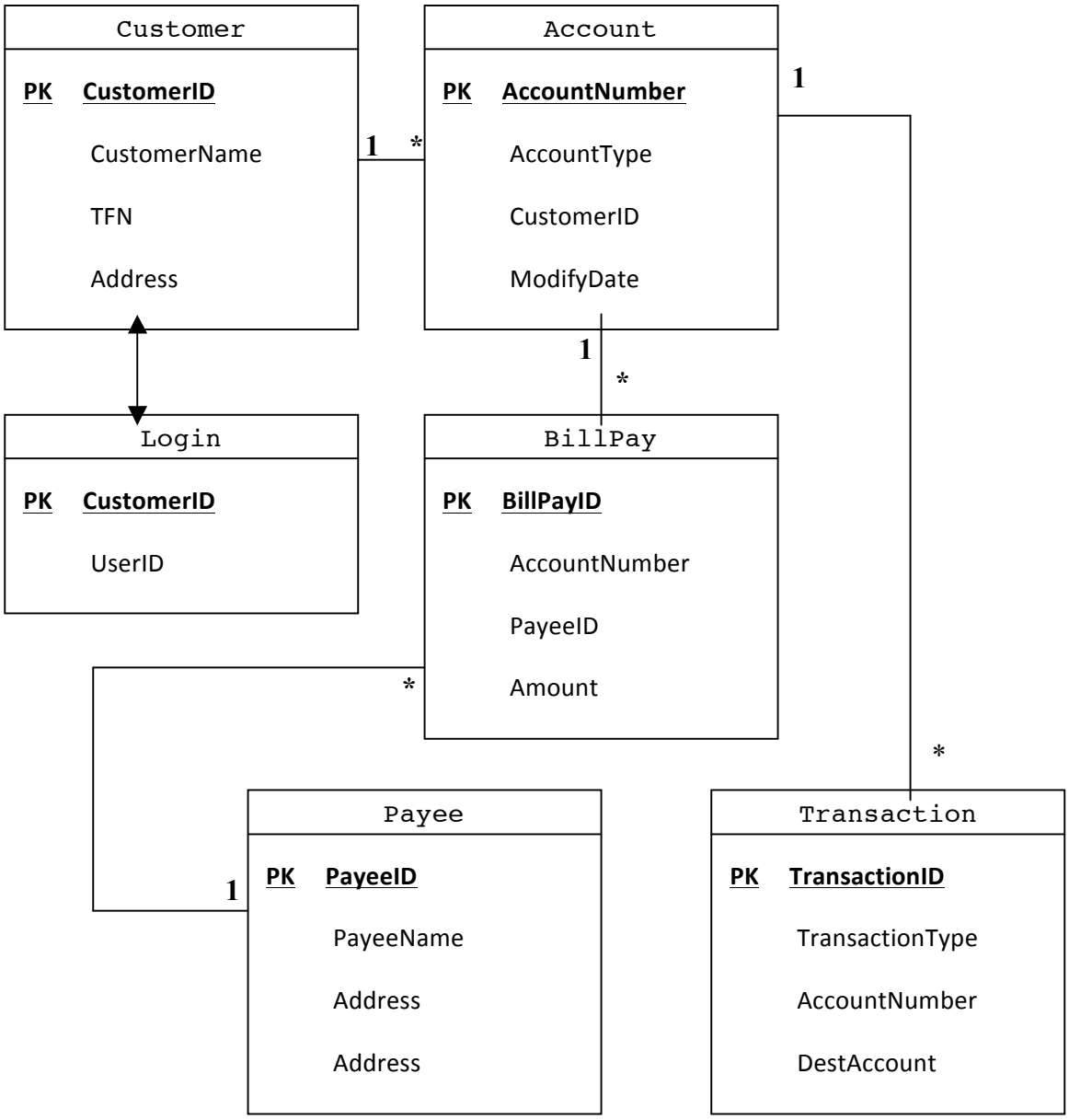

Figure 2

Customer Table

Field name	Data Type	Length	Description	Format	Allow Null
CustomerID	int	4	A unique ID for each customer		NOT NULL
CustomerName	nvarchar	50			NOT NULL
TFN	nvarchar	11	Tax File		

			Number- this is just for identification purposes- *no tax implications*		
Address	nvarchar	50			
City	nvarchar	40			
State	nvarchar	20		Must be 3 lettered Australian state	
PostCode	nvarchar	10		Must be a 4 digit number	
Phone	nvarchar	15		Must be of the format: (61)- XXXX XXXX	NOT NULL

Login Table

Field name	Data Type	Length	Description	Format	Allow Null
CustomerID	int	4	Refers back to customer		NOT NULL
UserID	nvarchar	50			NOT NULL
Password	nvarchar	20		Must be stored in an encrypted format	NOT NULL
ModifyDate	datetime	8	Last date/time the UserID and Password were modified		NOT NULL

Account Table

Field name	Data Type	Length	Description	Format	Allow Null
AccountNumber	int	4	Auto-generated unique ID for the account; also the customer's actual account number		NOT NULL
AccountType	nvarchar	1	The type of account such as checking or savings account	"C" or "S"	NOT NULL
CustomerID	int	4	Customer's		NOT

			unique ID		NULL
ModifyDate	datetime	8	Last date/time record was modified		NOT NULL

BillPay Table

Field name	Data Type	Length	Description	Format	Allow Null
BillPayID	int	4	Auto-generated unique ID for each billpay created		NOT NULL
AccountNumber	int	4	Source account number to withdraw funds from		NOT NULL
PayeeID	int	4	Payee to send payment to		NOT NULL
Amount	money	8	Amount of funds to be taken from the account		NOT NULL
ScheduleDate	datetime	8	Next schedule date for transaction to occur		NOT NULL
Period	nvarchar	1	How often will this occur?	Monthly (M), Quarterly (Q), Annually (Y) or Once off (S)	NOT NULL
ModifyDate	datetime	8	Last date/time record was modf.		NOT NULL

Payee Table

Field name	Data Type	Length	Description	Format	Allow Null
PayeeID	int	4	Auto-generated unique ID		NOT NULL
PayeeName	nvarchar	50			NOT NULL
Address	nvarchar	50	Payee's address		
City	nvarchar	40			
State	nvarchar	20		Must be 3 lettered Australian state	
PostCode	nvarchar	10		Must be a 4 digit number	
Phone	nvarchar	15		Must be of the format: (61)- XXXX XXXX	NOT NULL

Transaction Table

Field name	Data Type	Length	Description	Format	Allow Null
TransactionID	int	4	Auto-generated unique ID for each transaction type		NOT NULL
TransactionType	nvarchar	1	Type of transaction taking place	Refer to code table on next page	NOT NULL
AccountNumber	int	4	Source account number		NOT NULL
DestAccount	int	4	Destination account- used for transfers		
Amount	money	8	Amount of credit or debit		
Comment	nvarchar	255	Any comments the banker added to the transaction		
ModifyDate	datetime	8	Last date/time record was modified		

Transaction types:

D = Credit (Deposit money)
W = Debit (Withdrawal)
T = Debit (Transfer)
S = Debit (Service Charge)* - see business rules for more details on this
B = Debit (BillPay)

Thus in a nutshell,

Customer table is used to store all the customer information. We will distinguish between bank customers and payees by creating a separate Payee table. Only the bank's own customers will be held in the Customers table.

Account table is used to store all the information about the different accounts that any customer may have.

Transactions table us used to track all credit and debit items. Credit items will include transactions such as a customer's deposit of funds, a bank-applied credit (i.e. if any). Debit items will include items such as paying out on a check written by a customer, a monthly service charge, transfer of funds, ATM withdrawals, and so on. Thus, for each credit add the amount to the balance and for each debit, subtract the funds from the account.

BillPay table is used to schedule automatic payments to a third party such as a telephone bill, electricity bill, or a home mortgage company. You can schedule these payments to be made monthly, quarterly, annually, or on a specific date, as a one-off payment.

Login table is used to store customer's username and encrypted password so they can log in to the web site.

Business Rules:

1. NWBA does not pay any interest on any bank account.
2. Only two types of accounts are allowed- Checking and Savings.
3. Service charge is applied as follows:

Fee Type	**Fee (A$)**
Internet Transaction History	0.20
Internet Funds Transfer	0.20
Internet BPAY	0.30
ATM transactions	0.20

4. Multiple fund transfers are not allowed.
5. Only 4 free transactions are allowed.
6. For simplicity sake, no government taxes apply on any account.
7. Negative balances are not allowed.
8. All the dates must be displayed in the dd/mm/yy format.
9. A user can have both the savings and checking account.
10. Minimum balance to open a savings account = A$100 and A$500 for checking account.
11. Minimum balance allowed in a savings account = A$0 and A$200 for checking account.
12. Joint accounts are not allowed.

Assignment 2 Milestone 1 Tasks

Deadline: Sunday August 3, 2014 (11:59 pm Melbourne Time)

Marks allocated: 25

If you are working in the group, only the group owner needs to submit via PORTEDU.

1) (10 marks)

Every user must be presented with a login page (start page), which should have some introductory information about the bank and a form to log in. Also provide a sign off link.

Every logged in user to should first see the ATM page. This page mimics the ATM functionality.

Each page must have a navigation bar, which should have the following links-

Deposit/Withdraw (ATM page)

My Statement (to display transaction history)

My Profile (Account management)

A rough (suggested) layout of the ATM page is as follows:

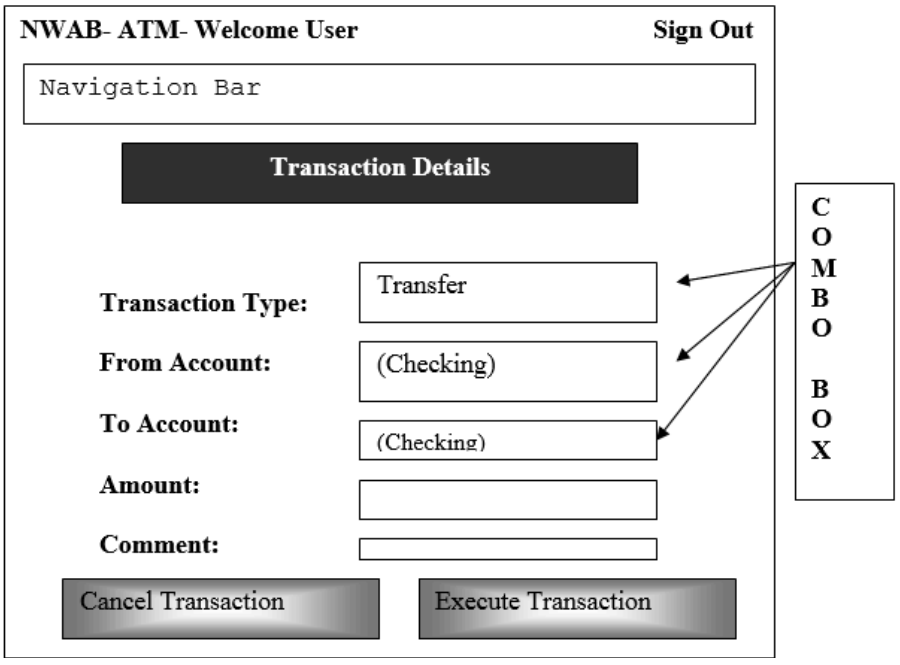

Figure 3

2) (5 marks)

My Statement page allows the user to see the current balance of a specific account and get a listing of all the transactions. User should first be presented with an option to choose the account type- checking or savings (drop-down box). Upon the selection of the type of account, the transactions must be displayed in a paged control (only 4 transactions per page).

3) (5 marks)

My Profile page should be used to modify user's information. It should have the form controls to edit information as- Name, Address, City, State, Post Code, Phone and TFN.

It should also have the functionality to change the password. Any changes made must be written back to the database.

4) (5 marks)

Apply the format constraints mentioned in the database tables.

What to submit?

Zip all of the following together and submit 1 zipped file via Portedu to the assessment labelled **Assignment 2 - Milestone 1 (25%)**-

1- Documentation Table (a template has been provided separately)
2- Readme file (*if applicable*)
3- The whole project directory
4- Sql file to create database

Extension queries

Send the extension queries to Maree Graham at ouacsit@rmit.edu.au .

Make sure you discuss queries at the Discussion Board and attend collaborate sessions for any help with the assignment.

Plagiarism is a serious offence. If you use any work borrowed from a book or net, you must reference it.

Assignment 2 Milestone 2 Tasks

Deadline: Sunday August 24, 2014 (11:59 pm Melbourne Time)

Marks allocated: 10

If you are working in the group, only the group owner needs to submit via PORTEDU.

5) (5 marks)

Add another link to the navigation bar- Pay Bills. Thus implement the bill pay option. When use clicks at this link, he/she is taken to a page where bill pays can be scheduled. A rough (suggested) layout of this page is as follows:

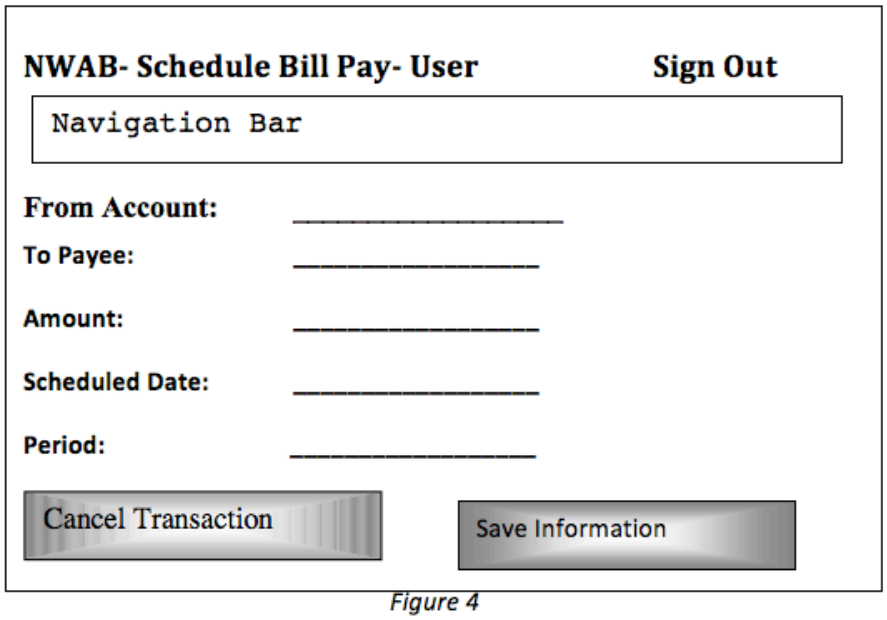

Figure 4

Implement another page, which displays all the scheduled bill pay information. This page should display the information in a grid-like manner, in which one the columns must have a modify button. A rough layout is shown as below:

	Payee	Amount	Schedule Date	Period
Modify	Telstra	$24.99	3/9/2006	One Time
Modify	RMIT	$1000	10/12/2006	Annual

Figure 5

Once the user clicks at the Modify button, he /she should be taken to form shown in figure 4.

6) (5 marks= 1 + 1 + 3)

Add the admin functionality to the project. Admin page should have a different look and feel.

After the administrator logs in using the login page, he/she is taken to the administrative page, where the following functions can be performed:

a- List account details (for a particular user)
b- List transaction details (for a particular user)
c- Stop scheduled payments

No layout has been provided here; feel free to create one yourself.

What to submit?

Zip all of the following together and submit 1 zipped file via Portedu to the assessment labelled **Assignment 2 - Milestone 2 (10%)**-

1- Documentation Table (a template has been provided separately)
2- Readme file (*if applicable*)
3- The whole project directory
4- Sql file to create database

Extension queries

Send the extension queries to Maree Graham at ouacsit@rmit.edu.au .

Make sure you discuss queries at the Discussion Board and attend collaborate sessions for any help with the assignment.

Plagiarism is a serious offence. If you use any work borrowed from a book or net, you must reference it.