

Thought Process / Plan

- 1) Create pseudo, admin accounts. (assume: admin accounts already exist and will be recognized by the system created)
- 2) Create login UI for support operatives (assume: is stored separately to customer data) - i.e. allow entering username & password
- 3) Login details have to pass validation (i.e. match details of login & password to a separate support operatives' admin account database)
- 4) Admin details (passwords) can also be hashed (*can be achieved by Generating a long random salt using a CSPRNG – e.g. using RNGCryptoServiceProvider() and then concatenate the entered password to the salt and hash it with a hashing function (e.g. the PBKDF2 hashing algorithm))
- 5) Login button leads to customer data UI
- 6) data UI - Enter data (input & edit option)
 - I. Name, Address, Phone number, Email address, Username, Password
 - II. After password is entered. It is stored as a hash (*)
- 7) data UI - View data in table:
 - I. Name, Address, Phone number, Email address, Username, Password – can be viewed in a table after inputted (assume: support operatives cannot/can only view hashed version of password – because its weird to be able to view customer passwords.)
- 8) Input & other validations to ensure accurate content and no input errors (e.g., predefined password requirements as specified (must be 10 characters minimum, contain 1 Upper, 1 lower, 1 number and 1 non alpha numeric character: can be achieved by using Microsoft.AspNet.Identity), email address has to contain '@', phone number should be certain number of digits and appear to be valid)

Unit tests could be completed for each of the points above. Some examples can be seen below.

Tested Examples:

(please note that I understand Visual Studio is capable of enabling its own unit tests, however due to time purposes I was unable to learn to do this)

This shows UI tested for 6(i) and 7(i):

Customer Support Desk

Please enter the following customer details:

Name:

Address:

Phone:

Email:

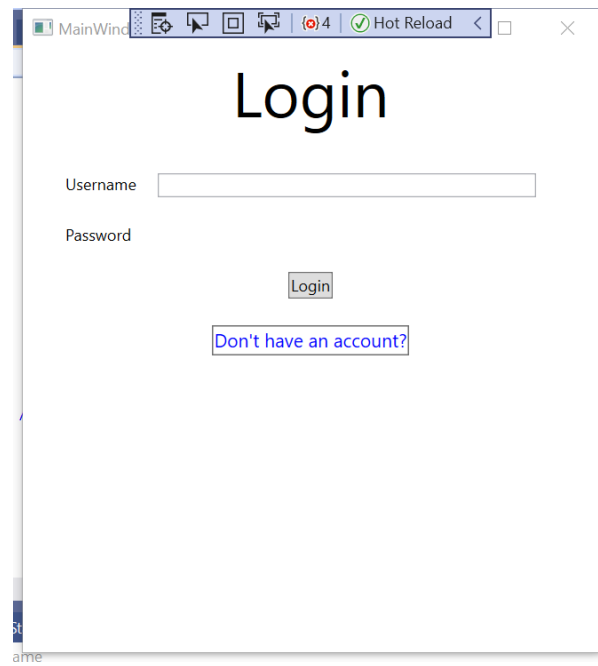
Username:

Password:

Customer Name	Customer Address	Customer Phone Number	Customer Email	Customer Username	Customer Password
Jojo	25 orchard	447505565551	jojo@gmail.com	Jojo123	Jojo123_123

<
>

This shows the incomplete UI tested for 2)



To run code for this, please switch to login view in the MainWindow.xaml

```
xmlns:vw="clr-namespace:customersupport.View" >
<!-- xmlns:vm="clr-namespace:customersupport.View"-->

<Grid>
  <vw:AddCustomerView></vw:AddCustomerView>

  <!-- <vm:LoginViews></vm:LoginViews> -->
```