

# Customer Relationship Management (CRM)

- **What is CRM?**

**Customer relationship management (CRM)** is a technology to manage and analyze customer interactions throughout the customer lifecycle, with the goal of improving business relationships with current customers and potential customers.

By:

Vajiheh Seyedzadeh

Winoto Janputra

Fall - 2017

# Background

- Mini CRM to Record current Customer information and potential customers
- Record every interaction between Customer and Sales Person
- Ability of show works day of Sales Person
- Ability to query all data with read only access
- As a Manager, track Sales Person activity
- Show reports as a data table and diagram with ability of export in to excel file and print in to
- The goal is simple: Improve business relationships and increase efficiency.

# My Works Day :

- show list of appointments

The screenshot shows the My Works Day application interface. At the top left is a 'Login' dialog box with fields for 'User Name' and 'Password', and buttons for 'Login' and 'Cancel'. Below it is the main application window titled 'MainWindow' with a dark header bar showing the date 'Monday, October 16, 2017' and time '4:32:09 PM'. The main area contains a sidebar with icons for 'MyWorksDay', 'AddressBook', 'Opportunities', and 'Reports'. The main content area displays a table of appointments:

Customer Name	Date	Start Time	End Time
winMiddle oto	2017/10/14	12:00 AM	3:00 AM
Active Green & Ross Complete Tire & Auto Centre	2017/10/26	3:00 AM	4:00 AM
Active Green & Ross Complete Tire & Auto Centre	2017/10/18	3:00 AM	4:00 AM
winoto	2017/10/16	1:00 AM	3:00 AM
Active Green & Ross Complete Tire & Auto Centre	2017/10/30	3:00 PM	4:00 PM
hepworth	2017/10/17	3:21 PM	5:21 PM

To the right of the table is a calendar for October 2017, with the 16th highlighted in red.

# Address Book

- is using to show all Customers and their information

The screenshot shows the Address Book application interface. At the top left is a 'MainWindow' window with a dark header bar showing the date 'Monday, October 16, 2017' and time '4:34:19 PM'. The main area contains a sidebar with icons for 'MyWorksDay', 'AddressBook', 'Opportunities', and 'Reports'. The main content area shows a list of 7 records in the 'Address Book':

Customer Name	Cust No	Status	Created On
Calcutta	2926	True	3/31/2015 12:00
Champions	2969	True	10/20/1999 12:00
Equities	3006	True	12/11/2015 12:00
Associates	3014	True	7/18/2016 12:00
winMiddle oto	3048	True	7/18/2016 12:00
winSenior oto	3050	True	7/18/2014 12:00
Test after add	3459	True	10/16/2017 12:00

To the right is a detailed 'AddCustomer' form:

Basic Info	Company Name: <input type="text"/>	Type: <input type="radio"/> Customer <input checked="" type="radio"/> Prospect
Phone Number	Main: <input type="text"/>	Email Address and WebSite: <input type="text"/>
Address	Street: <input type="text"/>	Province: <input type="text"/>
	City: <input type="text"/>	Country: <input type="text"/>
	Postal Code: <input type="text"/>	
Save Cancel		

Below the basic info section, there are tabs for 'Details', 'History', and 'Purchased'. The 'Details' tab is active, showing fields for First Name (Norton), Last Name (Casey), Company Name (Champions), Address (1 World Dr, Lakeview, qc, L5L 2B3, canada), Phone Number (5145145144), and Email Address (winotojanputra@gmail.com). There is also a note 'This Account belong to:' followed by an 'Edit' button.

# Report

- Number of Sales per Customer to excel and PDF
- Sales Per Sales Rep YTD

The screenshot shows a Windows application window titled "MainWindow". The title bar includes the application name, the user's name "Winoto Janputra", the date "Monday, October 16, 2017", and the time "4:38:55 PM". On the left side, there is a vertical toolbar with icons for "MyWorksDay", "AddressBook", "Opportunities", and "Reports". Below the toolbar, there are three report cards: "Number of Sales per customer", "Sales per Rep", and "Report3". The "Number of Sales per customer" card is currently active, displaying a table titled "Report1" with two buttons: "Export Excel" and "Print To PDF". The table lists various companies and their number of purchases:

Customer Name	Number of buy
2 Leos Food Inc.	1
3 A Graphics Corp.	9
A & P Convenience	1
A & T Human Resources	1
A B A A Beauty Supply	22
A C V Central Vacuum Corp.	1
A D M Milling Co.	8
A E C O M Canada Ltd.	1
A N & Associates	19
Ability Concepts Inc	1
Absolute Furnished Apartments by Mirage	1
Addison on Erin Mills	1
Adult Education, Mississauga Campus	1
Advance Travel	1
AJ Foods	1
Akhara of Champions	1
Alecca Auto Repair Shop	1
All Time Car Rental	1
Allan A. Martin Senior Public School	1
Alpha Express	1

The screenshot shows a CRM application interface. At the top, there is a header with the user's name "Winoto Janputra", the date "October 16, 2017", and the time "7:40:14 PM". On the left, there is a sidebar with icons for "MyWorksDay", "AddressBook", "Opportunities", and "Reports". The "AddressBook" section is currently active, showing a list of 7 records with columns for "Name", "Phone Number", and "Created on". The list includes entries like "Active Green & Ross Complete Tire & Auto Centre", "One World Auto Service Inc.", and "hepworth". To the right of the list is a search form titled "Address Book 7Record(s)" with fields for Company Name, First Name, Last Name, Type (radio buttons for Customer, Prospect, All), Phone, Email, Fax, WebSite, Street, City, Province, Postal Code, Country, Sales Rep (dropdown menu), and Last Purchase date before (date picker). There are also "Search" and "Cancel" buttons at the bottom of the search form.

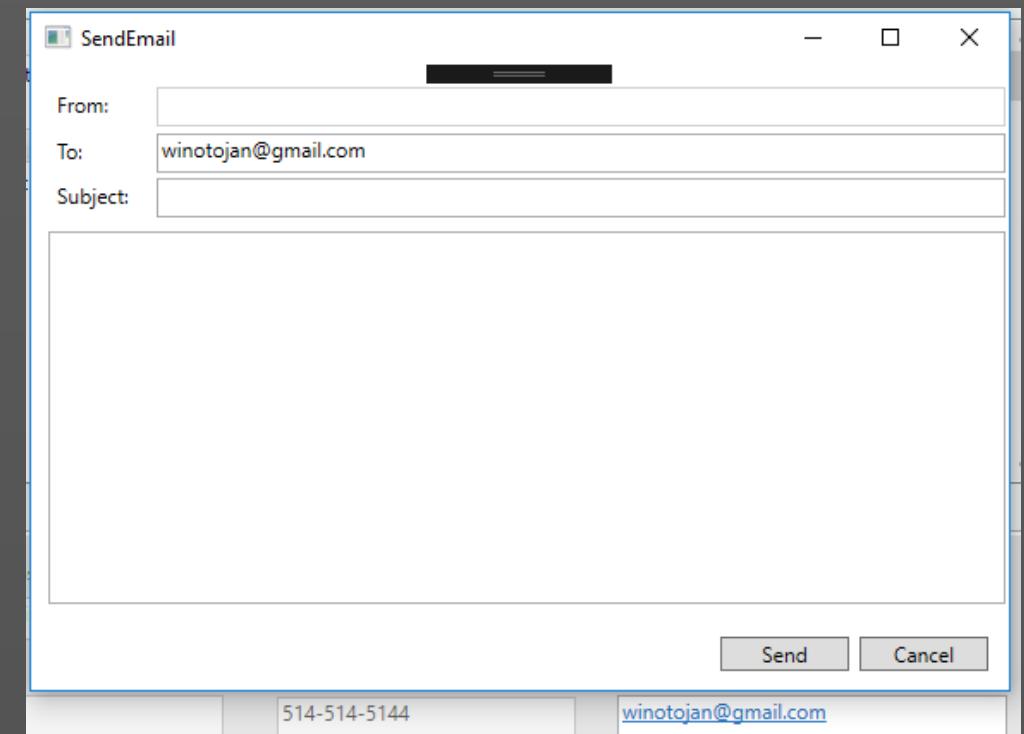
## Opportunities

- See potential customers list with its contact information, conversation
- Add new and Search prospect

# Challenges And Solutions

- Use Frame in Main Window
- Keep user id from login screen to be used in other pages
- Prevent user to close login screen without enter the information
- Send Email and Create Calendar using Outlook (Using Interop.Outlook)

```
Outlook._Application _app;  
  
try  
{  
    _app = (Outlook.Application)Marshal.GetActiveObject("Outlook.Application");  
}  
catch(Exception)  
{  
    _app = new Outlook.Application();  
}  
try  
{  
    Outlook.MailItem mail = (Outlook.MailItem)_app.CreateItem(Outlook.OlItemType.olMailItem);  
    mail.To = _Recipient;  
    mail.Subject = tbSubject.Text;  
    mail.Body = tbBody.Text;  
    mail.Importance = Outlook.OlImportance.olImportanceNormal;  
    ((Outlook._MailItem)mail).Send();  
    db.RecordMessage(tbSubject.Text, tbBody.Text, "Email", _CompanyId, int.Parse(_Sender));  
    tbSubject.Text = string.Empty;  
    tbBody.Text = string.Empty;  
    this.Close();  
}
```



- Changed the background of the calendar's day

## App.xaml

```

<Style TargetType="CalendarDayButton"
x:Key="CalendarDayButtonStyle">
    <Setter Property="MinWidth" Value="5" />
    <Setter Property="MinHeight" Value="5" />
    <Setter Property="FontSize" Value="10" />
    <Setter Property="HorizontalContentAlignment" Value="Center" />
    <Setter Property="VerticalContentAlignment" Value="Center" />
    <Setter Property="Template">
        <Setter.Value>
            <ControlTemplate TargetType="CalendarDayButton">
                <Grid>
                    <VisualStateManager.VisualStateGroups>
                        <VisualStateGroup Name="CommonStates">
                            <VisualStateGroup.Transitions>
                                <VisualTransition GeneratedDuration="0:0:0.1" />
                            </VisualStateGroup.Transitions>
                            <VisualState Name="Normal" />
                            <VisualState Name="MouseOver" ...>
                            <VisualState Name="Pressed" ...>
                            <VisualState Name="Disabled" ...>
                        </VisualStateGroup>
                        <VisualStateGroup Name="SelectionStates">
                            <VisualStateGroup.Transitions>
                                <VisualTransition GeneratedDuration="0:0:0.1" />
                            </VisualStateGroup.Transitions>
                            <VisualState Name="Selected" ...>
                            <Storyboard>
                                <DoubleAnimation Storyboard.TargetProperty="Background" To="LightBlue" Duration="0" />
                            </Storyboard>
                        </VisualStateGroup>
                    </VisualStateManager.VisualStateGroups>
                </Grid>
            </ControlTemplate>
        </Setter.Value>
    </Setter>
</Style>

```

```

public class AppointmentDayConverter : IValueConverter
{
    static Dictionary<DateTime, string> dict = new Dictionary<DateTime, string>();

    public static Dictionary<DateTime, string> Dict
    {
        get { return dict; }
    }

    public static void LoadAppointments(List<DateTime> date)
    {
        foreach (DateTime d in date)
        {
            var key = new DateTime(d.Year, d.Month, d.Day);
            if (!dict.ContainsKey(key))
            {
                dict.Add(key, "New Appointment");
            }
        }
    }

    public object Convert(object value, Type targetType, object parameter, CultureInfo culture)
    {
        string text;
        if (!dict.TryGetValue((DateTime)value, out text))
            text = null;
        return text;
    }

    public object ConvertBack(object value, Type targetType, object parameter, CultureInfo culture)
    {
        return value;
    }
}

```

The screenshot shows a Windows application window. At the top, there are two buttons: "Make New Appointment" and "Edit". Below them is a table listing ten appointments. To the right of the table is a calendar for October 2017.

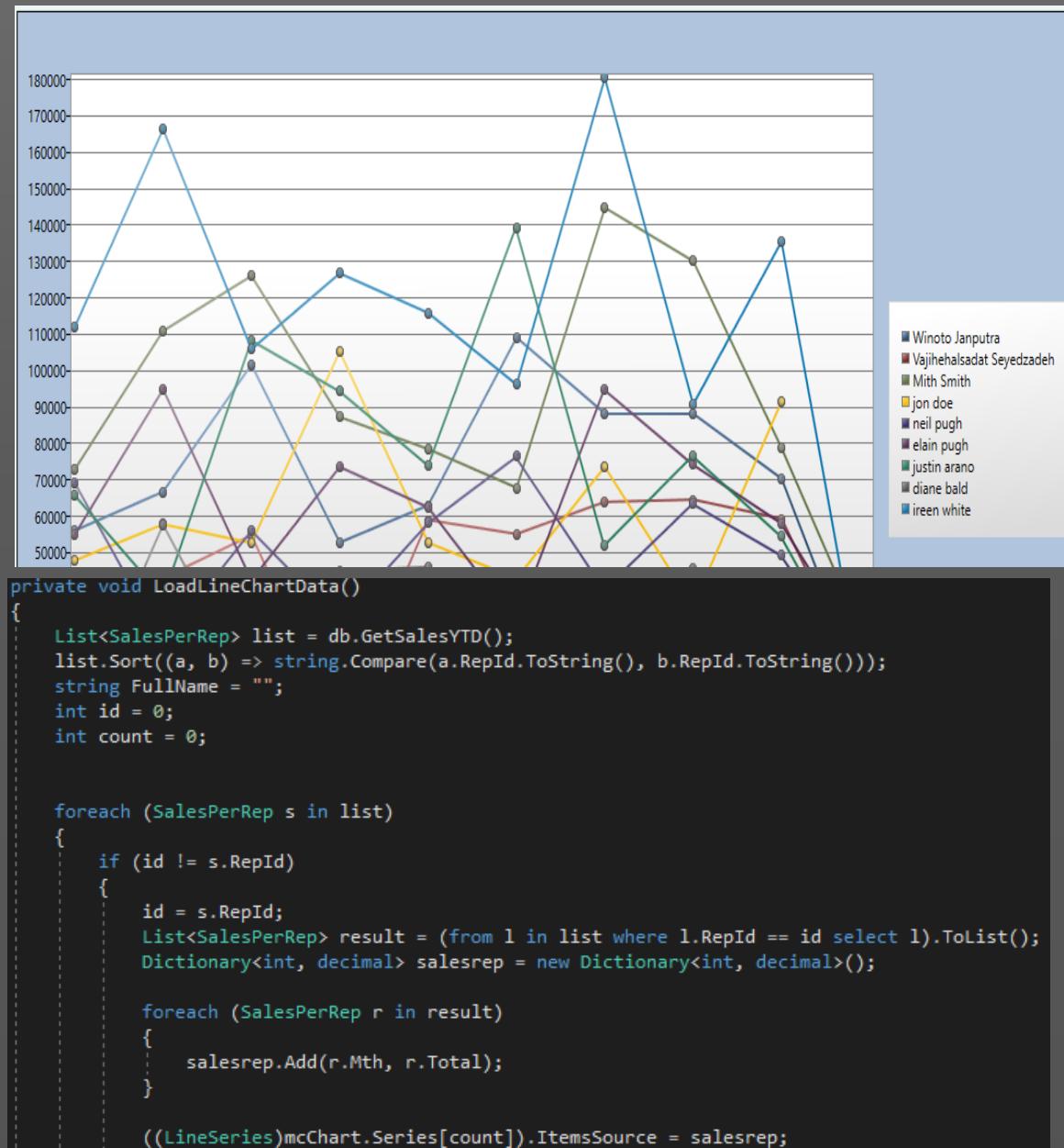
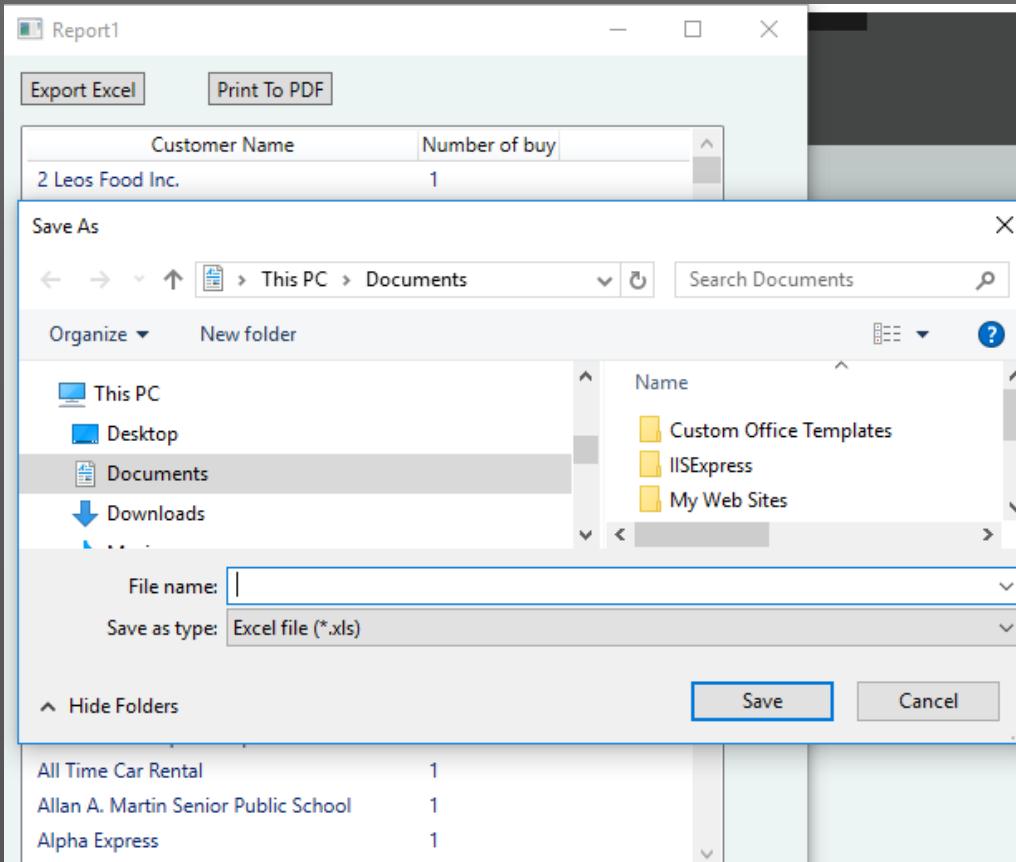
	Customer Name	Date	Start Time	End Time
<input type="checkbox"/>	winMiddle oto	2017/10/14	12:00 AM	3:00 AM
<input type="checkbox"/>	Active Green & Ross Complete Tire & Auto Centre	2017/10/26	3:00 AM	4:00 AM
<input type="checkbox"/>	Active Green & Ross Complete Tire & Auto Centre	2017/10/18	3:00 AM	4:00 AM
<input type="checkbox"/>	winoto	2017/10/16	1:00 AM	3:00 AM
<input type="checkbox"/>	Active Green & Ross Complete Tire & Auto Centre	2017/10/30	3:00 PM	4:00 PM
<input type="checkbox"/>	hepworth	2017/10/17	3:21 PM	5:21 PM
<input type="checkbox"/>	hepworth	2017/10/16	9:40 PM	10:10 PM
<input type="checkbox"/>	Active Green & Ross Complete Tire & Auto Centre	2017/10/18	11:42 PM	12:12 AM
<input type="checkbox"/>	winoto	2017/10/31	11:43 PM	12:13 AM

**Calendar View:**

Su	Mo	Tu	We	Th	Fr	Sa
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Appointment Day Convertor.cs

- Export to Excel and Print to PDF (Using DocumentFormatOpenXml and ITextSharp)
- Create Graphic with multiline



- Client side validation
- Sorting without MVVM

```

if (columnName == "ContactLastName")
{
    if (string.IsNullOrEmpty(ContactLastName) || ContactLastName.Length < 2)
        result = "Please enter a Name (2-50 Chars)";
}
if (columnName == "Email")
{
    Regex regex = new Regex(@"^[\w!#$%&'*+\-/=?\^_{}|~]+(\.[\w!#$%&'*+\-/=?\^_{}|~]+)*$");
    if (string.IsNullOrEmpty(Email))
        result = "Please enter an Email";
    else if (!regex.Match(Email).Success)
    {
        result = "Please enter a valid Email";
    }
}

```

Search Result: 312 Record(s)

Customer Name	Cust No	Status	Created on	Last Purch Date	Last
A B A A Beauty Supply	3187	True	10/31/2016 12:00:00 AM	2017-08-31 12:00:00 AM	
A E C O M Canada Ltd.	2898	True	7/21/2014 12:00:00 AM	2017-09-14 12:00:00 AM	
A N & Associates	2976	True	10/31/2015 12:00:00 AM	2017-06-12 12:00:00 AM	
Absolute Furnished Apartments by Mirage	3180	True	10/31/2016 12:00:00 AM	2017-06-17 12:00:00 AM	
Adult Education, Mississauga Campus	3028	True	7/18/2016 12:00:00 AM	2017-07-17 12:00:00 AM	
Adult Tutors	2104	True	8/8/2016 12:00:00 AM	2017-01-09 12:00:00 AM	

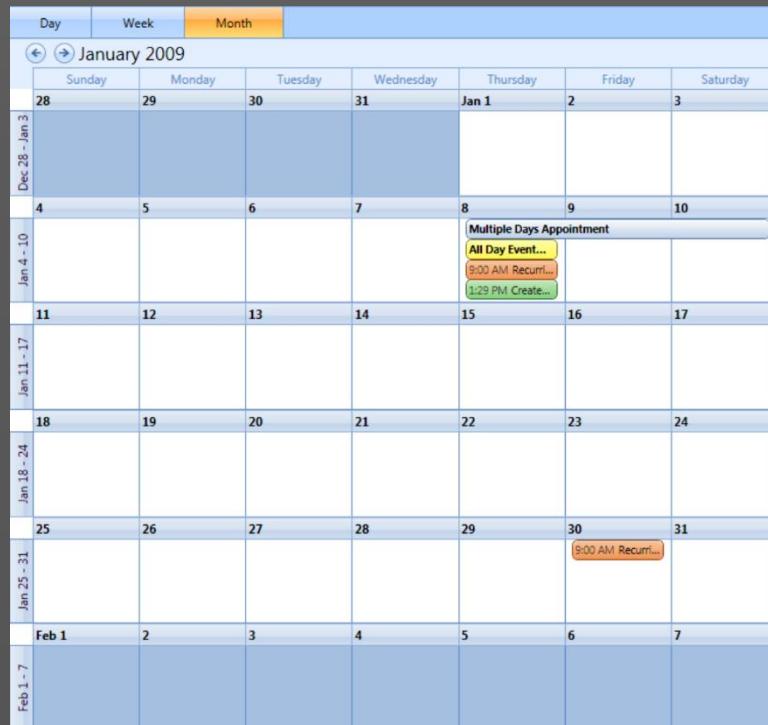
```

void GridViewColumnHeaderClickedHandler(object sender, RoutedEventArgs e)
{
    string header = ((GridViewColumnHeader)e.OriginalSource).Column.Header.ToString();
    List<Customer> list = db.GetAllCustomers();
    if (lastHeaderAddress == header)
    {
        switch (header)
        {
            case "Customer Name":
                list.Sort((x, y) => -1 * x.CompanyName.CompareTo(y.CompanyName));
                break;
            .....
        }
        lastHeaderAddress = string.Empty;
    }
    else
    {
        .....
        switch (header)
        {
            case "Last Purch Amount":
                list.Sort((x, y) => x.Amount.CompareTo(y.Amount));
                break;
            default:
                return;
        }
        lastHeaderAddress = header;
    }
    lvAddress.Items.Clear();
    foreach (Customer c in list)
    {
        lvAddress.Items.Add(c);
    }
}

```

# Future Works

- Localization(Localization is the translation of application resources into localized versions for the specific cultures that the application supports.)
- Installation
- Change the view of the Calendar (show the List of appointment like schedule )



## What we learned:

- Concept of Object Oriented Programming
- Using MVVM few part of the project (Separation of logic and presentation)
- Use different type of controls in WPF:
  - Data Display: DataGrid, ListView
  - Date Display and Selection: Calendar, DatePicker and TimePicker
  - Layout: DockPanel, StackPanel
  - Navigation: Frame, TabControl, Page, Window
- Make different Template and style for control

Thank you for your time!