

UWP-ו WCF

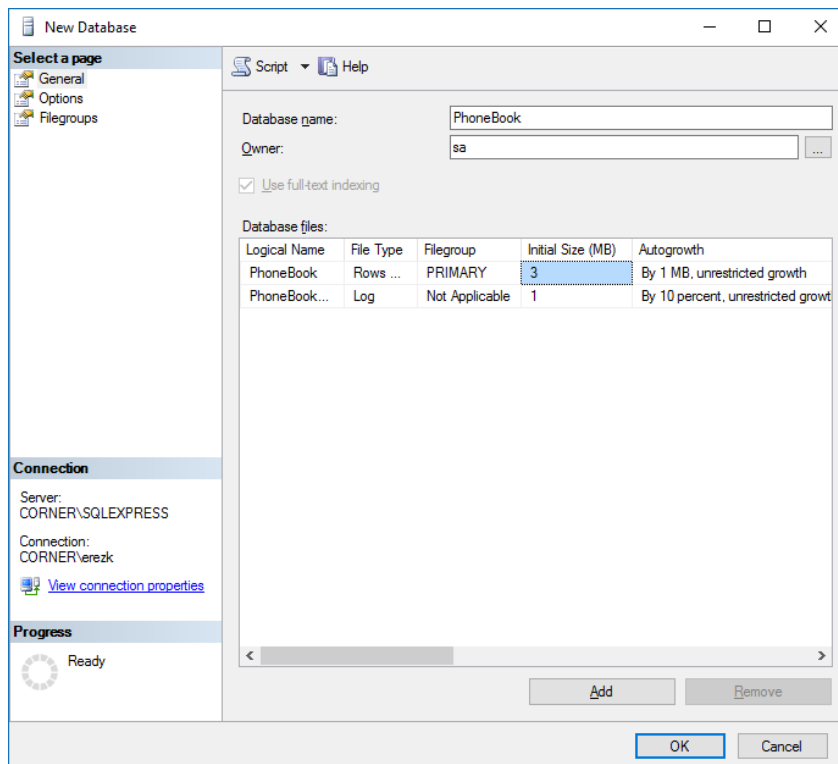
כאשר שומרים מידע במכשירים ניידים נרצה לגבות אותו במסד נתונים אשר נמצא השרת מרוחק. הדרך הפשוטה ביותר היא באמצעות שירות WCF.

היתרונות הם:

- גיבוי במידה והמכשיר מתקלקל או הולך לאיבוד.
- אפשרות לסנכרן מידי בין מכשירים.
- הורדת המידע למכשיר חדש.

1. מסד נתונים

תחילה נבנה מסד נתונים חדש:

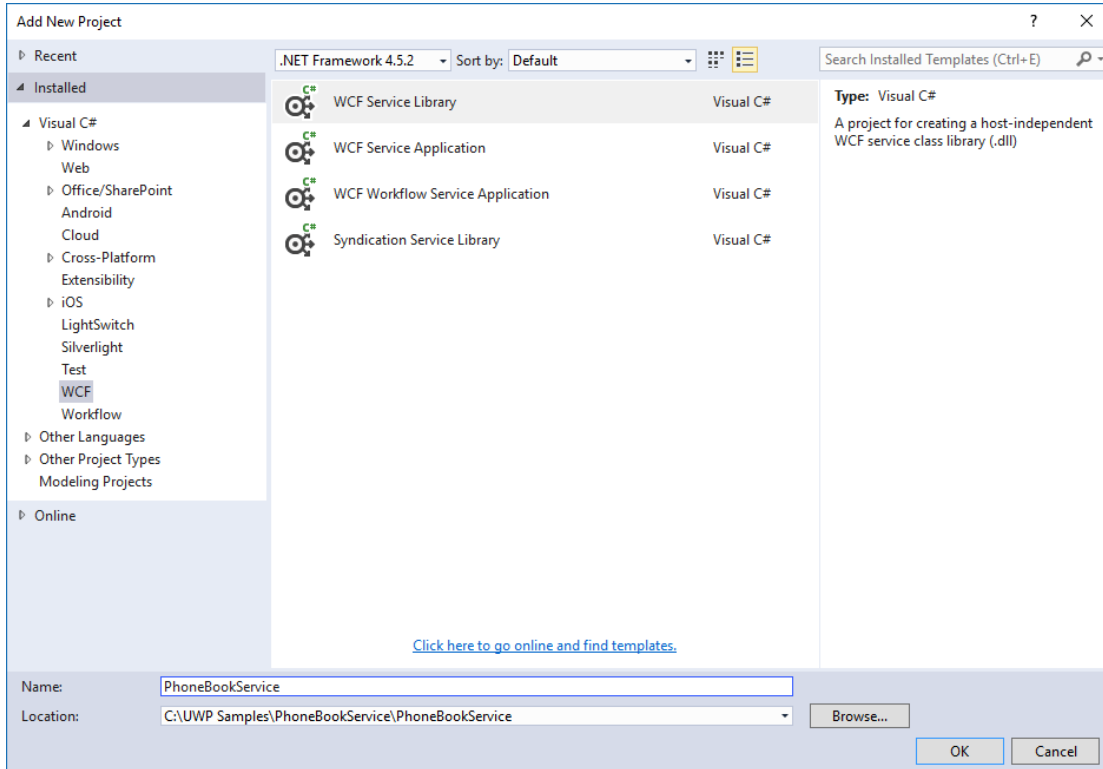


במסד נתונים נוסף טבלה אחת שנקראת Contacts (הערה – במציאות יהיו כמובן יותר טבלאות כולל טבלה המזהה את המשתמש משום שמסד הנתונים ישמש הרבה משתמשים).

Column Name	Data Type	Allow Nulls
ContactID	int	<input type="checkbox"/>
FirstName	nvarchar(20)	<input type="checkbox"/>
LastName	nvarchar(20)	<input type="checkbox"/>
DatOfBirth	datetime	<input checked="" type="checkbox"/>
PhoneNum	nvarchar(12)	<input type="checkbox"/>
		<input type="checkbox"/>

2. בניית שרות WCF

נגדיר פרויקט WCF Service Library חדש ששמו PhoneBookService



נמחק את הקבצים Service.cs ו-Service.cs

3. הגדרת DataContract:

נוסיף קובץ Contact.cs

```
[DataContract]
0 references
public class Contact
{
    [DataMember]
    0 references
    public int ContactID { get; set; }
    [DataMember]
    0 references
    public string LastName { get; set; }
    [DataMember]
    0 references
    public string FirstName { get; set; }
    [DataMember]
    0 references
    public DateTime DayOfBirth { get; set; }
    [DataMember]
    0 references
    public string PhoneNum { get; set; }
}
```

4. הגדרת ServiceContract:

נוסיף קובץ IPhoneService.cs

```

[OperationContract]
0 references
public interface IPhoneService
{
    [OperationContract]
    0 references
    bool SaveContact(Contact contact);
    [OperationContract]
    0 references
    bool DeleteContact(string id);
    [OperationContract]
    0 references
    bool UpdateContact(Contact contact);
    [OperationContract]
    0 references
    List<Contact> SelectAll();
}

```

5. נוסף את ה-ConnectionString ל-App.Config:

במקום לרשום אותו בכל פעם מחדש:

```

<appSettings>
  <add key="aspnet:UseTaskFriendlySynchronizationContext" value="true" />
  <add key="ConnectionString" value="Data Source=CORNER01\SQLEXPRESS;Initial Catalog=PhoneBook;Integrated Security=True"/>
</appSettings>

```

6. נגדיר את ה-Service:
נוסיף קובץ PhoneService.cs

```
0 references
public class PhoneService : IPhoneService
{
    1 reference
    public bool DeleteContact(string id)
    {
        throw new NotImplementedException();
    }

    1 reference
    public bool SaveContact(Contact contact)
    {
        throw new NotImplementedException();
    }

    1 reference
    public List<Contact> SelectAll()
    {
        throw new NotImplementedException();
    }

    1 reference
    public bool UpdateContact(Contact contact)
    {
        throw new NotImplementedException();
    }
}
```

7. נממש את ה-Service:

:Delete

```
public bool DeleteContact(string id)
{
    string connectionString = System.Configuration.ConfigurationManager.AppSettings["ConnectionString"];
    SqlConnection connection= new SqlConnection(connectionString);
    string query = string.Format("DELETE FROM Contact WHERE ContactID = {0}", id);
    SqlCommand cmd = new SqlCommand(query, connection);
    connection.Open();
    int result = cmd.ExecuteNonQuery();
    connection.Close();
    if(result!=0)
        return true;
    return false;
}
```

:Insert

```
public bool InsertContact(Contact contact)
{
    string connectionString = System.Configuration.ConfigurationManager.AppSettings["ConnectionString"];
    SqlConnection connection = new SqlConnection(connectionString);

    string query = string.Format("INSERT INTO Contact (FirstName, LastName, DayOfBirth, PhoneNum) VALUES ('{0}','{1}','{2}', '{3}') ",
        contact.FirstName, contact.LastName, contact.DayOfBirth,contact.PhoneNum);
    SqlCommand cmd = new SqlCommand(query, connection);
    connection.Open();
    int result = cmd.ExecuteNonQuery();
    connection.Close();
    if (result != 0)
        return true;
    return false;
}
```

:SelectAll

```
public List<Contact> SelectAll()
{
    string connectionString = System.Configuration.ConfigurationManager.AppSettings["ConnectionString"];
    SqlConnection connection = new SqlConnection(connectionString);
    SqlCommand cmd = new SqlCommand("SELECT * FROM Contact", connection);
    connection.Open();
    SqlDataReader sqlReader = cmd.ExecuteReader();
    Contact contact;
    List<Contact> list = new List<Contact>();
    while (sqlReader.Read())
    {
        contact = new Contact();
        contact.LastName = sqlReader["LastName"].ToString();
        contact.FirstName = sqlReader["FirstName"].ToString();
        contact.DayOfBirth = Convert.ToDateTime(sqlReader["DayOfBirth"].ToString());
        contact.ContactID = (int)sqlReader["ContactID"];
        list.Add(contact);
    }
    connection.Close();
    return list;
}
```

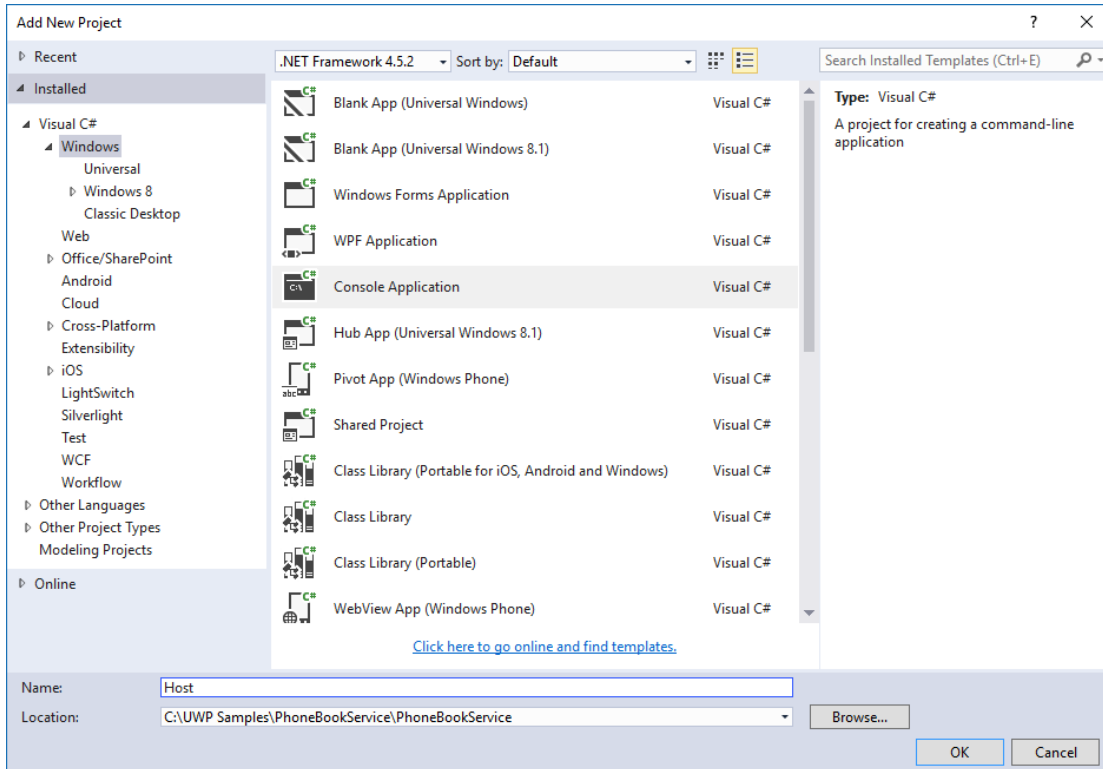
:Update

```
public bool UpdateContact(Contact contact)
{
    string connectionString = System.Configuration.ConfigurationManager.AppSettings["ConnectionString"];
    SqlConnection connection = new SqlConnection(connectionString);
    string query = string.Format("UPDATE Contact SET FirstName='{0}', LastName='{1}',DayOfBirth='{2}', PhoneNum='{3}' WHERE ContactID = {4}",
        contact.FirstName, contact.LastName, contact.DayOfBirth,contact.PhoneNum, contact.ContactID);
    SqlCommand cmd = new SqlCommand(query, connection);
    connection.Open();
    cmd.ExecuteNonQuery();
    connection.Close();
    return true;
}
```

8. נעדכן את ה-AppConfig

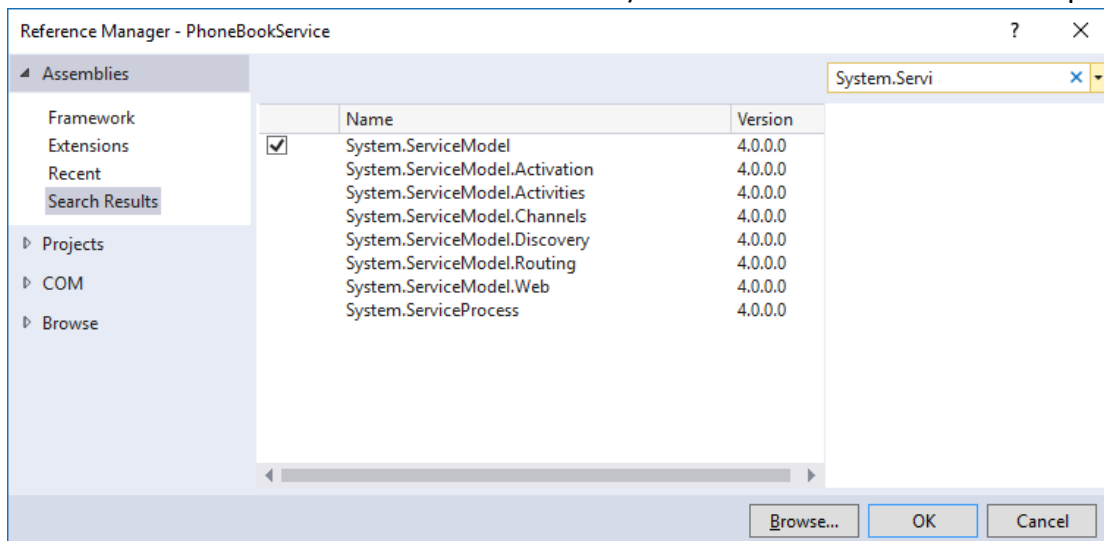
```
<services>
  <service name="PhoneBookService.PhoneService">
    <host>
      <baseAddresses>
        <add baseAddress = "http://localhost:8733/Design Time Addresses/PhoneBookService/PhoneService/" />
      </baseAddresses>
    </host>
    <!-- Service Endpoints -->
    <!-- Unless fully qualified, address is relative to base address supplied above -->
    <endpoint address="" binding="basicHttpBinding" contract="PhoneBookService.IPhoneService">
```

9. נכתוב Host שיריץ את ה- Service

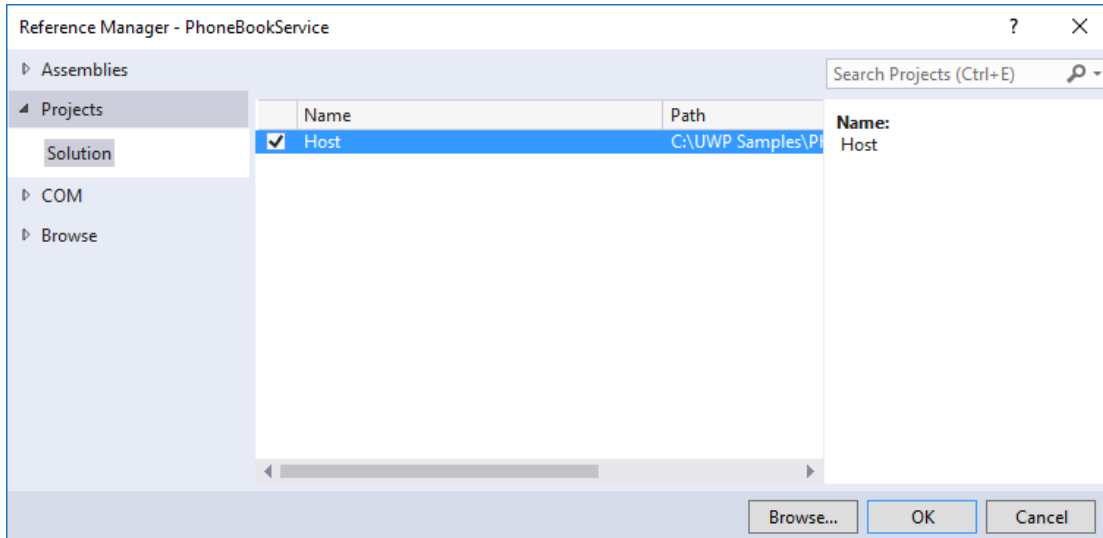


10. References

נוסף ל-Host ייחוס ל- :System.ServiceModel



וייחוס לפרויקט ה-Service שכתבנו:



11. נכתוב ב-Host את הקוד הבא:

```
class Program
{
    0 references
    static void Main(string[] args)
    {
        ServiceHost service = new
            ServiceHost(typeof(PhoneBookService.PhoneService));
        service.Open();
        Console.ReadLine();
    }
}
```

12. App.Config:

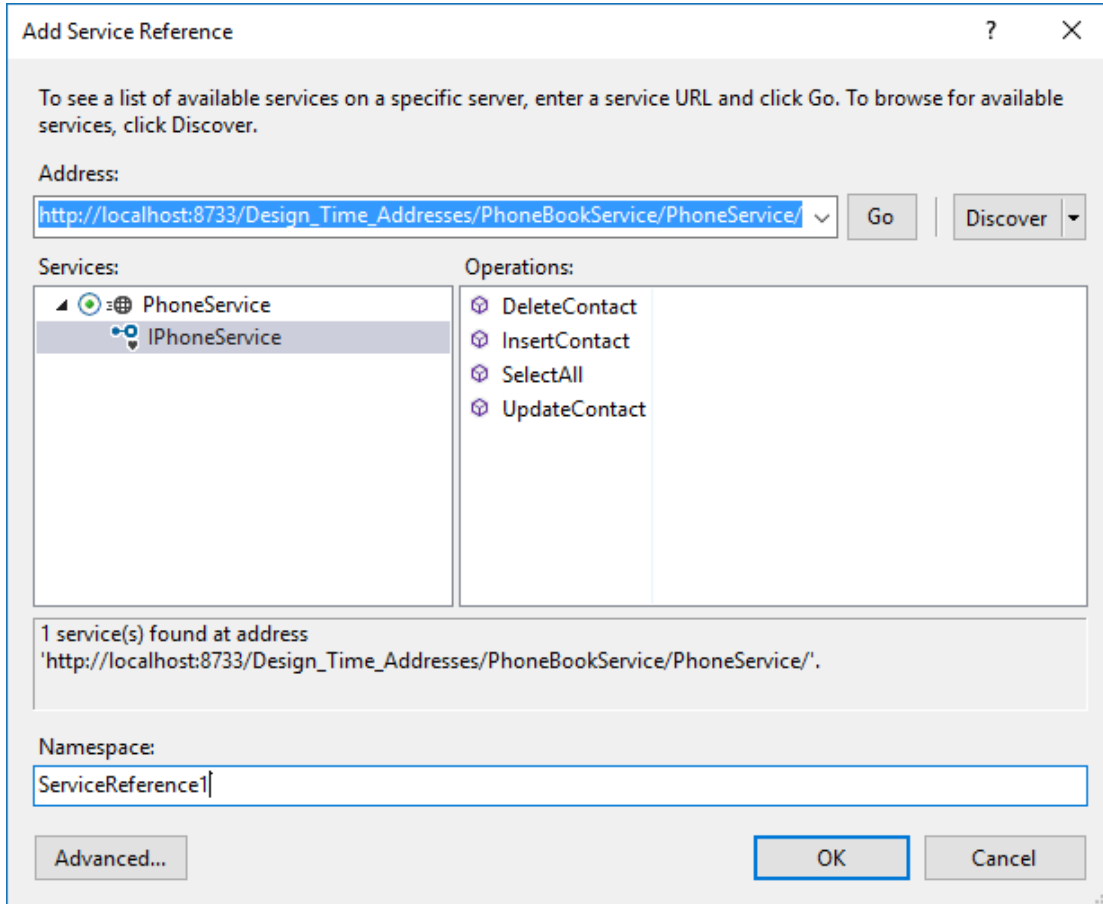
נעתיק את קובץ ה-App.Config מפרויקט השרות לפרויקט המארז.
נמחק את הקובץ App.Config מפרויקט השרות.

13. כתיבת אפליקציית הלקוח

נפתח פרויקט UWP חדש

14. נוסיף Service Reference לשרות

ה-Host חייב לרוץ בזמן הוספת Service Reference:



15. הקצאת פרוקסי וקריאת מידע:

```
private async void Page_Loading(FrameworkElement sender, object args)
{
    PhoneServiceClient proxy = new PhoneServiceClient();
    var result = await proxy.SelectAllAsync();
    lstContact.ItemsSource = result;
}
```

16. הצגת המידע:

```
<Grid Background="{ThemeResource ApplicationPageBackgroundThemeBrush}">
    <ListView Name="lstContact">
        <ListView.ItemTemplate>
            <DataTemplate>
                <Border BorderBrush="Red" BorderThickness="1" >
                    <StackPanel Orientation="Horizontal">
                        <TextBlock Text="{Binding LastName}" Margin="0,0,10,0"></TextBlock>
                        <TextBlock Text="{Binding FirstName}" Margin="0,0,10,0"></TextBlock>
                        <TextBlock Text="{Binding PhoneNum}" Margin="0,0,10,0"></TextBlock>
                    </StackPanel>
                </Border>
            </DataTemplate>
        </ListView.ItemTemplate>
    </ListView>
</Grid>
```