



ספר פרויקט: "Master Chef"

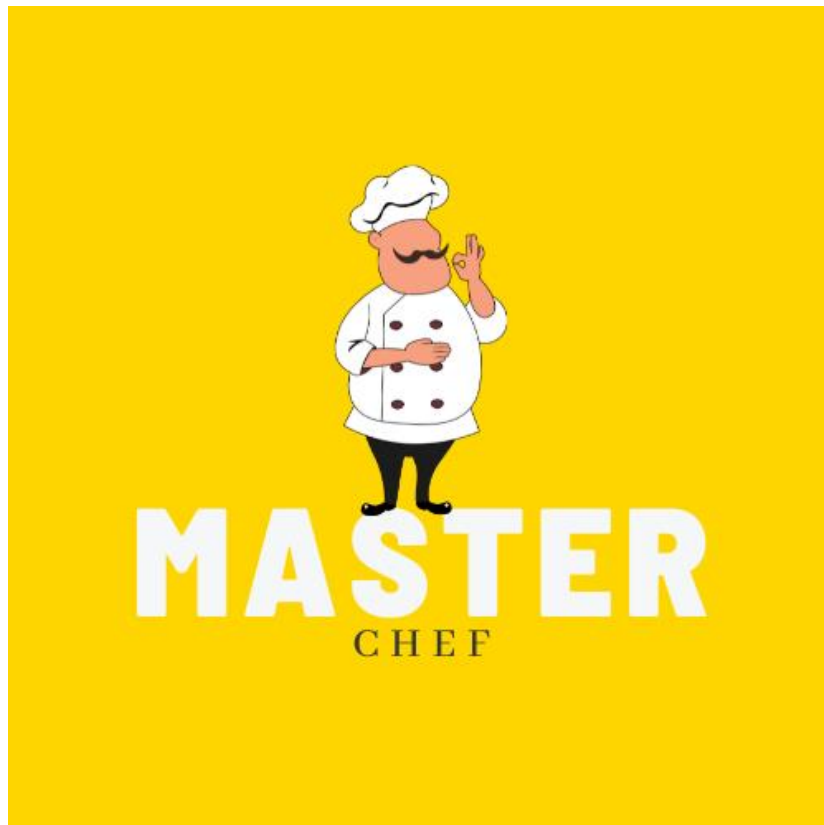
מגיש: נתנאל צ'רטקוב

כיתה: י"ג

בית ספר: מקיף ז' הקיריה

ת.ז.: 214330177

שם המורה: אולגה גנדלמן



על הפרויקט: מהלך תקופת הקורונה כולנו היינו סגורים בבתים מחפשים להעסיק את עצמנו במשהו. המשהו הזה בשבילי היה בישול, יחד עם כל המשפחה כל היום בישראל ואפילו. אז החלטתי לקחת את זה צעד אחד קדימה ופיתחתי משחק בישול אך בשונה מהשאר המתכונים בו אמיתיים לחלוטין ומאפשרים למשתמש אחר כך להכין את המתכון בבית.

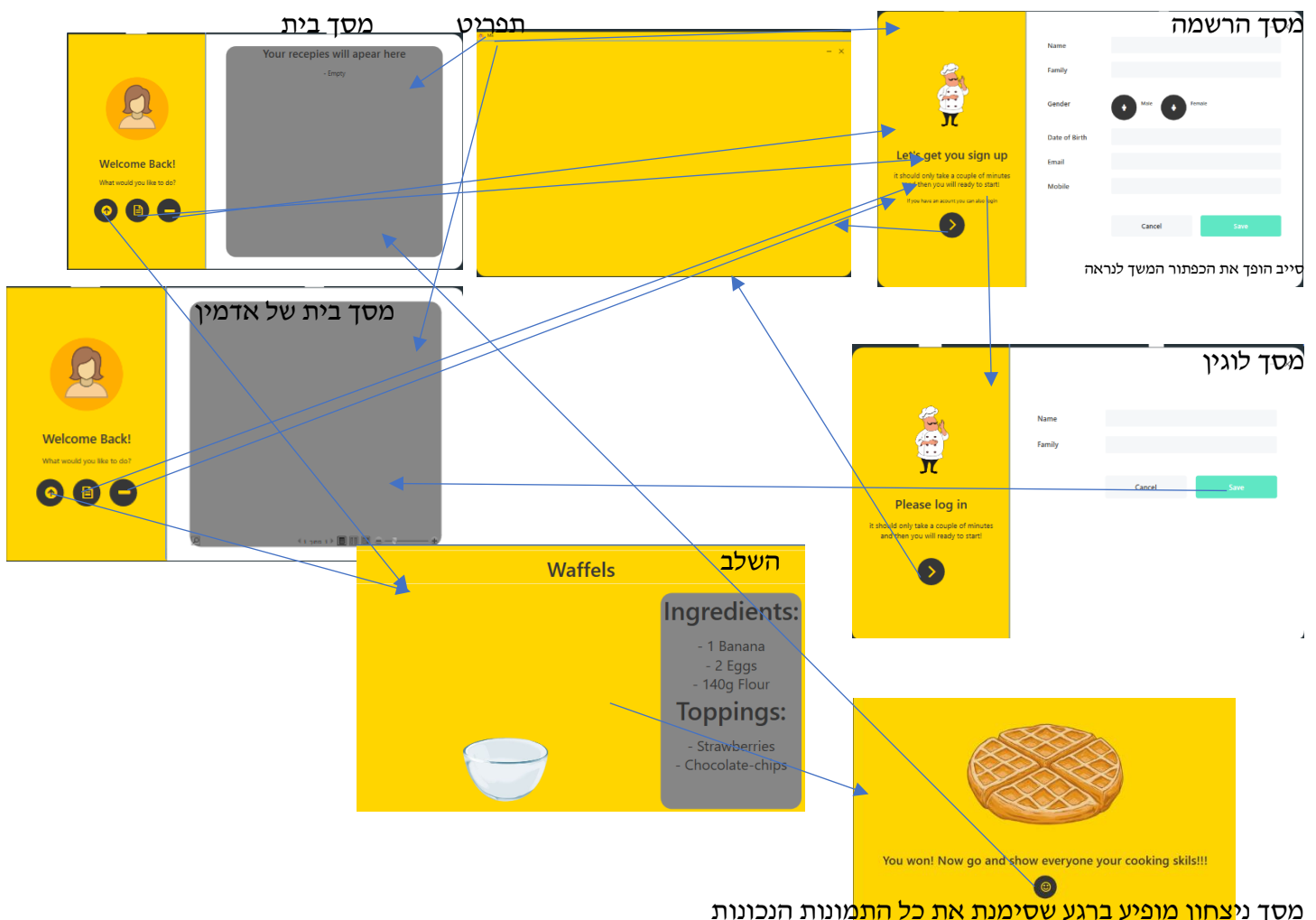
תיאור האפליקציה: למשתמש מוצג מתכון ומגוון מוצרים שונים. על המשתמש ללחוץ ולבחור את המוצרים המתאימים בלבד.

מדריך למשתמש: המשתמש נכנס לאפליקציה דרך מילוי הפרטים המתבקשים. לאחר מכן הוא יגיע אל מסך הבית ויוכל לבחור אם לצאת או להמשיך הלאה אל השלב עצמו. במסך הבית יוצגו כל המתכונים של המשתמש. בשלב המשחק המשתמש יצטרך ללחוץ על התמונות של המצרכים הרשומים לו במתכון עד להופעת מסך הניצחון.

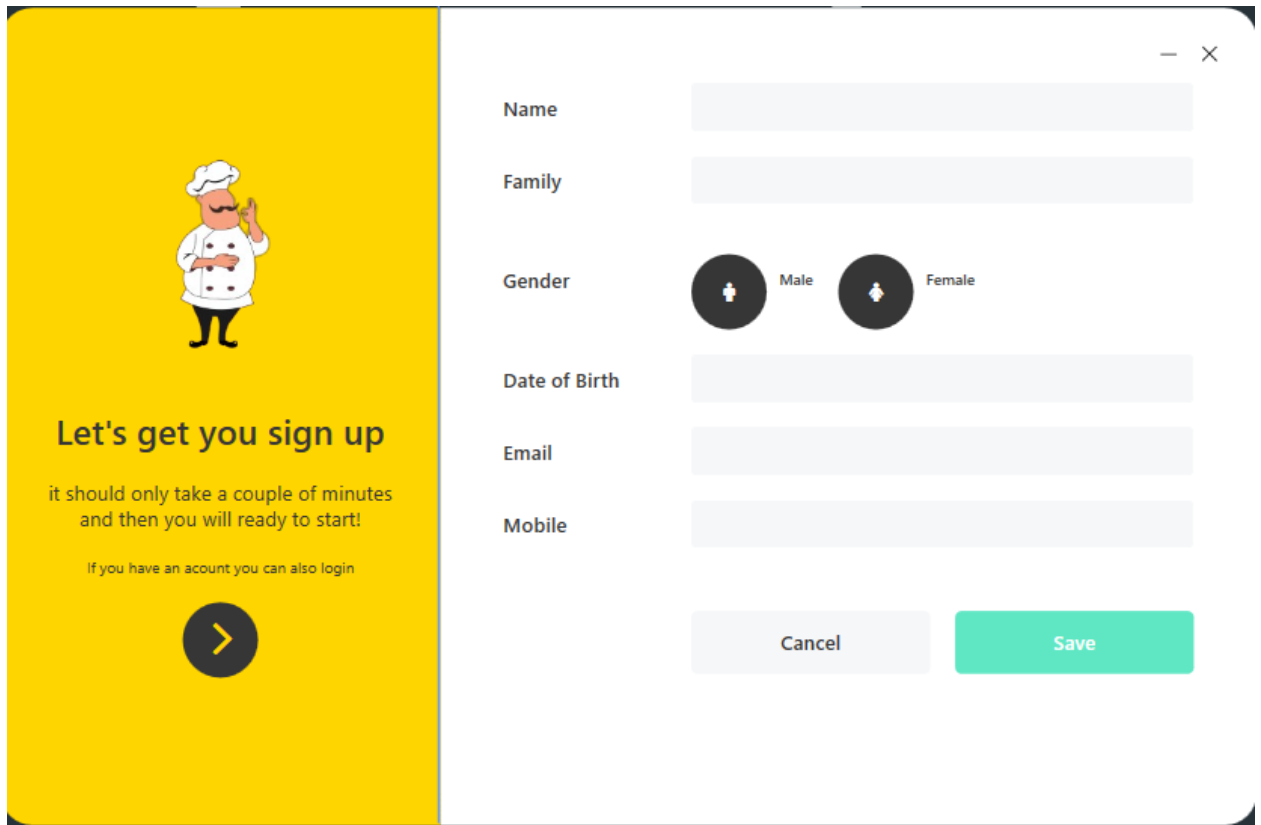
טבלה של בסיס נתונים:

שם שדה	סוג נתונים	מספר משתמש (משתמש מספר 1 - הינו המשתמש הראשון והמנהל)
User ID	מספור אוטומטי	
Username	טקסט קצר	שם המשתמש
Family	טקסט קצר	שם משפחה
Gender	מספר	מגדר: 0 - ז', 1 - נ'
Date of Birth	תאריך/שעה	יום הולדת
Email	טקסט קצר	דואר אלקטרוני
Mobile	מספר גדול	מספר טלפון נייד

מפת הפרויקט:



מסך ניצחון מופיע ברגע שסימנת את כל התמונות הנכונות



כפתור "המשך", כפתורי שמירה וביטול ותמונות לחיצות של סיגירות מסך, צמצום מסך ובחירת מגדר.

xaml

```
<Button Style="{StaticResource buttonBlack}" Click="ContinueToMainPage" x:Name="NextBtn" Visibility="Hidden">
  <fa:ImageAwesome Icon="AngleRight" Width="25" Height="25" Foreground="#FFD500" Margin="3 0 0 0"/>
</Button>

<Image Source="/Images/close.png" Width="25" Margin="0 4 5 0" Style="{StaticResource iconApp}" MouseDown="CloseSignUpWindow"/>
<Image Source="/Images/minimize.png" Width="20" Margin="0 7 40 0" Style="{StaticResource iconApp}" MouseDown="MinimizeSignUpWindow"/>

<Button Style="{StaticResource buttonMain}" Margin="0 0 10 0" Content="Cancel" Click="ClearRegistration"/>
<Button Grid.Column="1" Style="{StaticResource buttonMainGreen}" Margin="10 0 0 0" Content="Save" Click="Register"/>
```

```

private void Register(object sender, RoutedEventArgs e)
{
    win.Name.Text = Name.Text.ToString();
    win.Family.Text = Family.Text;
    if (Name.Text.ToString() != null)
    {
        Add(Name.Text.ToString(), "Username");
    }
    if (Family.Text.ToString() != null)
        Add(Family.Text.ToString(), "Family");
    if (Date.Text.ToString() != null)
        Add(Date.Text.ToString(), "Date of Birth");
    if (Email.Text.ToString() != null)
        Add(Email.Text.ToString(), "Email");
    if (Mobile.Text.ToString() != null)
        Add(Mobile.Text.ToString(), "Mobile");
    NextBtn.Visibility = Visibility.Visible;
}

5 references
public void Add(string str, string place)
{
    //הוספת פרטים
    string sqlCommand = $"insert into Users ({place}) values ('{str}')";
    int n = DAL.ExecuteNonQuery(sqlCommand);
}

1 reference
private void ContinueToMainPage(object sender, RoutedEventArgs e)
{
    if (Name.Text.ToString() != null)
    {
        HomePage p = new HomePage();
        p.Show();
        this.Close();
    }
}

1 reference
private void CloseSignUpWindow(object sender, MouseButtonEventArgs e)
{
    App.Current.Shutdown();
}

1 reference
private void MinimizeSignUpWindow(object sender, MouseButtonEventArgs e)
{
    this.WindowState = WindowState.Minimized;
}

0 references
private void Male(object sender, MouseButtonEventArgs e)
{
    string sqlCommand = $"insert into Users (Gender) values (0)";
    int n = DAL.ExecuteNonQuery(sqlCommand);
}

0 references
private void Female(object sender, MouseButtonEventArgs e)
{
    string sqlCommand = $"insert into Users (Gender) values (1)";
    int n = DAL.ExecuteNonQuery(sqlCommand);
}

1 reference
private void ClearRegistration(object sender, RoutedEventArgs e)
{
    this.Close();
}

1 reference
private void Login(object sender, MouseButtonEventArgs e)
{
    LogInWin win = new LogInWin();
    win.Show();
    this.Close();
}

```

בנייה סטטית של מסך : כל חלק במסך מוגדר מראש בגרפיקה ולא משתנה



You won! Now go and show everyone your cooking skills!!!

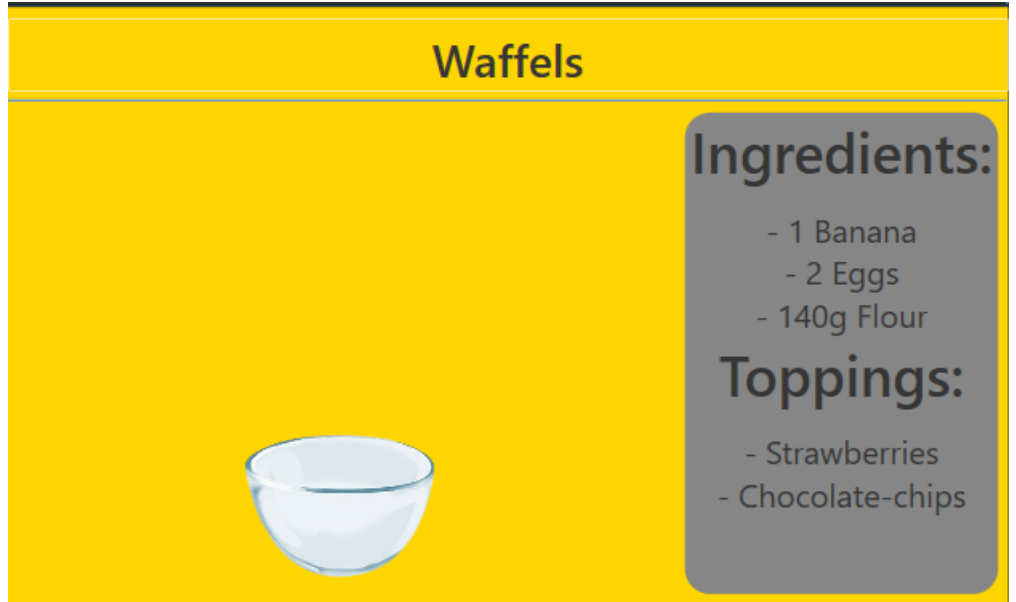


```

<Grid Background="#FFD500">
  <StackPanel>
    <Image Source="/images/waffels.png" Height="300" Width="500" Margin="50"/>
    <TextBlock TextAlignment="Center" FontWeight="SemiBold" FontSize="32" Text="You won! Now go and show everyone your cooking skills!!!"
      Foreground="#363636"/>
    <Button Style="{StaticResource buttonBlack}" Click="ContinueToMainPage" x:Name="NextBtn" Visibility="Visible">
      <fa:ImageAwesome Icon="SmileOutline" Width="25" Height="25" Foreground="#FFD500" Margin="3 0 0 0" HorizontalAlignment="Center"/>
    </Button>
  </StackPanel>
</Grid>

```

בנייה דינמית של המסך : כל המצרכים יוצרו רק בהפעלת המשחק דרך הסי שארפ



```

<UniformGrid x:Name="ingredients" Width="667" Height="300" Rows="2" FlowDirection="LeftToRight" Columns="6"/>
</UniformGrid>

```

```

private void Build()
{
  imgObj.Clear();
  ingredients.Children.Clear();

  List<string> imageNames = new List<string>(ALLIMAGESNAMES);
  int img = 0;
  while (imageNames.Count > 0)
  {
    string imgFullPath = Path() + "/" + imageNames[img];
    var bitmap = new BitmapImage(new Uri(imgFullPath));
    Image image = new Image();
    image.Source = bitmap;

    image.Height = 100;
    image.Tag = img;
    image.MouseDown += Check;

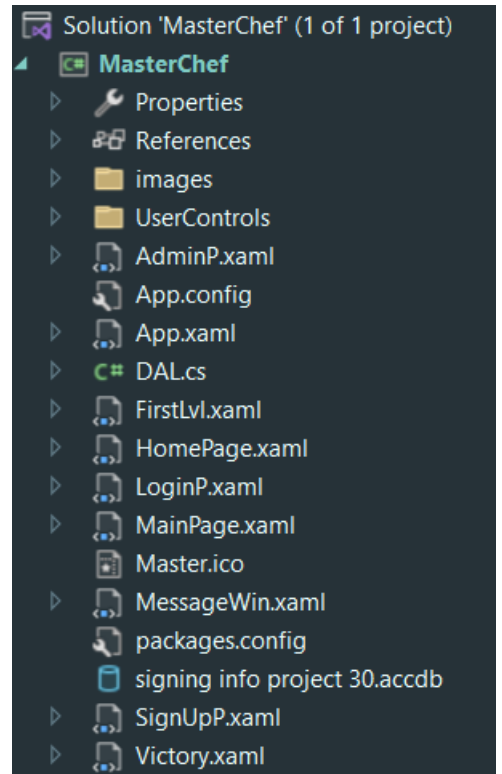
    ingredients.Children.Add(image);
    imageNames.RemoveAt(img);
    img++;
  }
}

```

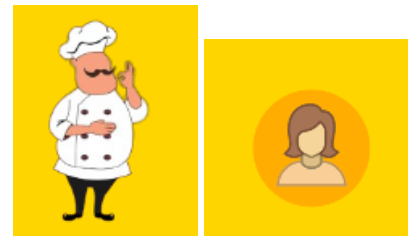
בנייה דינמית - טבלת בסיס נתונים :

Users						
User's ID	First Name	Last Name	Gender	Date of Birth	Email	Mobil e
1	Netanel	Chertkov	0	08/09/2007	netanel.chertko@gmail.com	532830809
5	System.Windows.Controls.Grid		0			0 0
6		System.Windows.Controls.Grid	0			0 0
7	System.Windows.Controls.Grid		0			0 0
8		System.Windows	0			0 0

עבודה עם עמודים ומסכים : השתמשתי במגוון רחב של פייג'ים ושל ווינדוס



תמונות :



```
<Image Source="images/chef.png" Margin="0 0 0 40" Width="160" Height="160"/>  
<Image Source="images/profileIcon.png" Margin="0 0 0 40" Width="160" Height="160"/>
```

שימוש בגריד: הגריד שימש אותי רבות לחלוקת המסך

```
<Grid>
  <Image Source="/Images/close.png" Width="25" Margin="0 4 5 0" Style="{StaticResource iconApp}" MouseDown="CloseSignUpWindow"/>
  <Image Source="/Images/minimize.png" Width="20" Margin="0 7 40 0" Style="{StaticResource iconApp}" MouseDown="MinimizeSignUpWindow"/>

  <Grid HorizontalAlignment="Center" VerticalAlignment="Center" Height="550" Width="550">
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="150"/>
      <ColumnDefinition Width="*"/>
    </Grid.ColumnDefinitions>

    <Grid.RowDefinitions>
      <RowDefinition Height="auto"/>
      <RowDefinition Height="auto"/>
      <RowDefinition Height="auto"/>
      <RowDefinition Height="auto"/>
      <RowDefinition Height="auto"/>
      <RowDefinition Height="auto"/>
      <RowDefinition Height="auto"/>
    </Grid.RowDefinitions>

    <TextBlock Grid.Row="0" Text="Name" Style="{StaticResource text}"/>
    <TextBlock Grid.Row="1" Text="Family" Style="{StaticResource text}"/>
    <TextBlock Grid.Row="2" Text="Gender" Style="{StaticResource text}"/>
    <TextBlock Grid.Row="3" Text="Date of Birth" Style="{StaticResource text}"/>
    <TextBlock Grid.Row="4" Text="Email" Style="{StaticResource text}"/>
    <TextBlock Grid.Row="5" Text="Mobile" Style="{StaticResource text}"/>

    <TextBox Grid.Column="1" Grid.Row="0" x:Name="Name"/>
    <TextBox Grid.Column="1" Grid.Row="1" x:Name="Family"/>
    <TextBox Grid.Column="1" Grid.Row="3" x:Name="Date"/>
    <TextBox Grid.Column="1" Grid.Row="4" x:Name="Email"/>
    <TextBox Grid.Column="1" Grid.Row="5" x:Name="Mobile"/>

    <StackPanel Orientation="Horizontal" Grid.Column="1" Grid.Row="2" Margin="0 10">
      <uc:MyOption Text="Male" Icon="Male" MouseDoubleClick="Male"/>
      <uc:MyOption Text="Female" Icon="Female" MouseDoubleClick="Female"/>
    </StackPanel>

    <Grid Grid.Column="1" Grid.Row="7" Margin="0 40 0 0">
      <Grid.ColumnDefinitions>
        <ColumnDefinition Width="*"/>
        <ColumnDefinition Width="*"/>
      </Grid.ColumnDefinitions>

      <Button Style="{StaticResource buttonMain}" Margin="0 0 10 0" Content="Cancel" Click="ClearRegistration"/>
      <Button Grid.Column="1" Style="{StaticResource buttonMainGreen}" Margin="10 0 0 0" Content="Save" Click="Register"/>
    </Grid>
  </Grid>
</Grid>
```

תפריט:

```
<DockPanel>
  <Menu DockPanel.Dock="Top" Background="#FFD500">
    <MenuItem Header="ME">
      <MenuItem.Icon>
        <Image Source="/images/profileIcon.png"/>
      </MenuItem.Icon>
      <MenuItem Header="Profile" Click="Profile" Visibility="Hidden" x:Name="pro"/>
      <MenuItem Header="Sign out" Click="LogOut" Visibility="Hidden" x:Name="so"/>
      <MenuItem Header="Sign in" Click="LogIN" Visibility="Visible" x:Name="si"/>
      <MenuItem Header="Sign up" Click="LogOut" Visibility="Visible" x:Name="su"/>
    </MenuItem>
  </Menu>
</DockPanel>
```

עבודה עם מוסד נתונים:

```
public void Add(string str, string place)
{
    //הוספת שורה
    string sqlCommand = $"insert into Users ({place}) values ('{str}')";
    int n = DAL.ExecuteNonQuery(sqlCommand);
}
```

```
private void Male(object sender, MouseButtonEventArgs e)
{
    string sqlCommand = $"insert into Users (Gender) values (0)";
    int n = DAL.ExecuteNonQuery(sqlCommand);
}
```

0 references

```
private void Female(object sender, MouseButtonEventArgs e)
{
    string sqlCommand = $"insert into Users (Gender) values (1)";
}
```

```

private void Login(object sender, RoutedEventArgs e)
{
    string username = Name.Content.ToString();
    string password = Family.Content.ToString();

    if(username == "Netanel" && password == "Chertkov")
    {
        isAdmin = true;
        HomePage homePage = new HomePage();
        homePage.Show();
        this.Close();
    }

    string connectionString = DAL.ConnectionString;

    OleDbConnection conn = new OleDbConnection(connectionString);

    string sql = "Select * from Users where [Username]='" + username + "' and [Family]='" + password + "'";

    OleDbCommand cmd = new OleDbCommand(sql, conn);

    conn.Open();

    OleDbDataReader reader;

    reader = cmd.ExecuteReader();

    if (reader.Read())
    {
        //קיים ID
        MessageBox.Show("Welcome Back " + username + "!");
        NextBtn.Visibility = Visibility.Visible;
    }

    else
    {
        //לא קיים ID
        MessageBox.Show("Sorry, couldn't found you. Please try again");
    }
}

```

```

public void FillTable1()
{
    table1.RowGroups.Clear(); //מנקים את כל התוכן הישן
    TableRow currentRow;
    int i = 2; //מסמנים את שורה ראשונה שממלאים באופן דינמי כי שורה ראשונה שמורה לכתובת הטבלה והשנייה לקטגוריות
    AddHeadLines();
    string connectionString = DAL.ConnectionString;

    OleDbConnection conn = new OleDbConnection(connectionString);
    string sql = "";
    sql = BuildSqlCommand();

    OleDbCommand cmd = new OleDbCommand(sql, conn);
    conn.Open();
    OleDbDataReader reader;
    reader = cmd.ExecuteReader();

    while (reader.Read())
    {
        currentRow = new TableRow();
        table1.RowGroups[0].Rows.Add(currentRow); //כל פעם נוסף שורה לטבלה

        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["User ID"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Username"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Family"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Gender"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Date of Birth"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Email"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Mobile"].ToString()))));
        currentRow.Cells.Add(new TableCell(new Paragraph(new Run(reader["Membership"].ToString()))));
    }
    reader.Close();
    conn.Close();
}

1 reference
private string BuildSqlCommand()
{
    string sql = "Select * from Users";
    return sql;
}

```


לימוד עצמי: לחיצות מקשי העכבר, מזעור וסגירת חלון ושוליים מעוגלות.

למדתי למה להשתמש בהם.

לדוגמה: ניתן לעגל שוליים כדי ליצור כפתורים מעוגלים וחלונות יותר מעוצבים ונוחים לעין במקום החלונות החדים והמשעממים.

```
<Button Style="{StaticResource buttonBlack}" Click="ContinueToMainPage" x:Name="NextBtn" Visibility="Visible">
  <fa:ImageAwesome Icon="AngleRight" Width="25" Height="25" Foreground="■" #FFD500" Margin="3 0 0 0"/>
</Button>
```

```
<Style x:Key="buttonBlack" TargetType="Button">
  <Setter Property="Background" Value="■" #363636"/>
  <Setter Property="BorderThickness" Value="0"/>
  <Setter Property="Width" Value="60"/>
  <Setter Property="Height" Value="60"/>
  <Setter Property="Margin" Value="0 20 0 0"/>
  <Setter Property="Template">
    <Setter.Value>
      <ControlTemplate TargetType="Button">
        <Border Background="{TemplateBinding Background}" CornerRadius="50" Padding="5">
          <ContentPresenter HorizontalAlignment="Center" VerticalAlignment="Center"/>
        </Border>
      </ControlTemplate>
    </Setter.Value>
  </Setter>
  <Style.Triggers>
    <Trigger Property="IsMouseOver" Value="True">
      <Setter Property="Background" Value="■" Black"/>
    </Trigger>
  </Style.Triggers>
</Style>
```

```
<Border Background="■" #FFD500" Padding="30" CornerRadius="25 0 0 25">
```

בנוסף למדתי כי אין חובת שימוש בכפתורי הברירת מחדל בצד הימני של החלון לכן יצרתי באופן עצמאי עם תמונות של מינוס עבור מזעור החלון ופלוס עבור סגירתו גרסה שלי ללחצנים הללו.

```
private void CloseSignUpWindow(object sender, MouseButtonEventArgs e)
{
  App.Current.Shutdown();
}

1 reference
private void MinimizeSignUpWindow(object sender, MouseButtonEventArgs e)
{
  this.WindowState = WindowState.Minimized;
}
```

ולבסוף למדתי שלא רק בכפתור קיימת פעולת לחיצה אלא על כמעט כל אלמנט בקסמל ניתן לחוץ או להעביר את העכבר והשתמשתי גם בזה ליצירת אפקטים של לחיצה, העברה וגר

```
private void Border_MouseDown(object sender, MouseButtonEventArgs e)
{
  if (e.ChangedButton == MouseButton.Left) this.DragMove();
}
```

```
<Border Grid.Column="1" Padding="20" MouseDown="Border_MouseDown" Background="■" White" CornerRadius="0 25 25 0">
```

```

<Style.Triggers>
  <Trigger Property="IsMouseOver" Value="True">
    <Setter Property="RenderTransform">
      <Setter.Value>
        <ScaleTransform ScaleX="1.2" ScaleY="1.2"/>
      </Setter.Value>
    </Setter>
    <Setter Property="Cursor" Value="Hand"/>
  </Trigger>
</Style.Triggers>

```

LoginP.xaml.cs

```

public partial class LoginP : Page
{
    public static bool isAdmin = false;
    public LoginP()
    {
        InitializeComponent();
    }
    private void Login(object sender, RoutedEventArgs e)
    {
        string username = Name.Text.ToString();

        string password = Family.Text.ToString();

        if (username == "Netanel" && password == "Chertkov")
        {
            isAdmin = true;
            AdminP sign = new AdminP();
            HomePage.myFrame.NavigationService.Navigate(sign);
            HomePage.signOut.Visibility = Visibility.Visible;
            HomePage.profile.Visibility = Visibility.Visible;
            HomePage.signIn.Visibility = Visibility.Hidden;
            HomePage.signUp.Visibility = Visibility.Hidden;
        }

        string connectionString = DAL.ConnectionString;

        OleDbConnection conn = new OleDbConnection(connectionString);

        string sql = "Select * from Users where [Username]='" + username + "'
and [Family]='" + password + "'";

        OleDbCommand cmd = new OleDbCommand(sql, conn);

        conn.Open();

        OleDbDataReader reader;

        reader = cmd.ExecuteReader();

        if (reader.Read())
        {
            //ויברא
            MessageBox.Show("Welcome Back " + username + "!");
            NextBtn.Visibility = Visibility.Visible;
            HomePage.signOut.Visibility = Visibility.Visible;
            HomePage.profile.Visibility = Visibility.Visible;
        }
    }
}

```

```

        HomePage.signIn.Visibility = Visibility.Hidden;
        HomePage.signUp.Visibility = Visibility.Hidden;
    }

    else
    {

        //MessageBox.Show("Sorry, couldn't found you. Please try again");
    }
}

private void ClearRegistration(object sender, RoutedEventArgs e)
{
    SignUpP sign = new SignUpP();
    HomePage.myFrame.NavigationService.Navigate(sign);
}

private void ContinueToMainPage(object sender, RoutedEventArgs e)
{
    MainPage sign = new MainPage();
    HomePage.myFrame.NavigationService.Navigate(sign);
    HomePage.signOut.Visibility = Visibility.Visible;
    HomePage.profile.Visibility = Visibility.Visible;
    HomePage.signIn.Visibility = Visibility.Hidden;
    HomePage.signUp.Visibility = Visibility.Hidden;
}
}

```

FirstLvl.xaml.cs

```

public partial class FirstLvl : Page
{
    bool flag1 = false, flag2 = false, flag3 = false, flag4 = false, flag5 =
false; // checks if the player won
    public Random rnd = new Random();
    public List<Image> imgObj = new List<Image>();
    readonly List<string> ALLIMAGESNAMES = new List<string>()
        { "banana.png", "egg.png", "flour.png", "Strawberry.png", "chocolate-
chip.png", "cucamber.png", "honey.png", "milk.png", "potato.png", "rice.png",
"sugar.png", "salt.png"};
    public FirstLvl()
    {
        InitializeComponent();
        Build();
    }
    private void Build()
    {
        imgObj.Clear();
        ingredients.Children.Clear();

        List<string> imageNames = new List<string>(ALLIMAGESNAMES);
        int img = 0;
        while (imageNames.Count() > img)
        {
            string imgFullPath = Path() + "/" + imageNames[img];
            var bitmap = new BitmapImage(new Uri(imgFullPath));
            Image image = new Image();
            image.Source = bitmap;

            image.Height = 100;
            image.Tag = img;
            image.MouseDown += Check;

            ingredients.Children.Add(image);
        }
    }
}

```

```

        //imageNames.RemoveAt(img);
        img++;
    }
}

public static string Path()
{
    string s = Environment.CurrentDirectory; // מקבלים כתובת של קובץ הרצה
    string[] ss = s.Split('\\'); // מפרקים את המחרוזת לחלקים כאשר בין החלקים
    int x = ss.Length - 2; // אנו בוחרים את החלקים שמעריך
    ss[x] = "images"; // אז במקום השלישי מהסוף מכניסים את שם התיקיה עם
    Array.Resize(ref ss, x + 1); // פותחים את המערך
    string s1 = String.Join("\\", ss); // מחזירים את המערך חזרה למחרוזת עם
    return s1;
}

private void Check(object sender, MouseButtonEventArgs e)
{
    Image image = (Image)sender;
    if (e.ChangedButton == MouseButton.Left)
    {
        if(image.Tag.Equals(0))
        {
            MessageBox.Show("Great job!");
            image.Visibility = Visibility.Collapsed;
            flag1 = true;
        }
        else if(image.Tag.Equals(1))
        {
            MessageBox.Show("Well done!");
            image.Visibility = Visibility.Collapsed;
            flag2 = true;
        }
        else if (image.Tag.Equals(2))
        {
            MessageBox.Show("Amazing!");
            image.Visibility = Visibility.Collapsed;
            flag3 = true;
        }
        else if (image.Tag.Equals(3))
        {
            MessageBox.Show("Amazing!");
            image.Visibility = Visibility.Collapsed;
            flag4 = true;
        }
        else if (image.Tag.Equals(4))
        {
            MessageBox.Show("Great job!");
            image.Visibility = Visibility.Collapsed;
            flag5 = true;
        }
        else
        {
            MessageBox.Show("Try again...");
        }
    }
    EndTheLvl(flag5, flag1, flag2, flag4, flag3);
}

public void EndTheLvl(bool f1, bool f2, bool f3, bool f4, bool f5)
{
    if (f1 && f2 && f3 && f4 && f5)

```

```

    {
        HomePage.myFrame.NavigationService.Navigate(new Victory());
    }
}
}

```

SignUpP.xaml.cs

```

public partial class SignUpP : Page
{
    public static SignUpP win = new SignUpP();
    public SignUpP()
    {
        InitializeComponent();
    }

    private void Register(object sender, RoutedEventArgs e)
    {
        win.Name.Text = Name.Text.ToString();
        win.Family.Text = Family.Text;
        if (Name.Text.ToString() != null)
        {
            Add(Name.Text.ToString(), "Username");
        }
        if (Family.Text.ToString() != null)
            Add(Family.Text.ToString(), "Family");
        if (Date.Text.ToString() != null)
            Add(Date.Text.ToString(), "Date of Birth");
        if (Email.Text.ToString() != null)
            Add(Email.Text.ToString(), "Email");
        if (Mobile.Text.ToString() != null)
            Add(Mobile.Text.ToString(), "Mobile");
        NextBtn.Visibility = Visibility.Visible;
        HomePage.signOut.Visibility = Visibility.Visible;
        HomePage.profile.Visibility = Visibility.Visible;
        HomePage.signIn.Visibility = Visibility.Hidden;
        HomePage.signUp.Visibility = Visibility.Hidden;
    }
    public void Add(string str, string place)
    {
        //הוספת שורה
        string sqlCommand = $"insert into Users ({place}) values ('{str}')";
        int n = DAL.ExecuteNonQuery(sqlCommand);
    }

    private void ContinueToMainPage(object sender, RoutedEventArgs e)
    {
        if (Name.Text.ToString() != null)
        {
            HomePage.myFrame.NavigationService.Navigate(new MainPage());
            HomePage.signOut.Visibility = Visibility.Visible;
            HomePage.profile.Visibility = Visibility.Visible;
            HomePage.signIn.Visibility = Visibility.Hidden;
            HomePage.signUp.Visibility = Visibility.Hidden;
        }
    }

    private void Male(object sender, MouseButtonEventArgs e)
    {
        string sqlCommand = $"insert into Users (Gender) values (0)";
        int n = DAL.ExecuteNonQuery(sqlCommand);
    }
}

```

```

private void Female(object sender, MouseButtonEventArgs e)
{
    string sqlCommand = $"insert into Users (Gender) values (1)";
    int n = DAL.ExecuteNonQuery(sqlCommand);
}

private void ClearRegistration(object sender, RoutedEventArgs e)
{
    Name.Clear();
    Family.Clear();
    Date.Clear();
    Mobile.Clear();
    Email.Clear();
}

private void Login(object sender, MouseButtonEventArgs e)
{
    LoginP win = new LoginP();
    HomePage.myFrame.NavigationService.Navigate(win);
}
}

```

MyOption.xaml.cs

```

public partial class MyOption : UserControl
{
    public MyOption()
    {
        InitializeComponent();
    }

    public string Text
    {
        get { return (string)GetValue(TextProperty); } // Getter for the Text
        set { SetValue(TextProperty, value); } // Setter for the Text property
    }
    // Defines a DependencyProperty called TextProperty of type string for the
    // MyOption class
    public static readonly DependencyProperty TextProperty =
    DependencyProperty.Register("Text", typeof(string), typeof(MyOption));

    public FontAwesome.WPF.FontAwesomeIcon Icon
    {
        get { return (FontAwesome.WPF.FontAwesomeIcon)GetValue(IconProperty); }
        set { SetValue(IconProperty, value); }
    }
    // Defines a DependencyProperty called IconProperty of type FontAwesomeIcon
    // for the MyOption class
    public static readonly DependencyProperty IconProperty =
    DependencyProperty.Register("Icon", typeof(FontAwesome.WPF.FontAwesomeIcon),
    typeof(MyOption));
}

```

MainPage.xaml.cs

```

public partial class MainPage : Page
{
    public MainPage()
    {
        InitializeComponent();
        if (Victory.WonLvl)
        {
            AddFirstRecepie();
        }
    }
}

```

```

    }
    private void AddFirstRecepie()
    {
        TitleR.Text = "Waffels";
        contentR.Text = "- 1 Banana\n- 2 Eggs\n- 140g Flour\n- Strawberries\n-
Chochlate Chips";
    }
    private void NextLvl(object sender, RoutedEventArgs e)
    {
        HomePage.myFrame.NavigationService.Navigate(new FirstLvl());
    }

    private void Update(object sender, RoutedEventArgs e)
    {
        SignUpP mainWindow = new SignUpP();
        HomePage.myFrame.NavigationService.Navigate(mainWindow);
    }
}

```

MessageWin.xaml.cs

```

public partial class MessageWin : Window
{
    public MessageWin()
    {
        InitializeComponent();
    }

    private void OK(object sender, RoutedEventArgs e)
    {
        int ID;
        try
        {
            ID = int.Parse(Message.Text.ToString());
            string sqlCommand = $"delete from Users where User ID= {ID}";
            int n = DAL.ExecuteNonQuery(sqlCommand);
        }
        catch(Exception ex) { }
    }
}

```

Victory.xaml.cs

```

public partial class Victory : Page
{
    public static bool WonLvl = false;
    public Victory()
    {
        InitializeComponent();
    }

    private void ContinueToMainPage(object sender, RoutedEventArgs e)
    {
        WonLvl = true;
        if (LogInWin.isAdmin)
            HomePage.myFrame.NavigationService.Navigate(new AdminP());
        else
        {
            MainPage mainPage = new MainPage();
            HomePage.myFrame.NavigationService.Navigate(mainPage);
        }
    }
}

```

AdminP.xaml.cs

```

public partial class AdminP : Page

```

```

{
    public AdminP()
    {
        InitializeComponent();
        FillTable1();
    }
    public void FillTable1()
    {
        table1.RowGroups.Clear(); // מנקים את כל בטוחן פישן
        TableRow currentRow;
        int i = 2; // שורה ראשונה ראשונה פנימי פנימי שורה ראשונה ראשונה
        שמורה לכותרת בטבלה פשהשנייה לקטגוריות
        AddHeadLines();
        string connectionString = DAL.ConnectionString;

        OleDbConnection conn = new OleDbConnection(connectionString);
        string sql = "";
        sql = BuildSqlCommand();

        OleDbCommand cmd = new OleDbCommand(sql, conn);
        conn.Open();
        OleDbDataReader reader;
        reader = cmd.ExecuteReader();

        while (reader.Read())
        {
            currentRow = new TableRow();
            table1.RowGroups[0].Rows.Add(currentRow); // כל פעם נוסף שורה בטבלה

            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["User ID"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Username"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Family"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Gender"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Date of Birth"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Email"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Mobile"].ToString()))));
            currentRow.Cells.Add(new TableCell(new Paragraph(new
Run(reader["Membership"].ToString()))));
        }
        reader.Close();
        conn.Close();
    }

    private string BuildSqlCommand()
    {
        string sql = "Select * from Users";
        return sql;
    }

    private void AddHeadLines()
    {
        table1.RowGroups.Add(new TableRowGroup());
    }
}

```



```

// Add the first (title) row.
table1.RowGroups[0].Rows.Add(new TableRow());
// Alias the current working row for easy reference.
TableRow currentRow = table1.RowGroups[0].Rows[0];
// Global formatting for the title row.
currentRow.Background = Brushes.Silver;
currentRow.FontSize = 40;
currentRow.FontWeight = FontWeights.Bold;

// Add the header row with content,
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("Users"))));
// and set the row to span all NUM_COLUMNS columns.
currentRow.Cells[0].ColumnSpan = 3;
table1.RowGroups[0].Rows.Add(new TableRow());
currentRow = table1.RowGroups[0].Rows[1];

// Global formatting for the header row.
currentRow.FontSize = 18;
currentRow.FontWeight = FontWeights.Bold;

// Add cells with content to the second row.
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("User's
ID"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("First
Name"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("Last
Name"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("Gender"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("Date of
Birth"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("Email"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new Run("Mobile"))));
currentRow.Cells.Add(new TableCell(new Paragraph(new
Run("Membership"))));
    }

private void Remove(object sender, RoutedEventArgs e)
{
    MessageWin m = new MessageWin();
    m.Show();
}

private void Update(object sender, RoutedEventArgs e)
{
    SignUpP mainWindow = new SignUpP();
    HomePage.myFrame.NavigatoinService.Navigate(mainWindow);
}

private void NextLvl(object sender, RoutedEventArgs e)
{
    HomePage.myFrame.NavigatoinService.Navigate(new FirstLvl());
}
}

```

HomePage.xaml.cs

```

public partial class HomePage : Window
{
    public static Frame myFrame; // frame
    public MainPage p = new MainPage(); // profile
    public AdminP admin = new AdminP(); // admin profile
    public static MenuItem signOut = new MenuItem(); // menu item: sign out
    public static MenuItem profile = new MenuItem(); // menu item: user's
profile
    public static MenuItem signIn = new MenuItem(); // menu item: sign in
}

```

```

public static MenuItem signUp = new MenuItem(); // menu item: sign up
public HomePage()
{
    InitializeComponent();
    myFrame = frame;
    signOut = so;
    profile = pro;
    signIn = si;
    signUp = su;
    frame.Navigate(new SignUpP());
}

private void Profile(object sender, RoutedEventArgs e)
{
    if (LoginP.isAdmin)
        frame.Navigate(admin);
    else frame.Navigate(p);
}

private void Logout(object sender, RoutedEventArgs e)
{
    SignUpP mainWindow = new SignUpP();
    frame.Navigate(mainWindow);
}

private void MinimizeWin(object sender, MouseButtonEventArgs e)
{
    this.WindowState = WindowState.Minimized;
}

private void Border_MouseDown(object sender, MouseButtonEventArgs e)
{
    if (e.ChangedButton == MouseButton.Left) this.DragMove();
}

private void CloseSwin(object sender, MouseButtonEventArgs e)
{
    App.Current.Shutdown();
}

private void LogIN(object sender, RoutedEventArgs e)
{
    myFrame.Navigate(new LoginP());
}
}

```