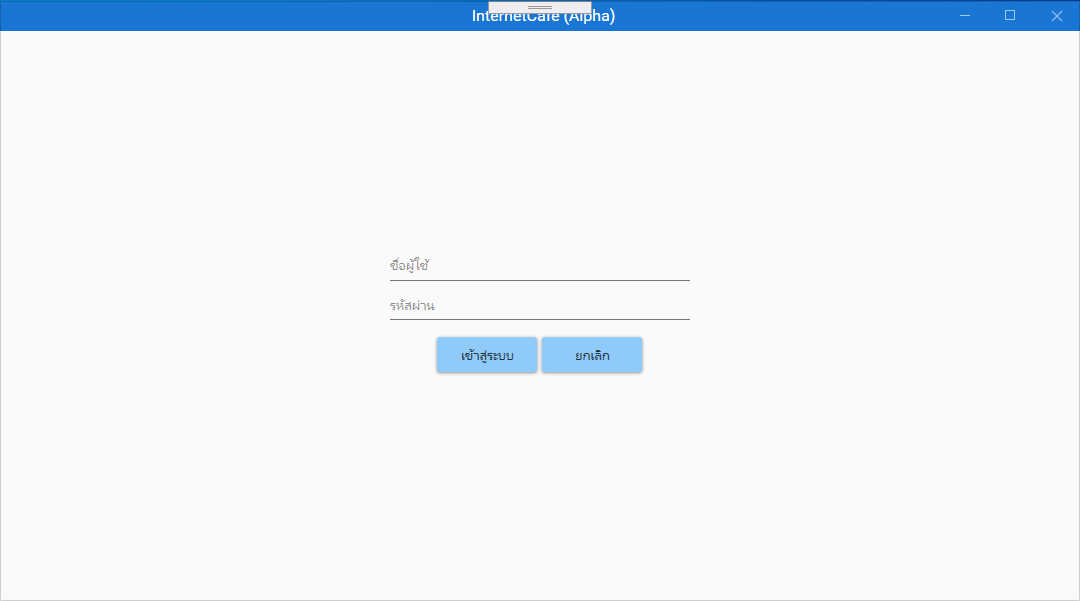
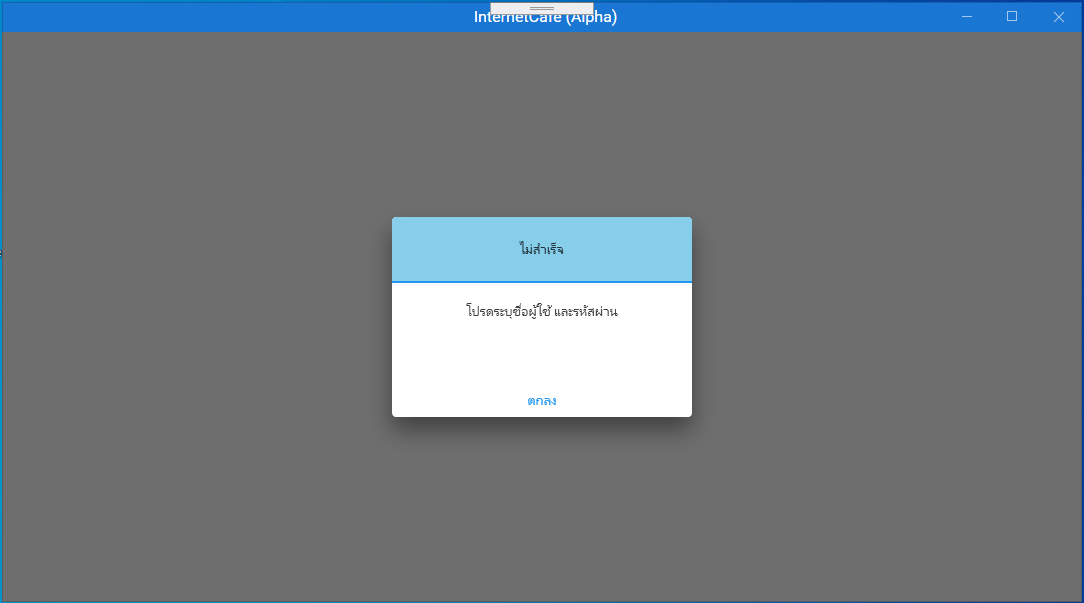
ฝั่งเครื่องแม่

หน้าล็อกอิน

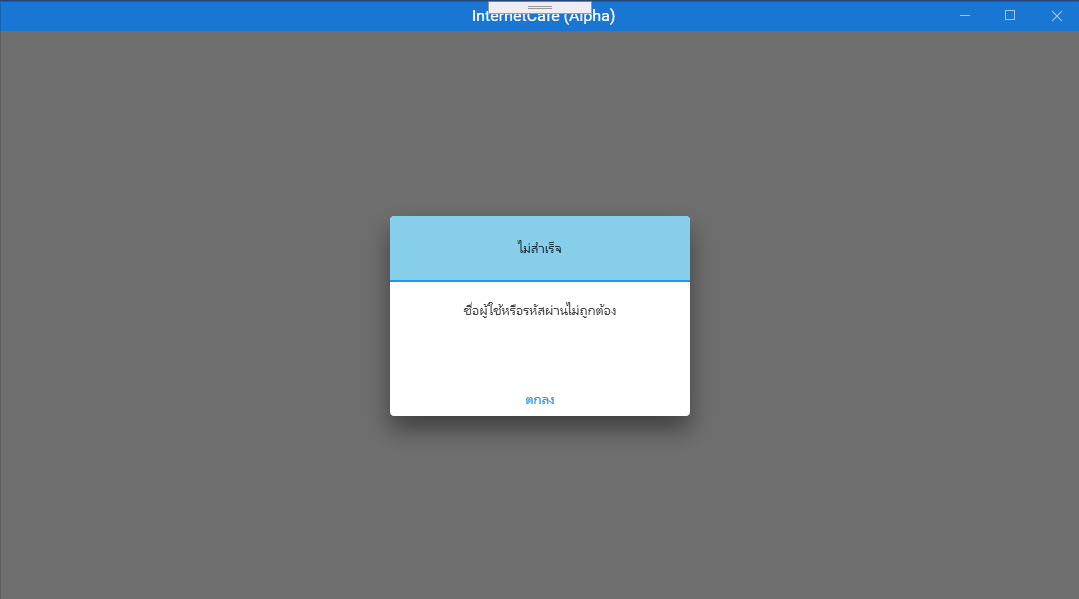
กรณีไม่ระบุข้อมูล



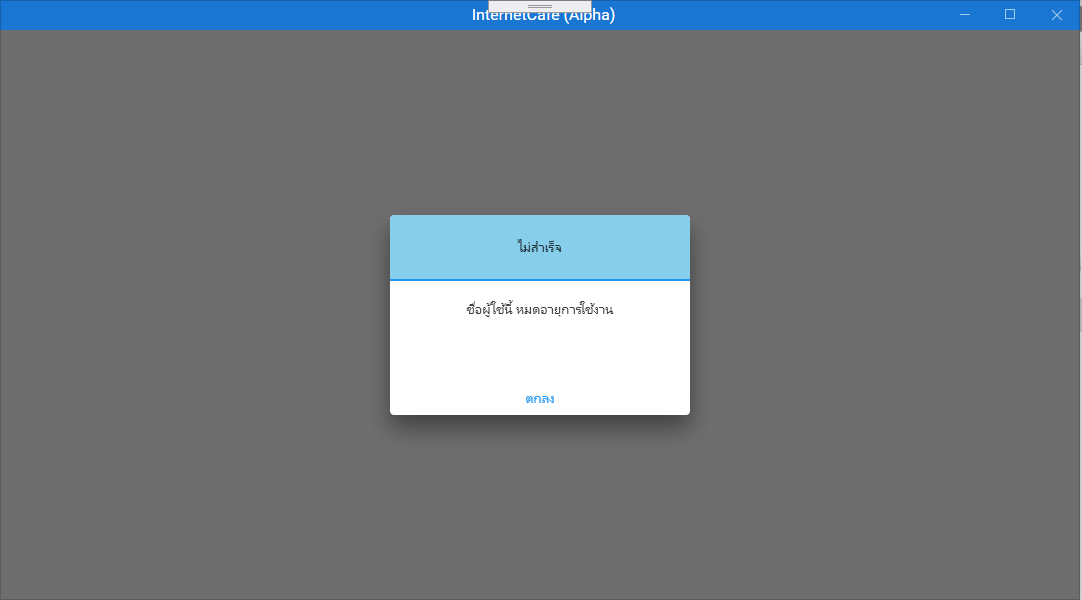
กรณีไม่ระบุข้อมูล หรือระบุข้อมูลไม่ครบ



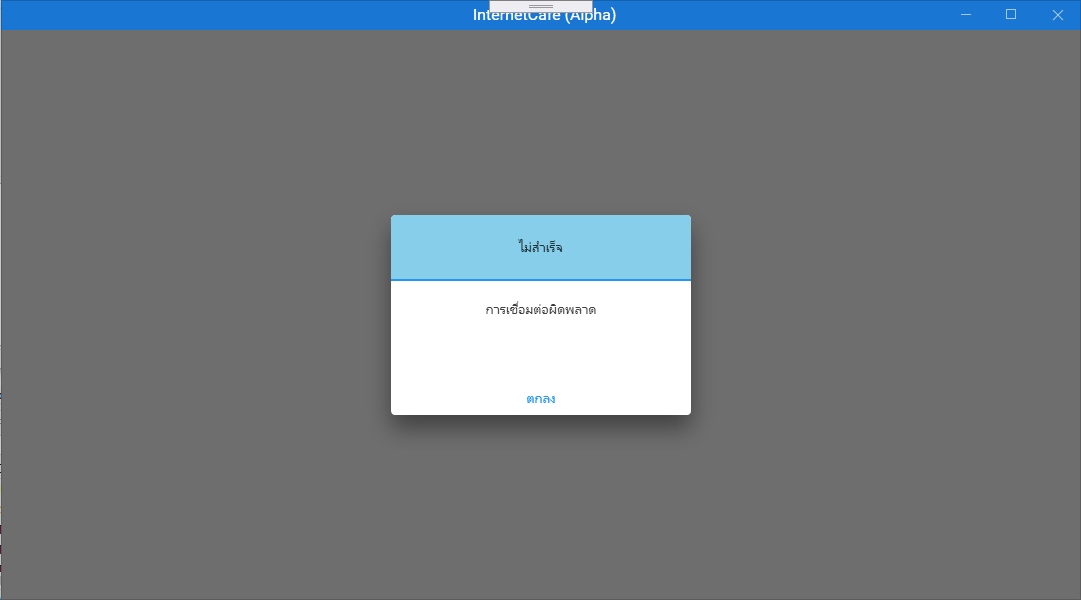
กรณีชื่อผู้ใช้หรือรหัสผ่านไม่ถูกต้อง



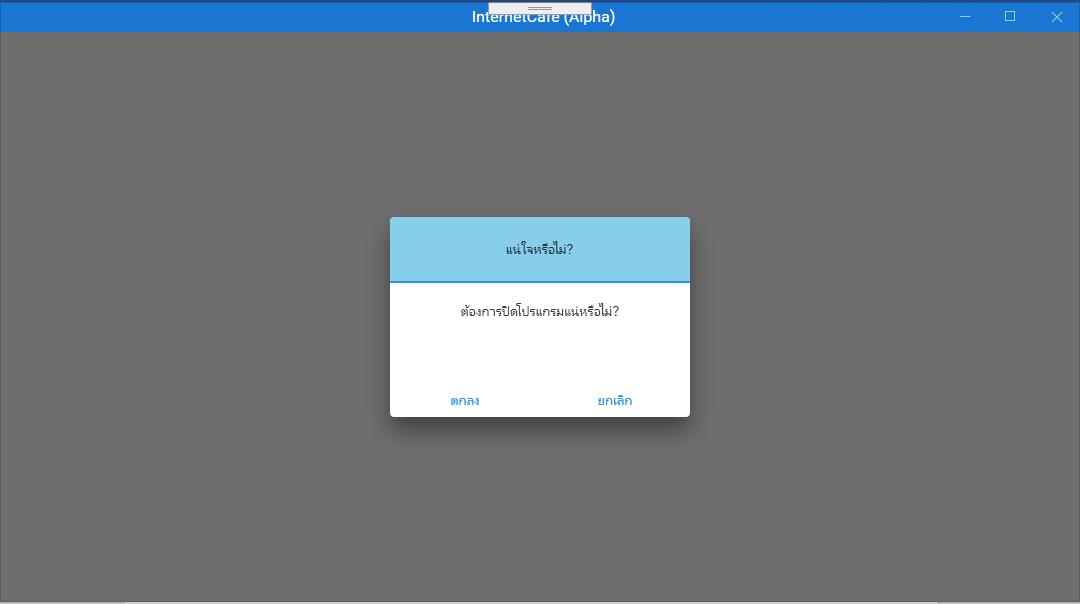
กรณีหมดอายุการใช้งาน



กรณีการเชื่อมต่อผิดพลาด



กรณีกดปุ่มยกเลิก



#region Constructor

public LoginViewModel()

{

pass = new RelayCommand(p=> GoPassChanged(p));

btn\_login = new RelayCommand(p =>

{

if (txt\_username == null || txt\_password == null ||

txt\_username == "" || txt\_password == "")

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("enter\_user\_pass");

DialogHost.Show(new WarningView(), "Msg");

}

else

{

if (IsLoginCheck() == true && data.Rows.Count > 0)

{

login\_data = data.Rows[0];

if (login\_data["exp\_check"].ToString() == "true")

{

if (login\_data["staff\_s\_date"].ToString() == "" &&

login\_data["type\_name"].ToString() != "admin")

{

if (IsUpdate() == true)

{

GoToMain();

}

}

else

{

GoToMain();

}

}

else

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("user\_expire");

DialogHost.Show(new WarningView(), "Msg");

}

}

else if (IsLoginCheck() == true && data.Rows.Count < 1)

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("unsuccess");

DialogHost.Show(new WarningView(), "Msg");

IoC.Application.CurrPage = ApplicationPage.Reset;

IoC.Application.CurrPage = ApplicationPage.Login;

}

}

});

btn\_cancel = new RelayCommand(p =>

{

IoC.ConfirmView.msg\_title = GetLocalizedValue<string>("title\_confirm");

IoC.ConfirmView.msg\_text = GetLocalizedValue<string>("cancel\_confirm");

DialogHost.Show(new ConfirmView(), "Msg", ConfirmClosingEventHandler);

});

}

#endregion

#region other method

public void GoToMain()

{

username = login\_data["staff\_username"].ToString();

login\_id = login\_data["staff\_id"].ToString();

IoC.MainView.btn\_control.Execute("");

IoC.Application.list\_menu = true;

IoC.Application.CurrPage = ApplicationPage.Reset;

IoC.Application.CurrPage = ApplicationPage.Main;

if (login\_data["type\_name"].ToString() == "staff")

{

IoC.OptionView.admin\_check = false;

IoC.OptionView.user\_group\_check = false;

IoC.OptionView.promotion\_check = false;

IoC.StaffView.add\_check = false;

IoC.StaffView.del\_check = false;

IoC.GenerateCouponView.option\_coupon\_check = false;

}

}

public bool IsLoginCheck()

{

string query = $"select \*,if(staff\_e\_date is null or staff\_e\_date >= current\_timestamp(),'true','false') as exp\_check " +

$"from staff " +

$"left join user\_group on user\_group.group\_id = staff.group\_id " +

$"left join type on type.type\_id = user\_group.type\_id " +

$"where staff\_username = '{txt\_username}' and staff\_password = '{txt\_password}' ";

try

{

if (OpenConnection() == true)

{

MySqlCommand cmd = new MySqlCommand(query, Sconn.conn);

MySqlDataAdapter adp = new MySqlDataAdapter(cmd);

DataTable dt = new DataTable();

adp.Fill(dt);

data = dt;

Sconn.conn.Close();

return true;

}

else

{

Sconn.conn.Close();

return false;

}

}

catch (MySqlException ex)

{

Sconn.conn.Close();

if (ex.Number == 0)

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("conn\_unsuccess");

DialogHost.Show(new WarningView(), "Msg");

}

else

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("conn\_unsuccess");

DialogHost.Show(new WarningView(), "Msg");

}

return false;

}

}

public bool IsUpdate()

{

string query = $"update staff set " +

$"staff\_s\_date = '{DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss", new CultureInfo("us-US", false))}' " +

$"where staff\_id = '{login\_data["staff\_id"].ToString()}' ";

try

{

if (OpenConnection() == true)

{

MySqlCommand cmd = new MySqlCommand(query, Sconn.conn);

MySqlDataReader reader = cmd.ExecuteReader();

reader.Close();

Sconn.conn.Close();

return true;

}

else

{

Sconn.conn.Close();

return false;

}

}

catch (MySqlException ex)

{

Sconn.conn.Close();

if (ex.Number == 0)

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("conn\_unsuccess");

DialogHost.Show(new WarningView(), "Msg");

}

else

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("conn\_unsuccess");

DialogHost.Show(new WarningView(), "Msg");

}

return false;

}

}

public void ConfirmClosingEventHandler(object sender, DialogClosingEventArgs eventArgs)

{

if ((bool)eventArgs.Parameter == true)

{

Application.Current.Shutdown();

}

}

public void GoPassChanged(object p)

{

var passwordBox = p as PasswordBox;

string EncryptionKey = "test123456key";

byte[] clearBytes = Encoding.Unicode.GetBytes(passwordBox.Password);

using (Aes encryptor = Aes.Create())

{

Rfc2898DeriveBytes pdb = new Rfc2898DeriveBytes(EncryptionKey, new byte[] { 0x49, 0x76, 0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65, 0x64, 0x65, 0x76 });

encryptor.Key = pdb.GetBytes(32);

encryptor.IV = pdb.GetBytes(16);

using (MemoryStream ms = new MemoryStream())

{

using (CryptoStream cs = new CryptoStream(ms, encryptor.CreateEncryptor(), CryptoStreamMode.Write))

{

cs.Write(clearBytes, 0, clearBytes.Length);

cs.Close();

}

txt\_password = Convert.ToBase64String(ms.ToArray());

}

}

}

public bool OpenConnection()

{

try

{

Sconn.conn.Open();

return true;

}

catch (MySqlException ex)

{

Sconn.conn.Close();

if (ex.Number == 0)

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("conn\_unsuccess");

DialogHost.Show(new WarningView(), "Msg");

}

else

{

IoC.WarningView.msg\_title = GetLocalizedValue<string>("title\_false");

IoC.WarningView.msg\_text = GetLocalizedValue<string>("conn\_unsuccess");

DialogHost.Show(new WarningView(), "Msg");

}

return false;

}

}

public void IsClear()

{

txt\_username = null;

txt\_password = null;

}

public static T GetLocalizedValue<T>(string key)

{

return LocExtension.GetLocalizedValue<T>(Assembly.GetCallingAssembly().GetName().Name + ":resLang:" + key);

}

#endregion