TRƯỜNG ĐẠI HỌC **SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH**

HCMC University of Technology and Education

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

NGÀNH CÔNG NGHỆ THÔNG TIN



***­BÁO CÁO ĐỒ ÁN :***

Manage Readable Document

Member’s group:

1/ Đặng Thị Kim Ngân 18110032

2/ Nguyễn Hoàng Vũ 18110398

Lecture: TS. Huỳnh Xuân Phụng

**TP. HỒ CHÍ MINH, THÁNG**

SCORE

|  |  |  |  |
| --- | --- | --- | --- |
| CRITERIA | CONTENT | PRESENT | TOTAL |
| SCORE |  |  |  |

ASSESSMENT OF THE TEACHERS:

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

……………………………………………………………………………………………………………………………...

***Lời Cảm Ơn***

*We would like to sincerely thank the High Quality Training Department, Ho Chi Minh City University of Technical Education for creating favorable conditions for us to study and complete this project.*

*We would like to express our deep gratitude to Huynh Xuan Phung for giving us wholehearted guidance and guidance during the implementation process.*

*We would like to thank the attention and help of friends during the implementation.*

*Although I have tried to complete the project within the scope and ability to allow but certainly inevitable shortcomings.*

*We hope to receive the sympathy, suggestions and devoted guidance of teachers and friends.*

TP.HCM, ngày 16 tháng 06 năm 2019

Nhóm sinh viên thực hiện

Nguyễn Hoàng Vũ-18110398

Đặn g Thị Kim Ngân-18110032

Table of Contents:

# Catalogue of shapes ………………………………………………

# List of tables………………………………………………………

# Chapter 1: Program Overview……………………………………

## General Introduction

## Software specification

# Chapter 2: Plan……………………………………………………

## Subdivision at work:

# Chapter 3: Software Design………………………………………

## Algorithm

## Design Class

# Chapter 4: Install and Test…………………………………………

# Chapter 5: Conclusion and Development…………………………

## Reference Material**Chapter 1 : Program Overview**

1. *Genneral Introduction*

Request: Manage readable document (like Kinlde)

* Manage : Information of user (username , password, list of document, note ,…)
* Readable document : include all of document that can read like : word(.doc), pdf (.pdf),…

Analysis : Creat database to save information and use control datagridview on winform to show data for user, use Richtextbox and Itext Sharp to read word and pdf file…

1. *Software Specfication*

* Program help open and read document. In addition, user can manage document through a collection with some features : add, edit, delete, note,..
* Input : File Document (link, ID user, name ,..) , User (ID, name, password..)
* Output : List of information about documents which is opened by user.
* Requested by the proposed project.
* Adopt object Oriented programming and new software technologies.
* Lightweight software, stable running.
* Tool and Technology used:

Entity framework

Visual studio 2019 (.NET framework)

Microsoft SQL server management studio 18

Chapter 2 : Plan

1. *Subdivision at work*

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên sinh viên | Desciption work | % Contribute |
| 2 | Đặng Thị Kim Ngân | Desgin Graphic and Report Design database | 40% |
| 1 | Nguyễn Hoàng Vũ | Design Program  Annalysis Interface  Write code  Design 3 layer model  Design and connect to database | 60% |

Chapter 3 : Software Design

**How to do :**

Need to solve 2 importance problem :

*First* : How to show a document on form

Solution : use Open File Dialog to open document and get link, use richtextbox to show the content of that file.

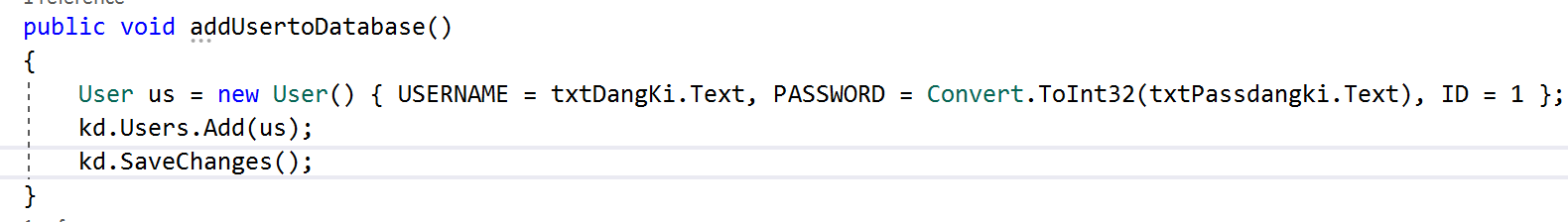
*Second* : Save and change information from form to database and upload data to form.

Solution : use SQL server 2019 to save information of user and document, use entityframework query to load data from database to form and use datagridview to show data.

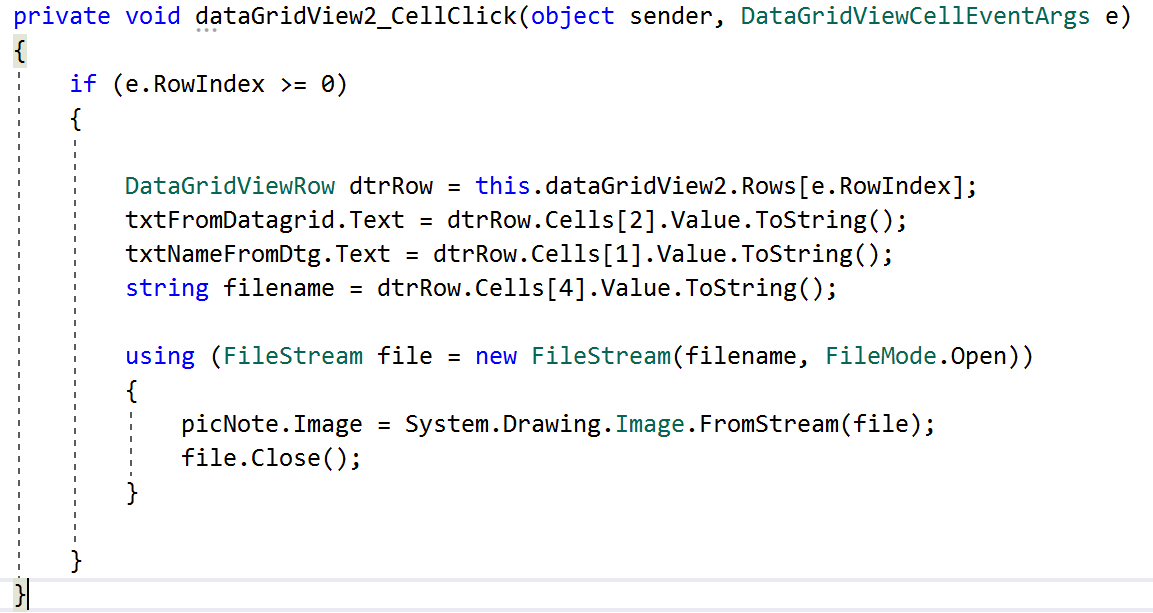
**Alogrithm:**

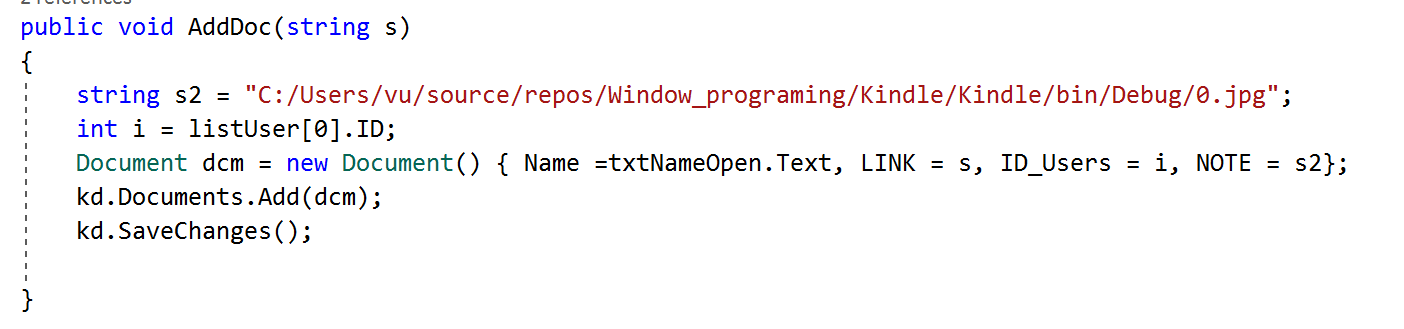
### Save Informatuin from Form to database

Add User



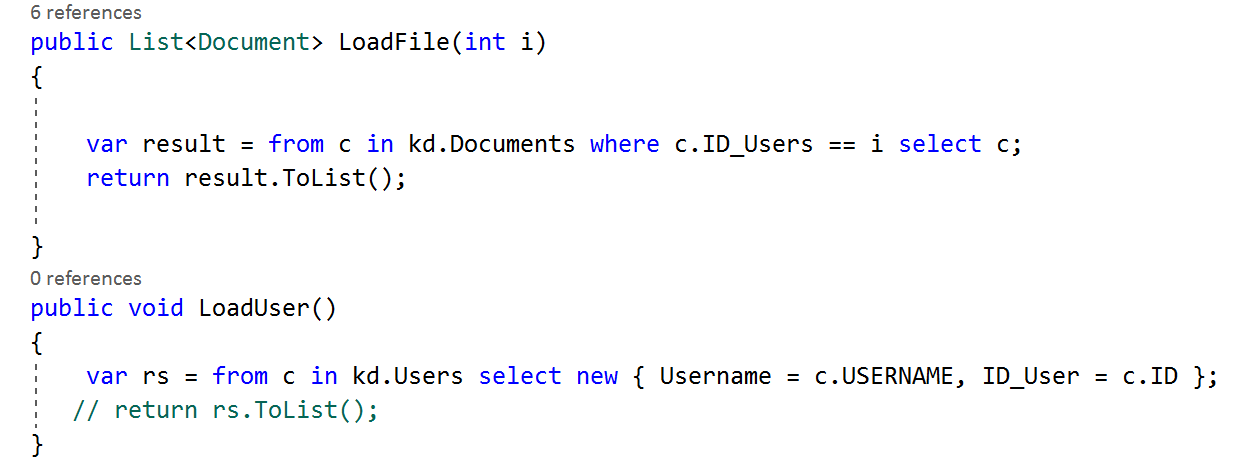
### Algorithm show note (image saved from Bitmap)



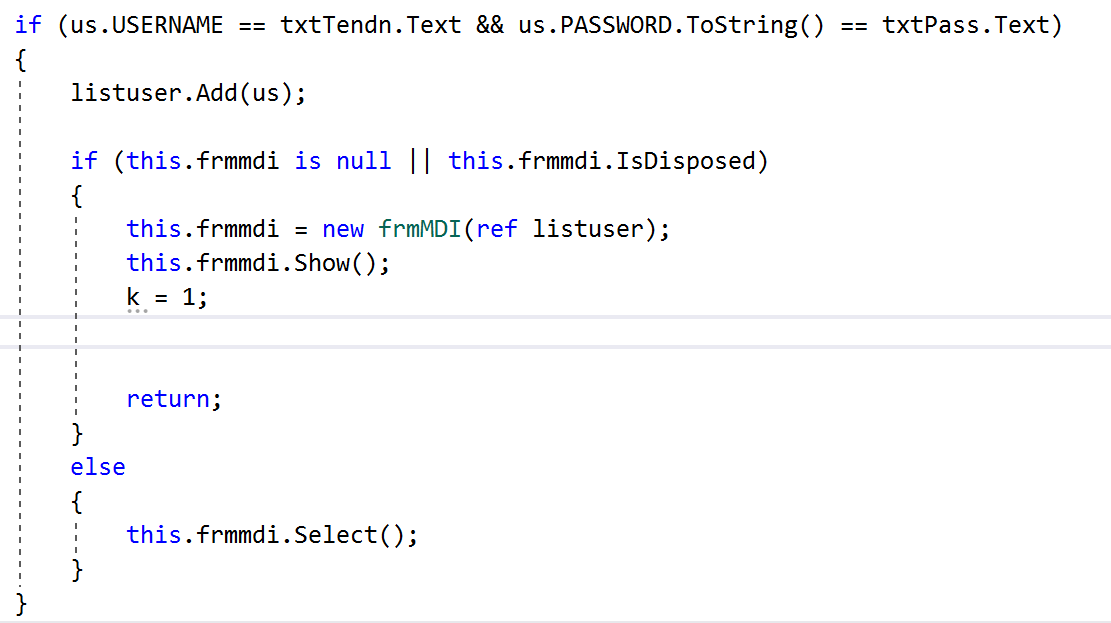


Add Document

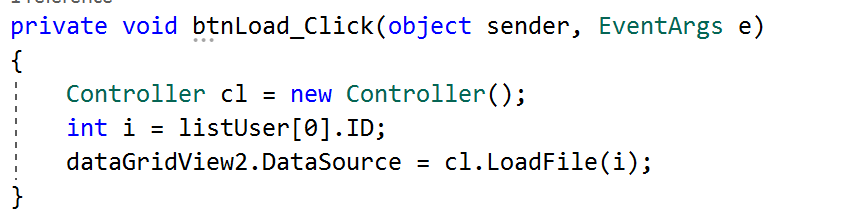
### Show data on datagridview



### Algorithm LOGIN



### Show documents by each user



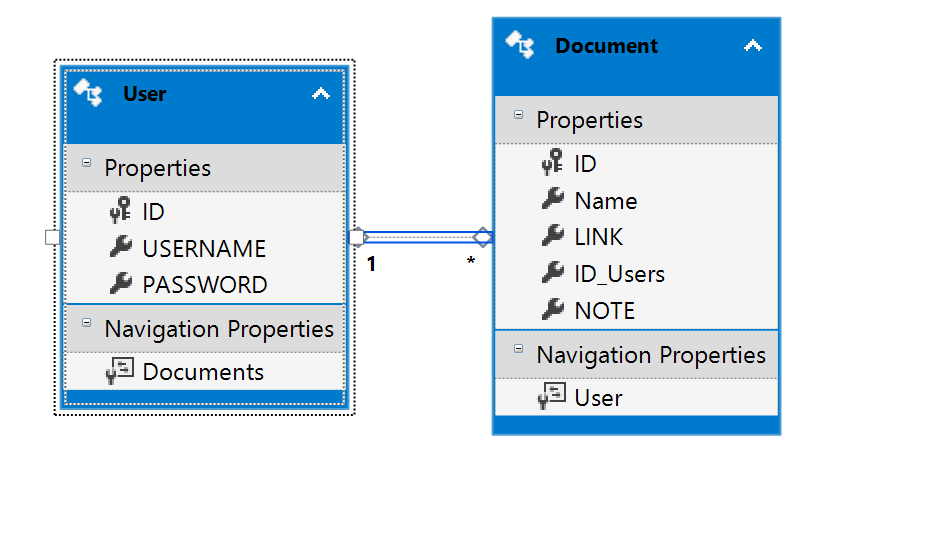
**Design :**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Name** | **Code** | **Function** |
| **1** | **Controllers** | | |
|  |  | class Controller  {  KindleEntities6 kd = new KindleEntities6();    public List<Document> LoadFile(int i)  {  var result = from c in kd.Documents where c.ID\_Users == i select c;  return result.ToList();    }  public void LoadUser()  {  var rs = from c in kd.Users select new { Username = c.USERNAME, ID\_User = c.ID };  // return rs.ToList();  }  public void delete(string s)  {  Document dcm = kd.Documents.Where(p => p.Name == s).FirstOrDefault();  kd.Documents.Remove(dcm);  kd.SaveChanges();  }  public void Add(string s, string s2, string s3, int i)  {  Document dcm = new Document() { LINK = s, Name = s2, ID\_Users = i, NOTE = s3};  kd.Documents.Add(dcm);  kd.SaveChanges();  }  public void Edit(int i , string s1, string s2, string s3)  {  Document dcm = kd.Documents.Find(i);  dcm.Name = s1;  dcm.LINK = s2;  dcm.NOTE = s3;  kd.SaveChanges();  } | Connect database to get data and up to datasource of Datagridview |
| **2** | **Models** | | |
|  | Documents | namespace Kindle  {  using System;  using System.Collections.Generic;    public partial class Document  {  public int ID { get; set; }  public string Name { get; set; }  public string LINK { get; set; }  public int ID\_Users { get; set; }  public string NOTE { get; set; }    public virtual User User { get; set; }  }  } | Basic information of a document |
|  | Users | public partial class User  {  [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]  public User()  {  this.Documents = new HashSet<Document>();  }    public int ID { get; set; }  public string USERNAME { get; set; }  public Nullable<int> PASSWORD { get; set; }    [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]  public virtual ICollection<Document> Documents { get; set; }  } | Basic information of a user |
| 3 | **Views** | | |
|  | frmLogin |  |  |
|  | btnLogin | public void Dangnhap(ref int id, ref string tdn, ref string pas)  {    tdn = txtTendn.Text;  pas = txtPass.Text;  int count = kd.Users.Count();  int k = 0;  for (int i = 1; i <= count; i++)  {  User us = kd.Users.Find(i);    if (us.USERNAME == txtTendn.Text && us.PASSWORD.ToString() == txtPass.Text)  {  listuser.Add(us);    if (this.frmmdi is null || this.frmmdi.IsDisposed)  {  this.frmmdi = new frmMDI(ref listuser);  this.frmmdi.Show();  k = 1;    return;  }  else  {  this.frmmdi.Select();  }  }  }  if (k == 0)  {  MessageBox.Show("Ten Nguoi Dung va Mat Khau khong dung !", "THONG BAO", MessageBoxButtons.OK);  }  } | Check the text in both of 2 textbox txtTendn and txtPass, if they are the same then allow to login |
|  | Form1 |  | To Open file and read Document file( word and pdf) |
| 1 | btnOpenfile | using (OpenFileDialog ofd = new OpenFileDialog() { ValidateNames = true, Multiselect = false, Filter = "Word 97 - 2003 | \*doc |Word Document | \*.docx" })  {  if (ofd.ShowDialog() == DialogResult.OK)  {  string filename = ofd.FileName;  AddDoc(filename);  object readOnly = false;  object visible = true;  object save = false;  object fileName = ofd.FileName;  object newTemplate = false;  object docType = 0;  object missing = Type.Missing;  Microsoft.Office.Interop.Word.\_Document document;  Microsoft.Office.Interop.Word.\_Application application = new Microsoft.Office.Interop.Word.Application()  {  Visible = false  };  rtfData.Visible = true;  document = application.Documents.Open(ref fileName, ref missing, ref readOnly, ref missing, ref missing, ref missing, ref missing,  ref missing, ref missing, ref missing, ref missing,  ref visible, ref missing, ref missing, ref missing, ref missing);  document.ActiveWindow.Selection.WholeStory();  document.ActiveWindow.Selection.Copy();  IDataObject dataObject = Clipboard.GetDataObject();  rtfData.Rtf = dataObject.GetData(DataFormats.Rtf).ToString();  application.Quit(ref missing, ref missing, ref missing);  }  } | When user click OK to open file,  A string which name is filename while be saved to field LINK in table Document in database, richtextbox will show on content inside file document |
| 2 | btnCLosefile | private void btnCloseFile\_Click(object sender, EventArgs e)  {  rtbPDf.Clear();  rtfData.Visible = false;    } | This button will clear all content in richtextbox and hide richtextbox |
|  | Form: frmCollection |  | To show and manage list of Document |
| 1 | yourCollectionToolStripMenustrip | private void yourCollectionToolStripMenuItem\_Click\_1(object sender, EventArgs e)  {  if (this.frmcollection is null || this.frmcollection.IsDisposed)  {  this.frmcollection = new frmCollection(ref this.listUser);  this.frmcollection.MdiParent = this;  this.frmcollection.Show();  }  else  {  this.frmcollection.Select();  }  } | This will show form collection to work with file have opened |
| 2 | btnDelete | private void btndelete\_Click(object sender, EventArgs e)  {  string s = dataGridView2.SelectedCells[0].OwningRow.Cells["Name"].Value.ToString();  Controller cl = new Controller();  cl.delete(s);  MessageBox.Show("Xoa document thanh cong ! Hay Bam Nut (Load) de xem ket qua ");  dataGridView2.DataSource= cl.LoadFile(listUser[0].ID);  } | This button will delete the row selected on datagridview |
| 3 | btnLoad | private void btnLoad\_Click(object sender, EventArgs e)  {  Controller cl = new Controller();  int i = listUser[0].ID;  dataGridView2.DataSource= cl.LoadFile(i);  } | This button will load the data from database to dataSource of datagridview and show |
| 4 | Add | private void button6\_Click(object sender, EventArgs e)  {  Controller cl = new Controller();  if(txtAddDoc.Text == "" || txtAddLinkDOc.Text =="" )  {  MessageBox.Show("Vui long dien thong tin day du !");  return;  }  //bitmapNote.Save(string.Format("{0}.png", listUser[0].ID));  string s = "";  cl.Add(txtAddLinkDOc.Text, txtAddDoc.Text, s, (Convert.ToInt32(listUser[0].ID)));  dataGridView2.DataSource = cl.LoadFile(listUser[0].ID);  } | This button add new doc which information : name, link , note ,…. Add to database |
| 5 | btnUpdate | private void btnOKEdit\_Click(object sender, EventArgs e)  {    Controller cl = new Controller();  int id = Convert.ToInt32(dataGridView2.SelectedCells[0].OwningRow.Cells["ID"].Value.ToString());  string s = string.Format("C:/Users/vu/source/repos/Window\_programing/Kindle/Kindle/bin/Debug/{0}.jpg", id);  cl.Edit(id, txtNameFromDtg.Text, txtFromDatagrid.Text, s);  dataGridView2.DataSource = cl.LoadFile(listUser[0].ID);  MessageBox.Show("Update thanh cong");  } | This button will update name or link or note of selected row in datagridview if user want to change. |
| 6 | btnUpdateNode | private void button9\_Click(object sender, EventArgs e)  {  Controller cl = new Controller();  int id = Convert.ToInt32(dataGridView2.SelectedCells[0].OwningRow.Cells["ID"].Value.ToString());  string s = string.Format("C:/Users/vu/source/repos/Window\_programing/Kindle/Kindle/bin/Debug/{0}.jpg", id);  if (File.Exists(s))  {  File.Delete(s);  }    cl.Edit(id, txtNameFromDtg.Text, txtFromDatagrid.Text, s);  dataGridView2.DataSource = cl.LoadFile(listUser[0].ID);  bitmapNote.Save(string.Format("{0}.jpg",id));  bitmapNote.Dispose();  MessageBox.Show("Update thanh cong ");  } | This button will delete the old bitmapNote and replace by new bitmapnote which is created by user |
| 7 | btnOpen | private void btnOpenfile\_Click(object sender, EventArgs e)  {  //form1 = new Form1(ref listUser);  string path = txtFromDatagrid.Text;  object readOnly = false;  object visible = true;  object save = false;  object fileName = path;  object newTemplate = false;  object docType = 0;  object missing = Type.Missing;  //Microsoft.Office.Interop.Word.\_Document document;  Microsoft.Office.Interop.Word.\_Application application = new Microsoft.Office.Interop.Word.Application();  {  Visible =true;  }  application.Documents.Open(ref fileName, ref missing, ref readOnly, ref missing, ref missing, ref missing, ref missing,  ref missing, ref missing, ref missing, ref missing,  ref visible, ref missing, ref missing, ref missing, ref missing);  } | This button will open the document through link which get from datagridview |
|  |  |  |  |

**Method:**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Name | Code | Function |
| 1 | LoadFile | public List<Document> LoadFile(int i)  {  var result = from c in kd.Documents where c.ID\_Users == i select c;  return result.ToList();    } | Method used to load Document form database |
| 2 | LoadUser | public void LoadUser()  {  var rs = from c in kd.Users select new { Username = c.USERNAME, ID\_User = c.ID };  // return rs.ToList();  } | Method used to  Load User from database to datasource of datagridview |
| 3 | delete | public void delete(string s)  {  Document dcm = kd.Documents.Where(p => p.Name == s).FirstOrDefault();  kd.Documents.Remove(dcm);  kd.SaveChanges();  } | Method to delete data in both of database and datasource of listview |
| 4 | AddDoc | public void AddDoc(string s)  {  string s2 = "C:/Users/vu/source/repos/Window\_programing/Kindle/Kindle/bin/Debug/0.jpg";  int i = listUser[0].ID;  Document dcm = new Document() { Name =txtNameOpen.Text, LINK = s, ID\_Users = i, NOTE = s2};  kd.Documents.Add(dcm);  kd.SaveChanges();    } | This method used to add new document to Document table in database |
| 5 | AddPDF | public void AddPDF(string s)  {  string s2 = "C:/Users/vu/source/repos/Window\_programing/Kindle/Kindle/bin/Debug/0.jpg";  int i = listUser[0].ID;  Document dcm = new Document() { Name = txtPDFname.Text , LINK = s, ID\_Users = i ,NOTE = s2 };  kd.Documents.Add(dcm);  kd.SaveChanges();      } | This method used to add new Pdf file to database |
| 6 | Dangnhap | public void Dangnhap(ref int id, ref string tdn, ref string pas)  {    tdn = txtTendn.Text;  pas = txtPass.Text;  int count = kd.Users.Count();  int k = 0;  for (int i = 1; i <= count; i++)  {  User us = kd.Users.Find(i);    if (us.USERNAME == txtTendn.Text && us.PASSWORD.ToString() == txtPass.Text)  {  listuser.Add(us);    if (this.frmmdi is null || this.frmmdi.IsDisposed)  {  this.frmmdi = new frmMDI(ref listuser);  this.frmmdi.Show();  k = 1;    return;  }  else  {  this.frmmdi.Select();  }  }  }  if (k == 0)  {  MessageBox.Show("Ten Nguoi Dung va Mat Khau khong dung !", "THONG BAO", MessageBoxButtons.OK);  } | This method used to check information to alow user log in or not |

**Design Database:**

****

|  |  |  |
| --- | --- | --- |
| **TT** | **Name of Table** | **Purpose** |
| 1 | Document | Save information of file which is opened by user |
| 2 | User | Save username , password and id user to manage collection |

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Field** | **Type** | **Purpose** |
| 1 | Name | Nvarchar(100) | Save name of document |
| 2 | LINK | Nvarchar(200) | Save Link of document |
| 3 | Id\_Users | Nvarchar(50) | Save Id user of document |
| 4 | NOTE | Nvarchar(200) | Save link of bitmapnote in picture box |

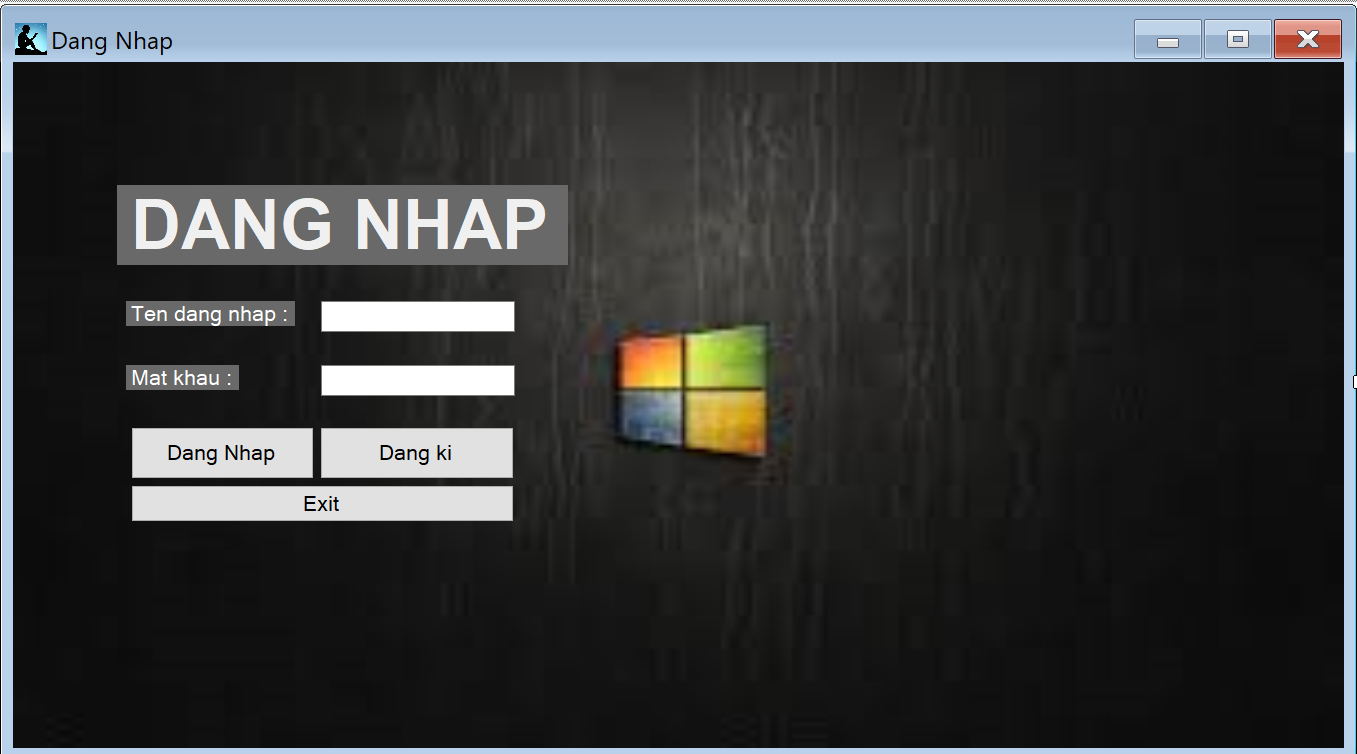
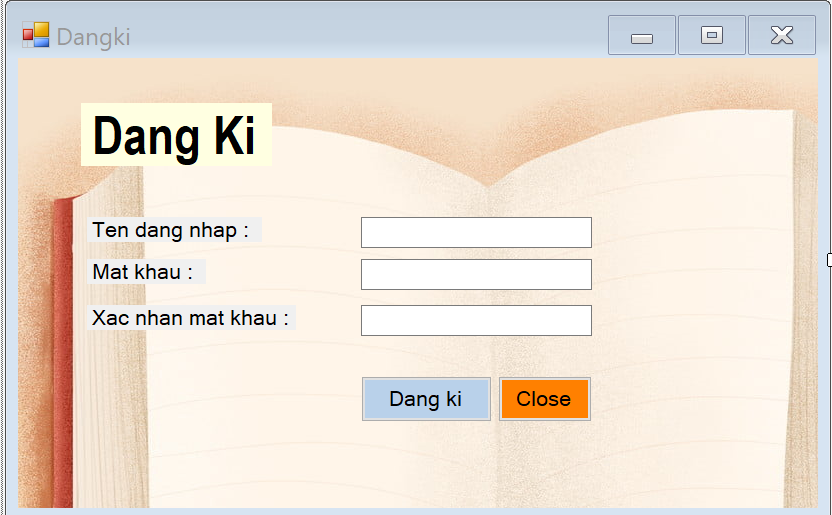
Document table :

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Field** | **Type** | **Purpose** |
| 1 | Username | Nvarchar(100) | Save username of user to login |
| 2 | Password | Nvarchar(100) | Save Password of user to login |

User table:

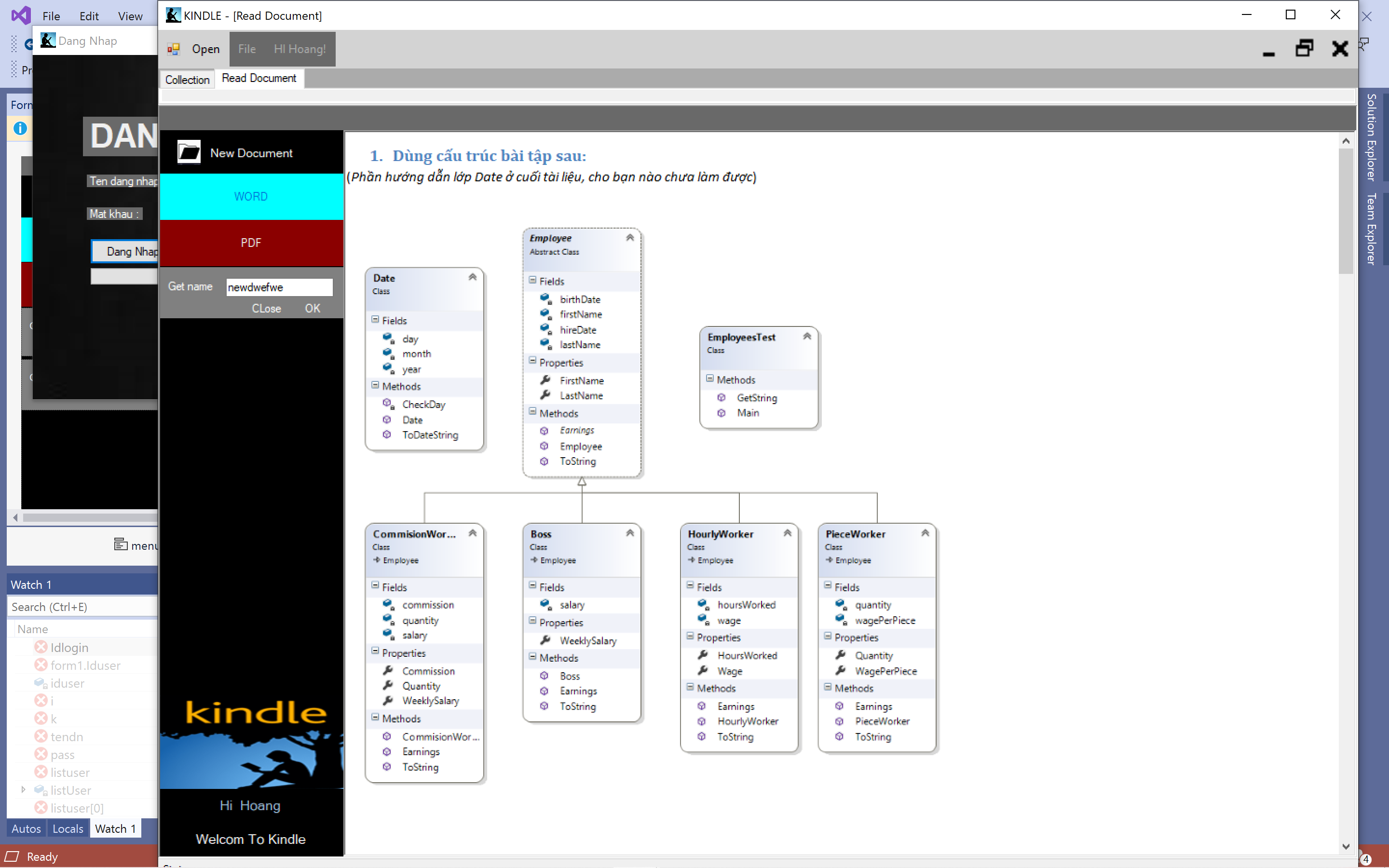
**Design Interface:**

*Form LOGIN and Sign-up:*

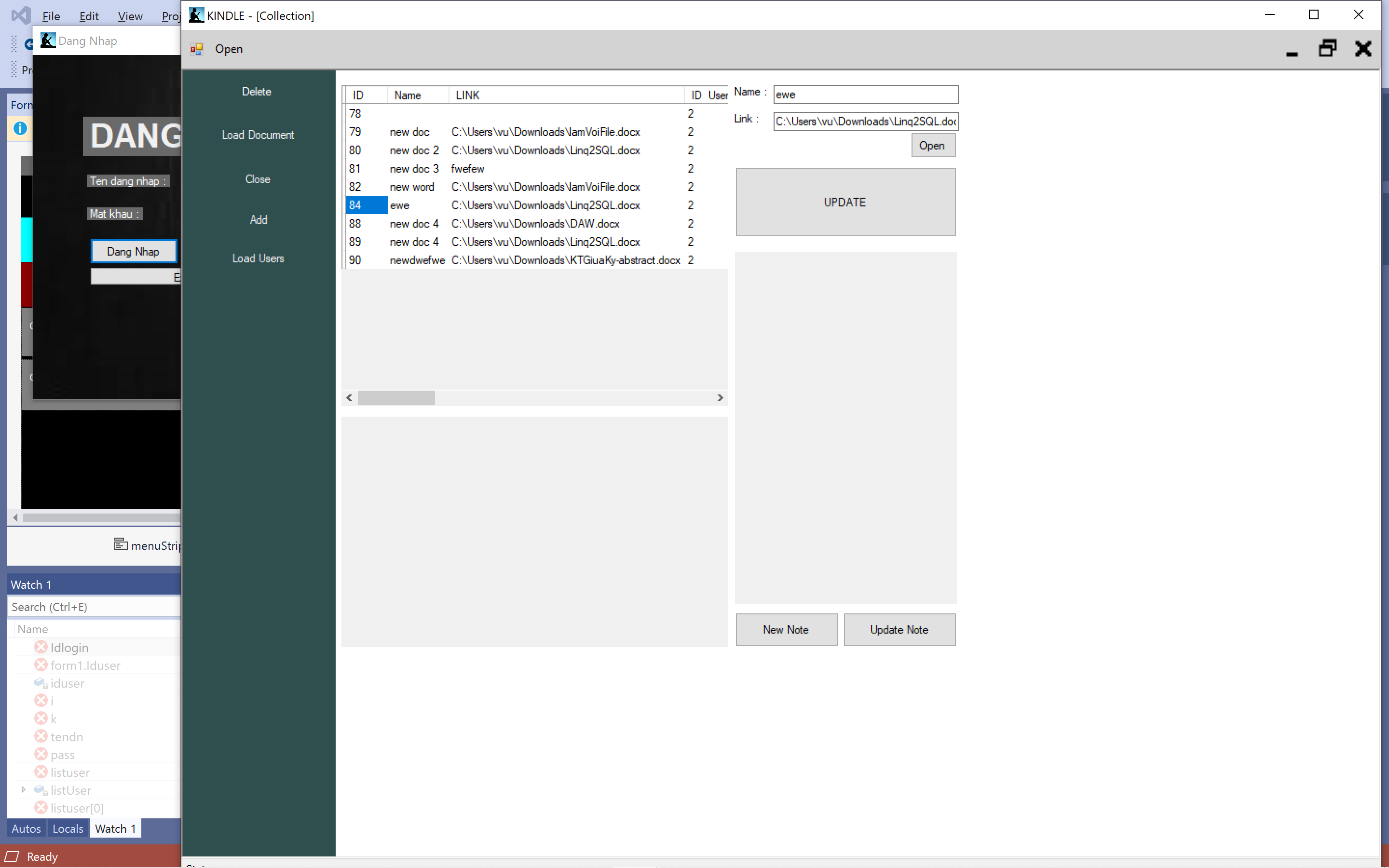
****

|  |  |  |
| --- | --- | --- |
| *Form Read Document:* | To open and read document | Nguyễn Hoàng Vũ  Button and some control are located on the left because all of the right side of form will be used for richtextbox show the content of file document |
| *Form Your Collection:* | To show and manage document have opened | Kim Ngân  List of document need a large region to show all the information so it must be center and buttons are located in the left side. I don’t put the button update to the left because it must near the text box name and link. This will help user more conveniently |

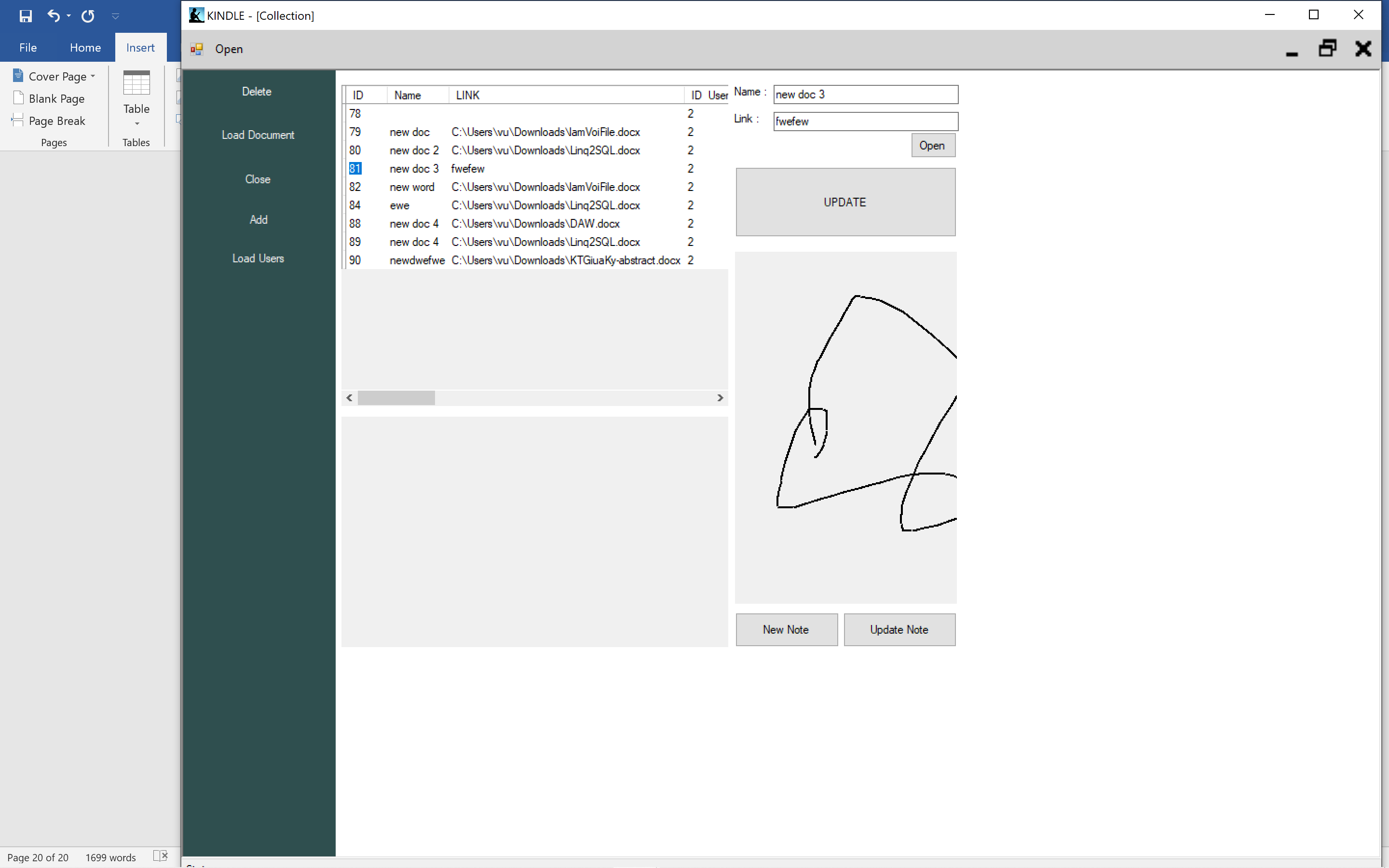
Chapter 4 : Install and Test



Kiểm thử 1 với User có username là “Hoang”



Kiểm thử danh sách các document



Kiểm thử tính năng update Note

Chapter 5 : Development

* Improve the interface
* Personalize the program : Change name, avatar..
* Can open many type of ducument
* Can edit the content of document
* Develop the program similar to the Kindle of Amazon

***Tài liệu tham khảo :***

[1]Entity Framework (Microsoft) <https://docs.microsoft.com/en-us/ef/>

[2] C#: Learn C# in One Day and Learn It Well. C# for Beginners with Hands-on Project. (Learn Coding Fast with Hands-On Project Book 3) Kindle Edition by Jamin Chan <https://www.amazon.com/Beginners-Hands-Project-Coding-Project-ebook/dp/B016Z18MLG>

[3]C# 8.0 and NET.Core 3.0 – Modern Cross Platform Development Fourth Edition(2019).pdf

[4] Documentation windown forms’ on Website Microsoft <https://docs.microsoft.com/en-us/dotnet/framework/winforms/>