Gabriel Molina

**Computer Dictionary**

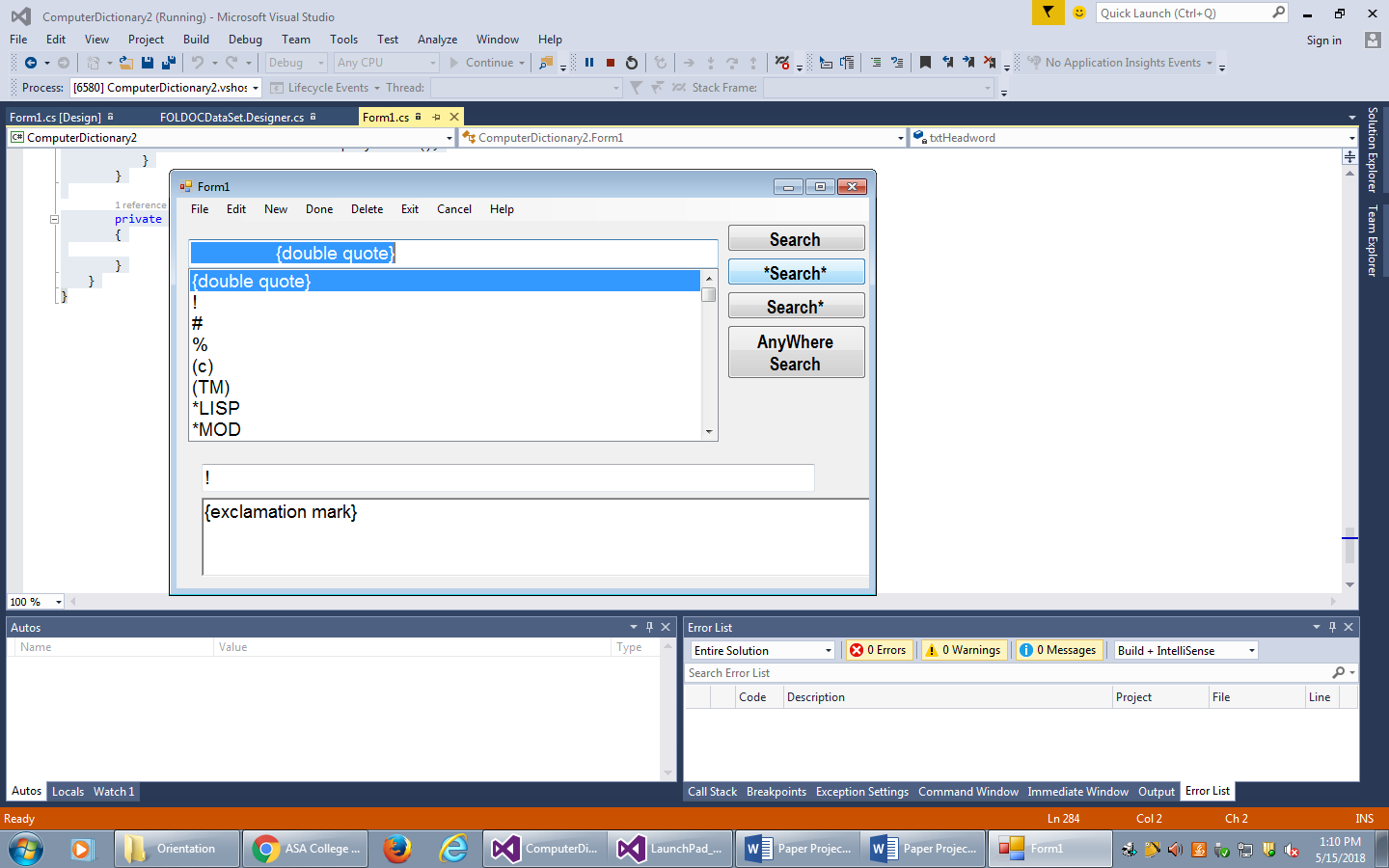
There are so many gadgets or applications that make Windows an excellent Operation System. Windows has several useful applications already installed. These applications can help the people with many common tasks, including browsing the internet, shopping for music, games for entertaining, etc. The Computer Dictionary is a book that lists the words of a language in alphabetical order and gives their meaning, or gives the equivalent meaning. This project describes an application named The Computer Dictionary, it works like the dictionary but instead of a regular vocabulary, the Computer Dictionary’s job is defining the computer languages meaning or vocabulary. If the person has a new word that doesn’t exist, they can also create new meanings or new words and define them.

This application people can use for any computer job. If the person is working as a data administrator, he or she can use for the look a definition that they don’t understand within of database, he or she can go to the Computer Dictionary and look for the word or define a new word. Or a programming teacher that save all the meaning of each term, and when the student isn’t clear with the explanation of any words the student has the possibility to look into the Computer Dictionary that is proportionate for the teacher for a better understanding. Also, a cashier of any store can use this application adding and describing new items.

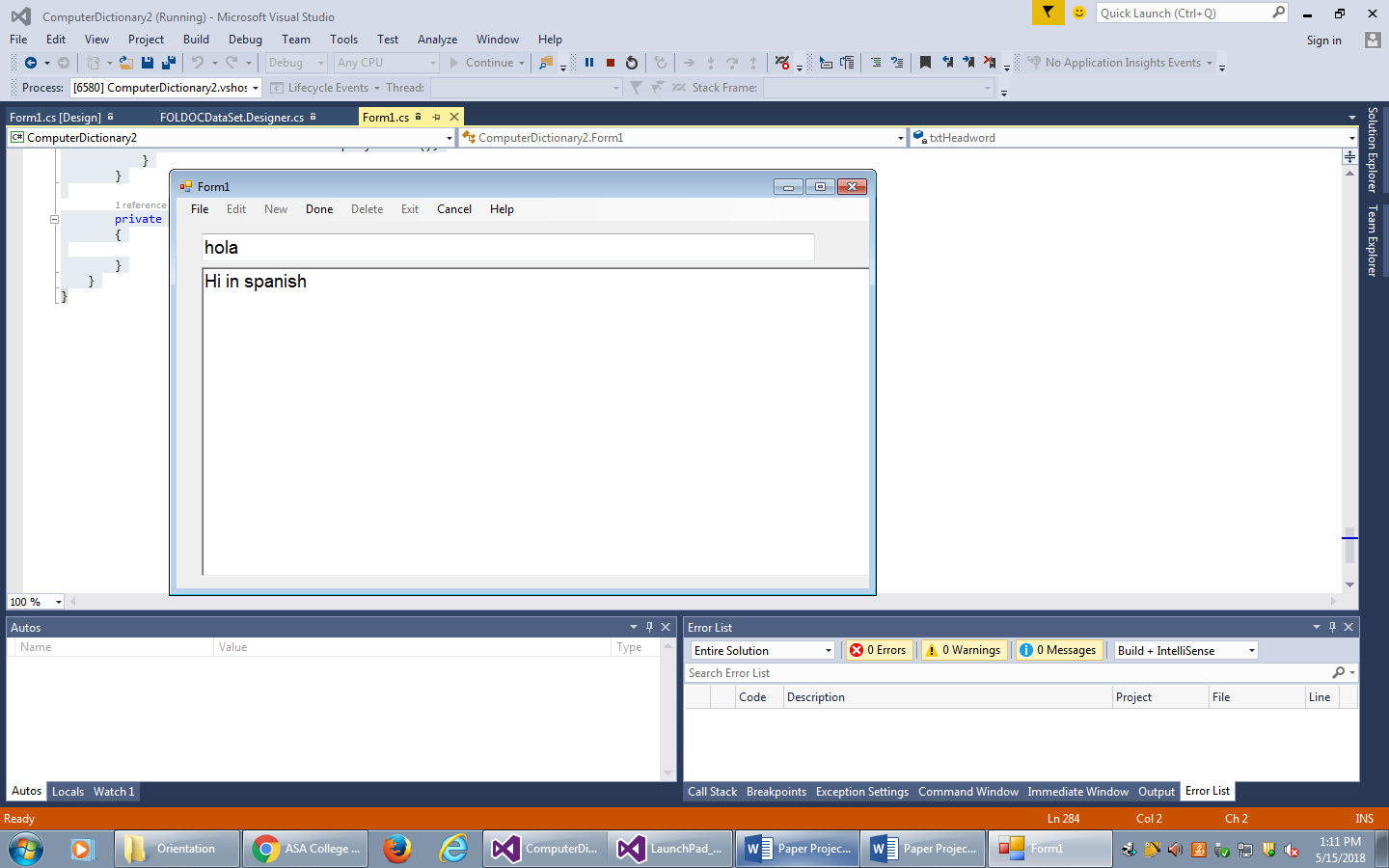
This project has an objective, what it helps the people to have a better understanding of one topic, and people have the words to review them again and again. Also if the person comes with a word that the application doesn’t have, they can always add a new word a meaning

**Screenshots**

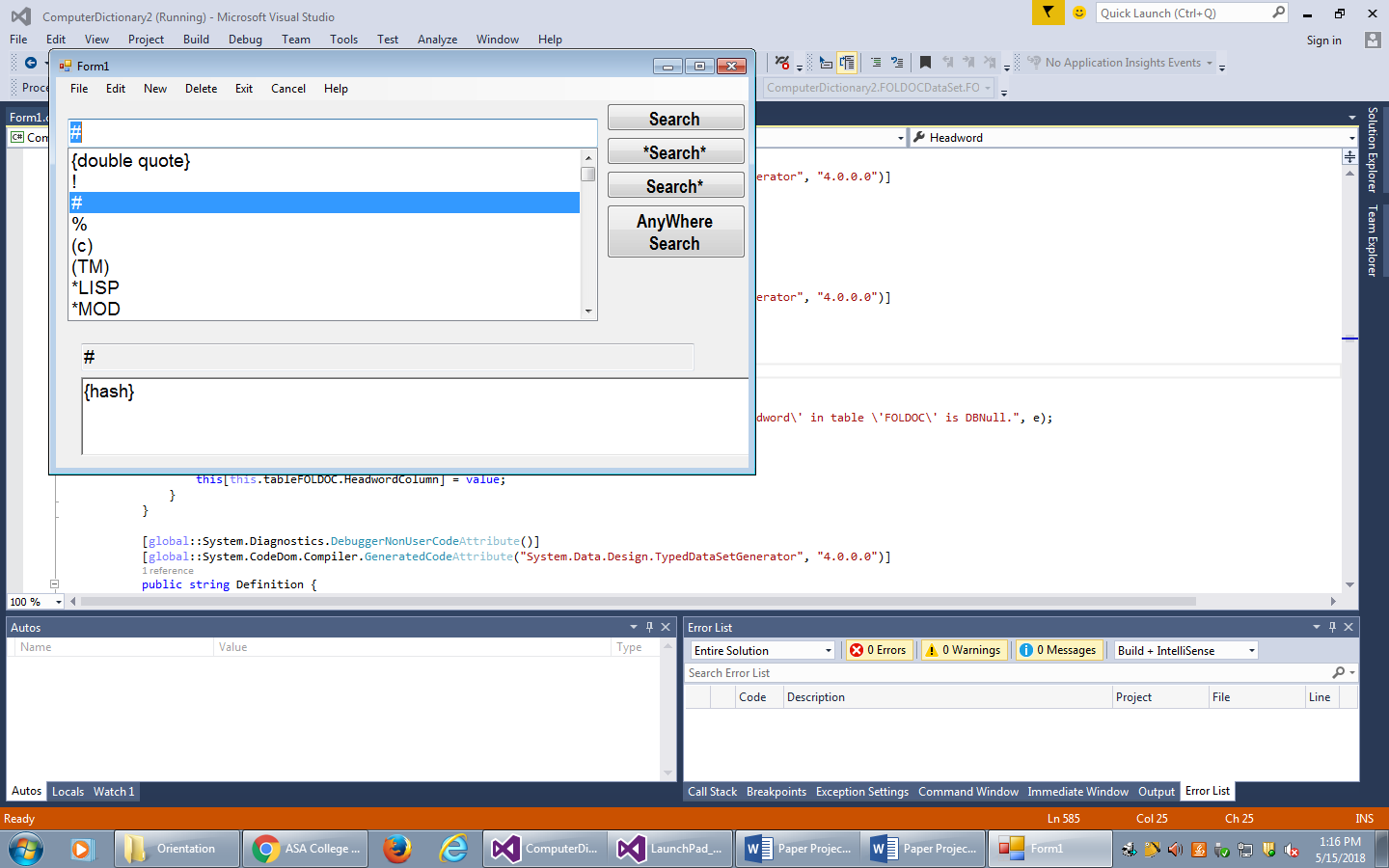
1. Starting the program



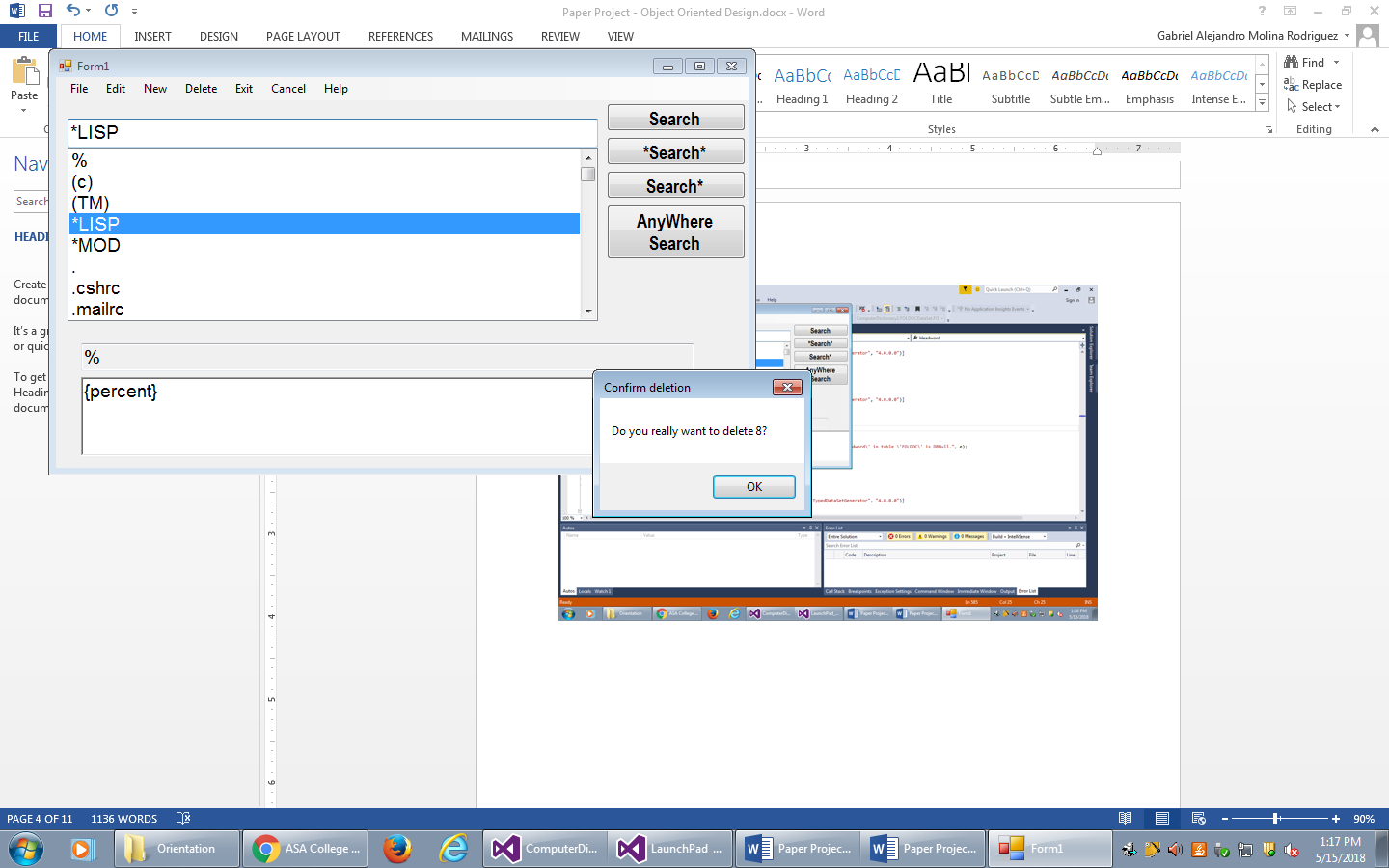
1. New word



1. Searching a word



1. Deleting a word



**Code**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace ComputerDictionary2

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

// Here we create a EditingMode() which means that

// they are going to be allow or not

private void EditingMode(bool newrecord = false)

{

splitContainer1.Panel1Collapsed = true;

exitToolStripMenuItem.Enabled = false;

newToolStripMenuItem.Enabled = false;

editToolStripMenuItem.Enabled = false;

deleteToolStripMenuItem.Enabled = false;

doneToolStripMenuItem.Visible = true;

if (newrecord == true)

{

cancelToolStripMenuItem.Visible = true;

}

txtHeadword.TabStop = true;

rtbDefinition.TabStop = true;

txtHeadword.ReadOnly = false;

rtbDefinition.ReadOnly = false;

txtHeadword.Focus();

}

// Here we create a SearchingMode() which means that

// They are going to be allow or not

private void SearchingMode()

{

splitContainer1.Panel1Collapsed = false;

exitToolStripMenuItem.Enabled = true;

newToolStripMenuItem.Enabled = true;

editToolStripMenuItem.Enabled = true;

deleteToolStripMenuItem.Enabled = true;

doneToolStripMenuItem.Visible = false;

txtHeadword.TabStop = false;

rtbDefinition.TabStop = false;

txtHeadword.ReadOnly = true;

rtbDefinition.ReadOnly = true;

cboHeadword.Focus();

}

// This code here shows all the record

private void ComboBoxRefresh()

{

var query = from word in fOLDOCDataSet.FOLDOC.AsEnumerable()

orderby word.Headword

select word;

cboHeadword.DataBindings.Clear();

cboHeadword.DisplayMember = "Headword";

cboHeadword.ValueMember = "key";

cboHeadword.DataSource = query.ToList();

}

// With this code It helps to end with the editing mode

private void fOLDOCBindingNavigatorSaveItem\_Click(object sender, EventArgs e)

{

this.Validate();

this.fOLDOCBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.fOLDOCDataSet);

}

private void Form1\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'fOLDOCDataSet.FOLDOC'

// table. You can move, or remove it, as needed.

this.fOLDOCTableAdapter.Fill(this.fOLDOCDataSet.FOLDOC);

var query = from word in fOLDOCDataSet.FOLDOC

orderby word.Headword

select word;

cboHeadword.DisplayMember = "Headword";

cboHeadword.ValueMember = "key";

cboHeadword.DataSource = query.ToList();

}

// The Search Button is action that provide a searching to any word in the dictionary

private void btnSearch\_Click(object sender, EventArgs e)

{

// any headword containing whatever typed in txtSearch

if (cboHeadword.Text.Trim().Length > 0)

{

var query = from word in fOLDOCDataSet.FOLDOC

where word.Headword.ToUpper().StartsWith(cboHeadword.Text.ToUpper())

select word;

cboHeadword.DisplayMember = "Headword";

cboHeadword.ValueMember = "key";

cboHeadword.DataSource = query.ToList();

}

}

// The code below is to find any position of each word

private void cboHeadword\_SelectedIndexChanged(object sender, EventArgs e)

{

if (cboHeadword.SelectedValue != null)

{

try

{

int index = fOLDOCBindingSource.Find("key", cboHeadword.SelectedValue);

fOLDOCBindingSource.Position = index;

}

catch (Exception x)

{

MessageBox.Show(x.Message);

}

}

}

// The Search Button is action that provide a searching to any word in the dictionary

// With the first letter

private void btnBeginSearch\_Click(object sender, EventArgs e)

{

if (cboHeadword.Text.Trim().Length > 0)

{

var query = from word in fOLDOCDataSet.FOLDOC

where word.Headword.ToUpper().Contains(cboHeadword.Text.ToUpper())

select word;

cboHeadword.DisplayMember = "Headword";

cboHeadword.ValueMember = "Key";

cboHeadword.DataSource = query.ToList();

}

}

// The Search Button is action that provide a searching to any word in the dictionary in

// anywhere

private void btnAnyWhere\_Click(object sender, EventArgs e)

{

if (cboHeadword.Text.Trim().Length > 0)

{

var query = from word in fOLDOCDataSet.FOLDOC

where word.Headword.ToUpper().Contains(cboHeadword.Text.ToUpper()) ||

word.Definition.ToUpper().Contains(cboHeadword.Text.ToUpper())

select word;

cboHeadword.DisplayMember = "Headword";

cboHeadword.ValueMember = "Key";

cboHeadword.DataSource = query.ToList();

}

}

// The Search Button is action that provide a searching to any word in the dictionary

// With the last letter

private void btnEndSearch\_Click(object sender, EventArgs e)

{

if (cboHeadword.Text.Trim().Length > 0)

{

var query = from word in fOLDOCDataSet.FOLDOC

where word.Headword.ToUpper().StartsWith(cboHeadword.Text.ToUpper())

select word;

cboHeadword.DisplayMember = "Headword";

cboHeadword.ValueMember = "Key";

cboHeadword.DataSource = query.ToList();

}

}

// The code of the Menu, Edit tab can allow the people edit the word

private void editToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

EditingMode(false);

}

// New tab is going to create a new word and definition

// in the dictionary, with the key word AddNew

private void newToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

fOLDOCBindingSource.AddNew();

EditingMode(true);

}

// This code for the Done tab is for save the new word and definition, and if the person

// doesn’t finish with a definition it will show the person a message

private void doneToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

if (txtHeadword.Text.Trim().Length == 0

|| rtbDefinition.Text.Trim().Length == 0)

{

MessageBox.Show(@"Head and/or Definition not completed!

Add more information please.", "Missing infromation reminder");

}

else

{

this.Validate();

this.fOLDOCBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.fOLDOCDataSet);

SearchingMode();

fOLDOCBindingSource.ResetBindings(false);

}

}

// Delete tab button, here the people can delete a word from the dictionary with the Key word RemoveCurrent,

// and many question if the person is not sure about to delete the word

private void deleteToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

if (fOLDOCBindingSource.Position > -1)

{

string currentWord = cboHeadword.SelectedValue.ToString();

DialogResult = MessageBox.Show("Do you really want to delete " +

currentWord + "?", "Confirm deletion");

if (DialogResult == DialogResult.OK)

{

try

{

fOLDOCBindingSource.RemoveCurrent();

fOLDOCTableAdapter.Update(fOLDOCDataSet.FOLDOC);

Form1\_Load(this, null);

}

catch (Exception ex)

{

MessageBox.Show(currentWord + @" not remove." + ex.Message);

}

}

else

{

MessageBox.Show(currentWord + @"not removed. Operation cancelled by user");

}

}

}

// Code for the exit button

private void exitToolStripMenuItem\_Click\_2(object sender, EventArgs e)

{

this.Close();

}

// This code the person can cancel the search and start a new one

private void cancelToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

fOLDOCBindingSource.RemoveCurrent();

SearchingMode();

}

// The About tab button is a comment for a CopyRight with a MessageBox

private void aboutToolStripMenuItem\_Click(object sender, EventArgs e)

{

MessageBox.Show(@"COPYRIGHT 2018 Gabriel Molina, Estate of New York City

INTRODUCTION COPYRIGHT 2018 Gabriel Molina

All rights reserved

No part of this publication may be reproduced, stored in a retrival system

or transmitted in any form or by any means, mechanical, electronic, photo-

copying, recording or otherwise the prior written permission of the

publisher. For information, address Microsoft Co., Inc, 107 West, New York 11020

Piblished By Microsoft Co., Inc

New York/New York

Manufactored in the United States of America

First Coding

Designed by Lixin Shao");

}

}

}

**Recommendations**

This project shows people how a common dictionary looks, but why it can be better? There are a few things that the people can add to this Computer Dictionary to improve it. One of the improvement is to make the Computer Dictionary bilingual, instead to read them in one language the person can read in other languages. This improvement can be applied to putting a new window with any translated page, with that the person can copy and paste the definition and get the definition and the word translated. Also, the programmer can improve with a new button named Favorites, which means that people can save their words and definitions. This is really useful because the person has a reminder list for his or her words. Another improvement is to look for a new word that the people don’t know, it is a big problem when the situation comes. The programmer can create a new button named Help that opens a browser in a window to look the new word.