ECONSTOR
Make Your Publications Visible.

A Service of

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Novac, Ovidiu; Novac, Mihaela

Conference Paper

Software application using .NET platform for a company that produce and sells wine

Provided in Cooperation with:
The Research Institute for Agriculture Economy and Rural Development (ICEADR), Bucharest

Suggested Citation: Novac, Ovidiu; Novac, Mihaela (2012) : Software application using .NET platform for a company that produce and sells wine, In: Agrarian Economy and Rural Development - Realities and Perspectives for Romania. 3rd Edition of the International Symposium, October 2012, Bucharest, The Research Institute for Agricultural Economy and Rural Development (ICEADR), Bucharest, pp. 268-276

This Version is available at:
https://hdl.handle.net/10419/76825

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

SOFTWARE APPLICATION USING . NET PLATFORM FOR A COMPANY THAT PRODUCE AND SELLS WINE

NOVAC OVIDIU[1], NOVAC MIHAELA[2]

Abstract

In the first part of the paper is presented a short review of the main concepts regarding .NET platform and VSTO. In the second part of the paper we present application of the software that will attach a code, to a Word document.The software application attach Office code to a document, for a company that produces and sells wine, using the .NET platform and this application transform a static document, in a dynamic document. The developed application is important in wine sales, because by querying the database application we can obtain different information that can bring good profits and optimize the costs of a company that produces and sells wine.

Key words: *VSTO - Visual Studio Tools for Office, framework, EOM – Excel Object Model, WOM - Word Object Mode*

INTRODUCTION

The software application presented in this paper, attach Office code to a document, for a company that sells wine, using .NET platform, and provides novelty in domain in that it is an application that combines text formatting in a document, attribute that is specific for text editing applications, layout facility, allowing to a user to customize the document. Adding such a code to a document, transforms a static document, in a dynamic document and user can implement different functionalities. Thus this software application is different from other software applications used in online virtual space of commerce, being a new marketing concept.

Viticulture is a production branch of agriculture and through placing on the market of vine and wine products (grapes, raisins, grape juice, wine, wine distillate) we can achieve significant profits. Using a software application for a company that produces and sells wine is necessary because it can optimize costs and increase profits.

We present below some information about .NET platform [2]. From historically point of view, .NET Framework [3] has evolved continuously since his launch, each version added new features and capabilities [4] main ones are illustrated in Figure 1.

In terms of architecture .NET Framework contains three components [1]: Common Language Run-time (CLR), the Base Class Library and finally frameworks and development technologies that are reusable solutions which can customize a wide variety of programming tasks [7].

We can say that .NET Framework [9] is an application development platform that implements an efficient mechanism for memory allocation to store data and instruction [10].

[1] Associate professor Novac Ovidiu Constantin, PhD Eng, University of Oradea, Romania, Department of Decorative Arts and Design, Faculty of Visual Arts, e-mail: ovnovac@uoradea.ro

[2] Lecturer Novac Cornelia Mihaela, PhD Eng, The University of Oradea, Department of Electrical Engineering, Faculty of Electrical Engineering and Information Technology, e-mail: mnovac@uoradea.ro

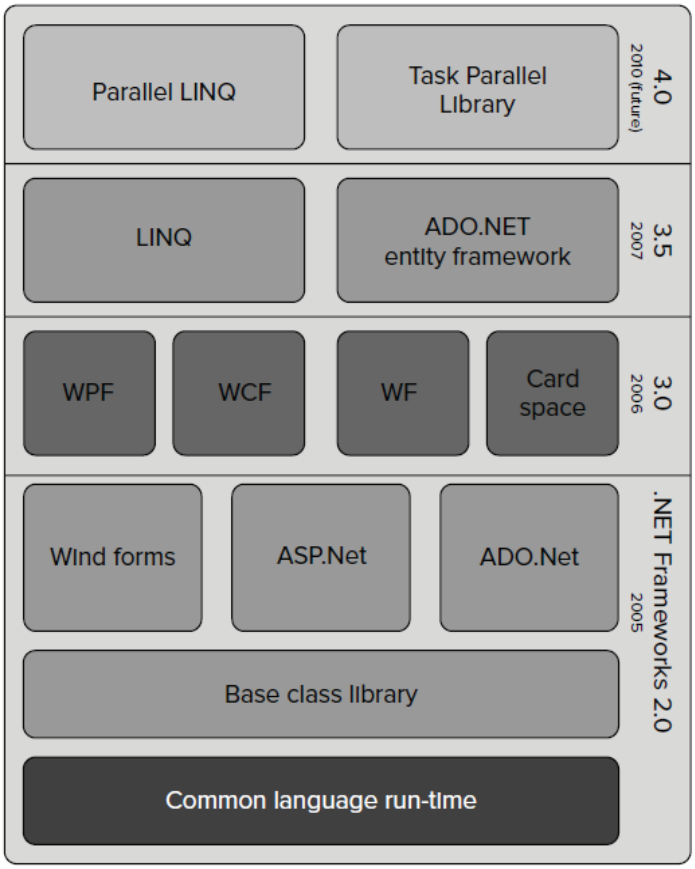

Figure 1. Evolution of .NET Framework platform
Source: Dârdală Marian, 2012

RESULTS AND DISCUSSION

Development of Software Applications based on Microsoft Office

Applications based on programs from Office suite are particularly useful in practice because they use existing facilities of Microsoft Office applications. To develop such applications or extensions on .NET platform, there was built a framework known as Visual Studio Tools for Office (VSTO). It provides .NET programming support for Word, Excel, Outlook, PowerPoint, Project, Visio and InfoPath in Visual Studio. VSTO framework allows Word and Excel documents to use .NET programming features such as support for data binding, controls that can be used in the windows form's etc. Writing code for applications based on Office require the use of a model known as OOM - Office Object Model. This model contains a set of classes and objects needed to control Office applications. Models are customized according to applications they control, for example EOM - Excel Object Model, WOM - Word Object Model etc. In general, these models contain a hierarchy of classes and are organized so that in the root of the hierarchy Application class is located, that models behaviour of a particular application from Microsoft Office software package. In addition to Application class which is present in all hierarchies, there are particular classes in the hierarchy that depend on the application itself and correspond to entities that are actually manipulated in Office applications. For an Excel application [2], the main classes and their relationships are shown in Figure [5].

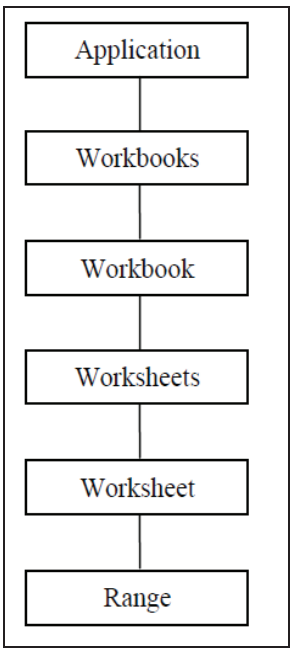

Figure 2. Excel Object Model hierarchy model
Source: Dârdală Marian, 2012

For Word application, the main classes and their relationships are shown in Figure 3.

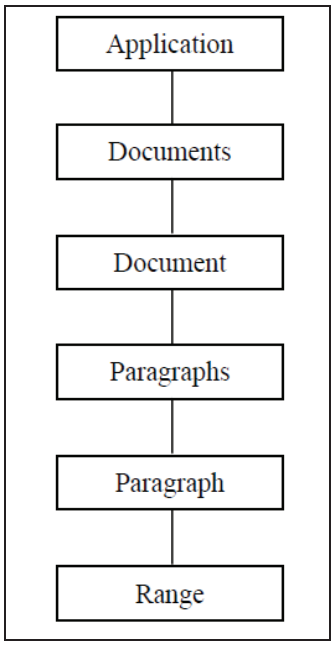

Figure 3. Word Object Model hierarchy model
Source: Dârdală Marian, 2012

Using a series of OOM programming model [6] applications can be developed in different ways [12]:

- Construction of existing functional modules as dynamic link libraries (DLL - Dynamic Link Library).
- User applications that interact with Office applications [8] .
- Attaching code to Office documents.

Presenting Software Application that attach code to a document for a company that produces and sells wine using Net platform

Software application will attach a code to a Word document. This application was developed using the framework Visual Studio Tools for Office (VSTO) from the .NET platform and is designed for a company that produces and sells wine. Software application was performed to define custom operations that will be performed only on the data of a Word document well defined. The application will build a document template that will insert specific controls to a Word document, such a user to be able to customize the document. The application will attach to the document a Document Actions type window that will contain two buttons with custom functionality: *Add data in table (Adauga date in tabela)* and *Upload data in document (Incarca date in document).*

We have developed an application in Visual Studio 2010, building a project based on Office 2010, of Word 2010 Document type. Application made by us will be attached to a document, building a new document with Visual Studio interface, document called WordDocument5.

Whether we are in design mode, Word application interface will allow us to edit a new document as we can see in Figure 4. The document is built as a template because it has a fixed part represented through regular text and a part that will customize the information about wine which is represented in the document, through appropriate controls for interaction. In this case we have a document template for describing a variety of wine through the following ten characteristics: *wine category (categoria de vin), origin vineyard (podgoria de proveniență), wine type (tipul de vin), grade (clasa de calitate), manufacturer (producator), alcoholic strength (taria alcoolică), crop year (an recoltă), stock (stoc), price (preț) and quantity of grapes (cantitate de struguri).* For data editing, we assign controls from the Word Controls group existing in the window ToolBox of Visual Studio with these features. For fields: *wine category (categoria de vin), origin vineyard (podgoria de proveniență), grade (clasa de calitate), manufacturer (producator), alcoholic strength (taria alcoolică), crop year (an recoltă), stock (stoc), price (preț) and quantity of grapes (cantitate de struguri)* were used Plain Text Content Control type controls in order to edit values, while for the *wine type (tipul de vin)* field we have used Drop Down List Content Control type control, to choose from several options value (in our case we have three values: *White (Alb), Red (Rosu)* and *Rose (Rose).*

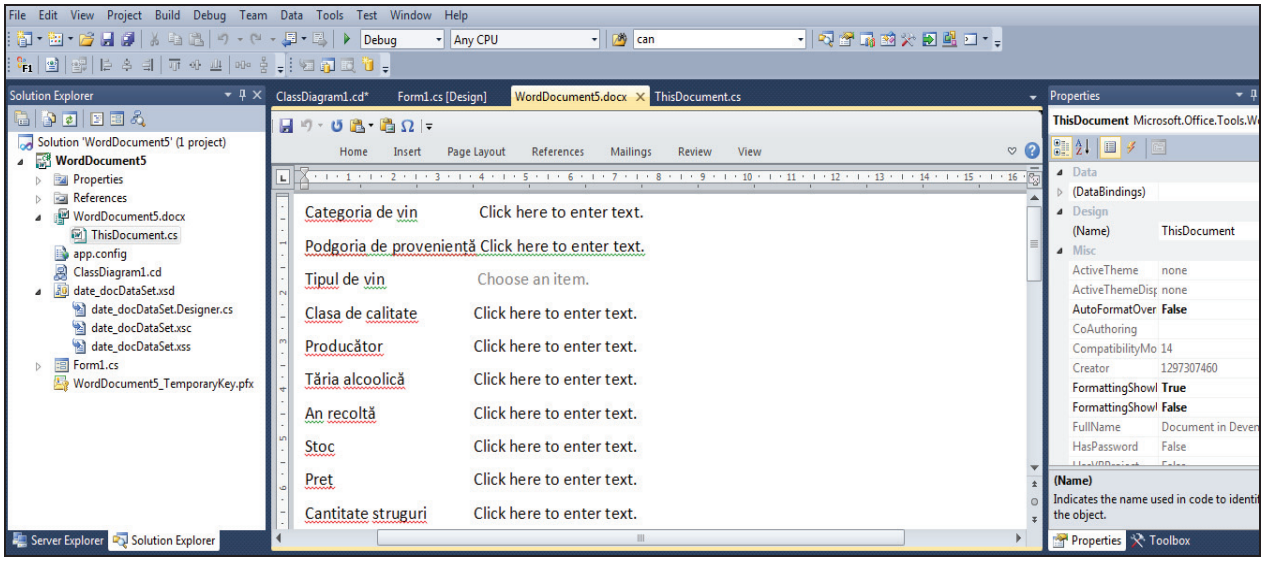

Figure 4. Editing the new Word document

A new document WordDocument5.docx is created and it has an associated This Document class that allows to any programmer to respond to events triggered in working with the document and to extend document functionality. Data will be introduced in document controls, and these will be retrieved and will be stored in a database called date_doc.accbd. This database was created in

Access program from Microsoft Office suite, creating a table called *Wines (Vinuri)*. *Wines* table fields are: *ID, wine category (categoria de vin), origin vineyard (podgoria de proveniență), wine type (tipul de vin), grade (clasa de calitate), manufacturer (producator), alcoholic strength (taria alcoolică), crop year (an recoltă), stock (stoc), price (preț) and quantity of grapes (cantitate de struguri).*

We will build two buttons: *Add data in table (Adauga date in tabela)* and *Upload data in document* (*Incarca date in document). Add data in table (Adauga date in tabela)* button serves as a trigger backup operation on the database. *Upload data in document* (*Incarca date in document)* button is used to load data from Word template table.

By launching the application from Visual Studio we will load the document template in Word, as can be seen in Figure 6. On the right side of the document, in Figure 5, we can observe that appeared the Document Actions form and in it appeared two buttons that were defined in the method described above.

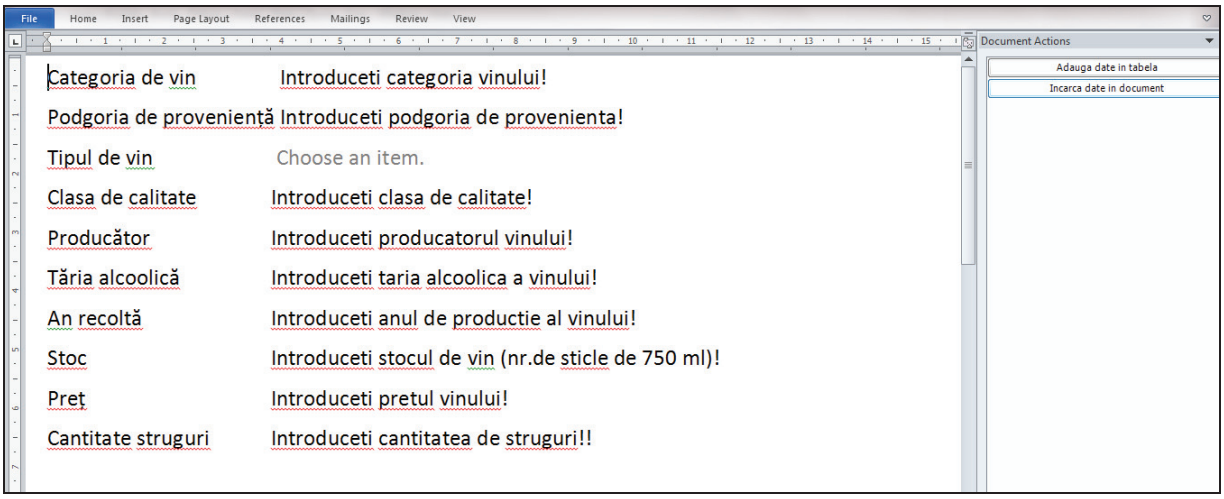

Figure 5

Next controls will be filled with data from the document, and if is pressed the *Add data in table (Adauga date in tabela)* button the data will be added to the existing data in *Wines (Vinuri)* table from date_doc.accdb database. To enter data into the *type of wine (Tipul de vin)*, the user must choose one of three available options *White (Alb), Red (Rosu)* or *Rose (Rose).* If the user does not select any option, it is advised that it must choose something from the list, to add data in the table. The existing options in the selection list for the *Type of wine (Tipul de vin)* field can be seen in Figure 6.

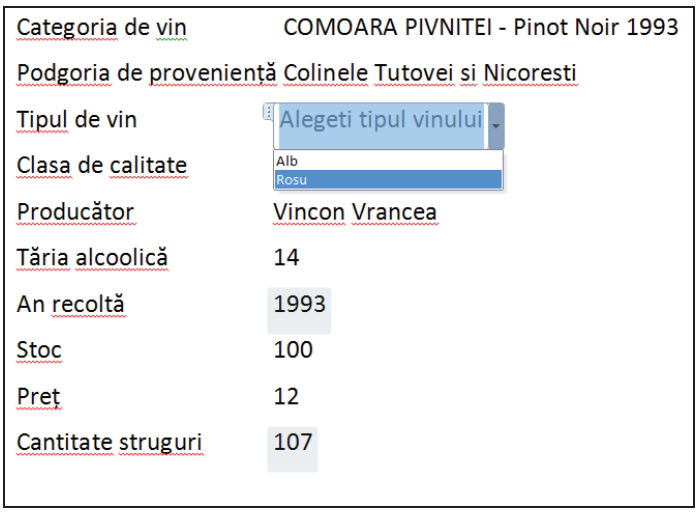

Figure 6 Selection options for *Type of wine (Tipul de vin)* field

Retrieving data from table and loading them into Word document template is done by pressing the *Upload data in document (Incarca date in document*), which will cause a dialog window that displays a DataGridView from which the user can select their desired tuple.

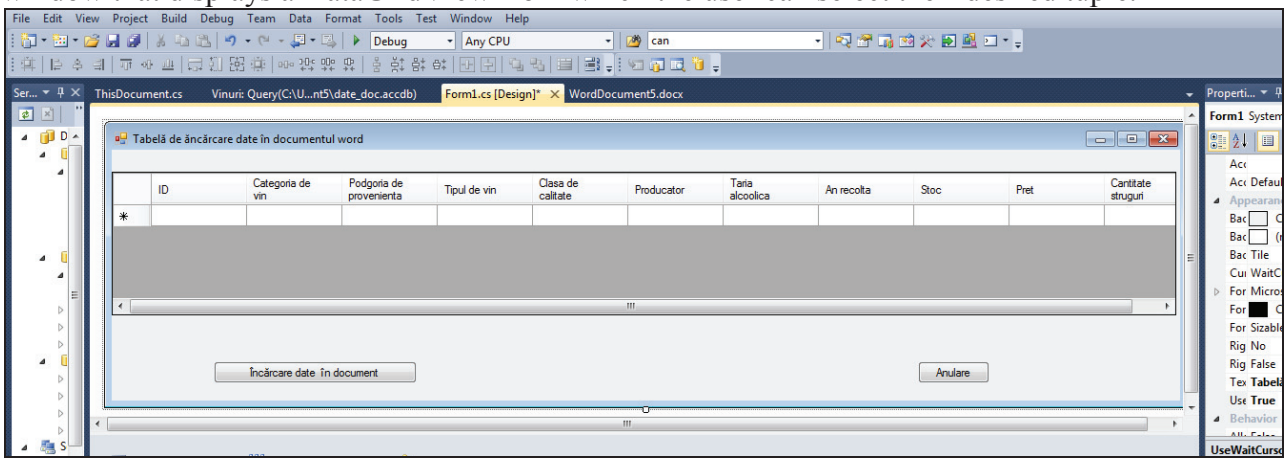

Figure 7 Dialog window for data selection

To create *Upload data in document* (*Incarca date in document)* button functionality, we must add a new form to our project through Windows Form application from Visual Studio.

New Form1 form is created, the name Form1 is the default name given by Visual Studio. We will add to this form a DataGridView type control (gv) that will bind to *Wines* (*Vinuri)* table for displaying tuples. Fields that will be displayed on the grid are: *Wine category (Categoria de vin), Origin vineyard (Podgoria de proveniență), Wine type (Tipul de vin), Grade (Clasa de calitate), Manufacturer (Producator), Alcoholic strength (Taria alcoolică), Crop year (An recoltă), Stock (Stoc), Price (preț) and Quantity of grapes (Cantitate de struguri).* The selection of a tuple from DataGridView type control marks that data will be loaded into the Word document, which is reflected by pressing *Load data in the document* (*Încărcare date în document)* button. The second operation possible is cancellation of selection by pressing *Cancel (Anulare)* button. Dialog window is closed by pressing either of two buttons. In Figure 7 we can see the newly created form that contains selection grid for tuples. In this figure we can see the two buttons, *Upload document data* (*Încărcare date în document)* and Cancel *(Anulare*). We will introduce in document template new data wine categories, as is shown in Figure 8.

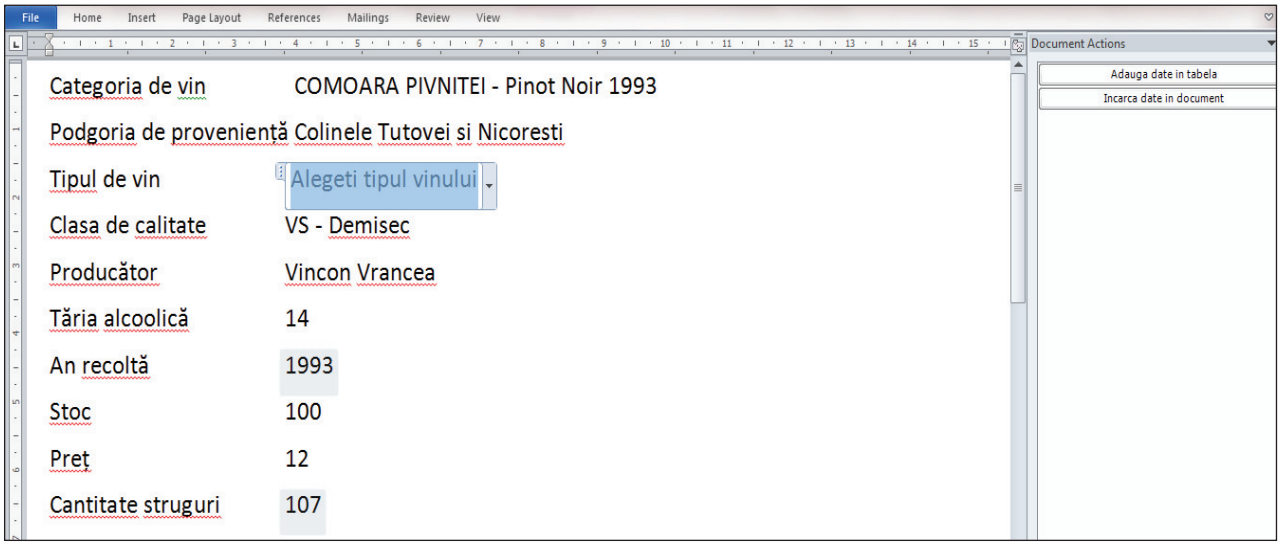

Figure 8

If the new wine type is not introduced, we will press *Upload document data* (*Încărcare date în document)* button in the document and a message appears in the window shown in Figure 9

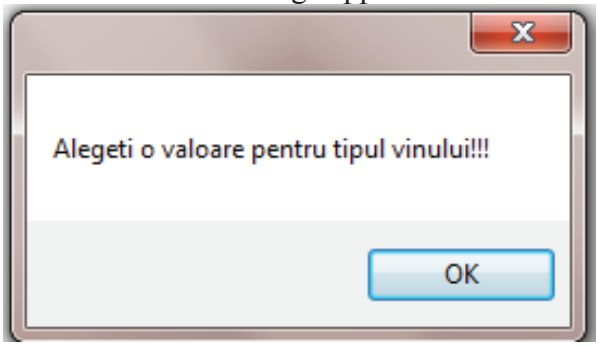

Figure 9

After that we will introduce *wine type (tipul vinului),* which in our case is *Red (Rosu)*, we have in the Word document all information about a new category of wine, as can be seen in Figure 10.

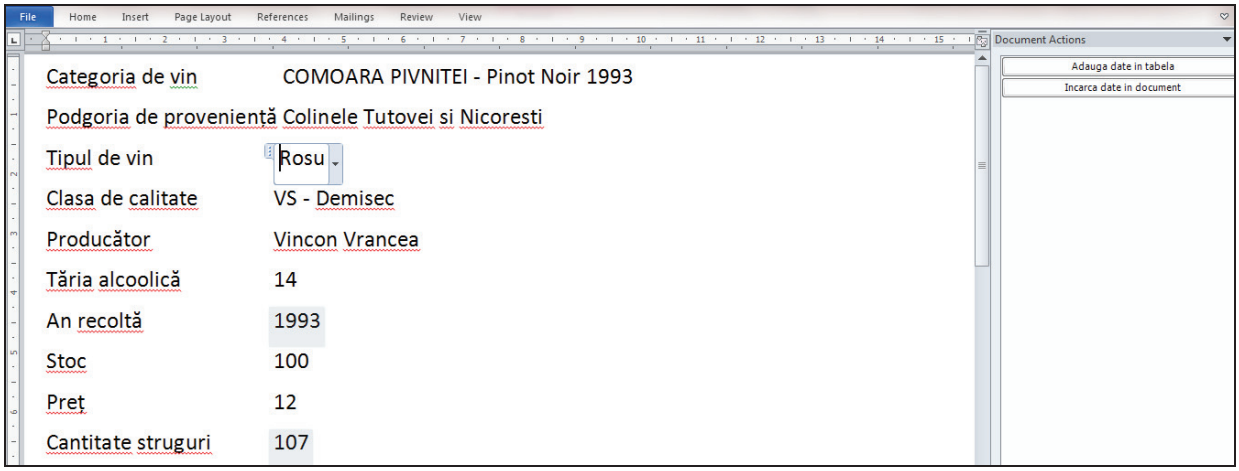

Figure 10

It is time to introduce new data in the table. This is done by pressing the *Upload data in document (Incarca date in document)* the effect will be the input and displaying data in table successfully, as can be seen in Figure 11.

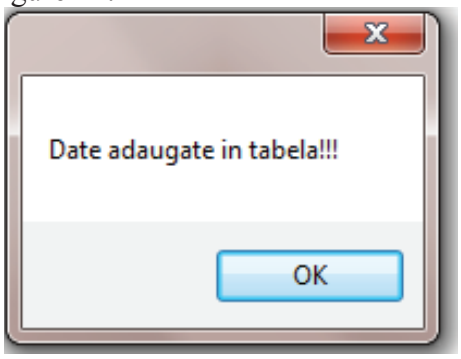

Figure 11

Retrieval of data from the database and uploading them into the Word document is presented below, as we can see in Figures 12 and 13. From Document Actions form of Word document, we press the second button, *Upload data to document (Incarca date in document)* and dialog window is now built into the project with grid control populated with data from table *Wines*

(Vinuri) of date_doc.accdb database. An application user may select a tuple from grid as can be seen in Figure 12, and then it has two options: either to load selected data or cancel the operation.

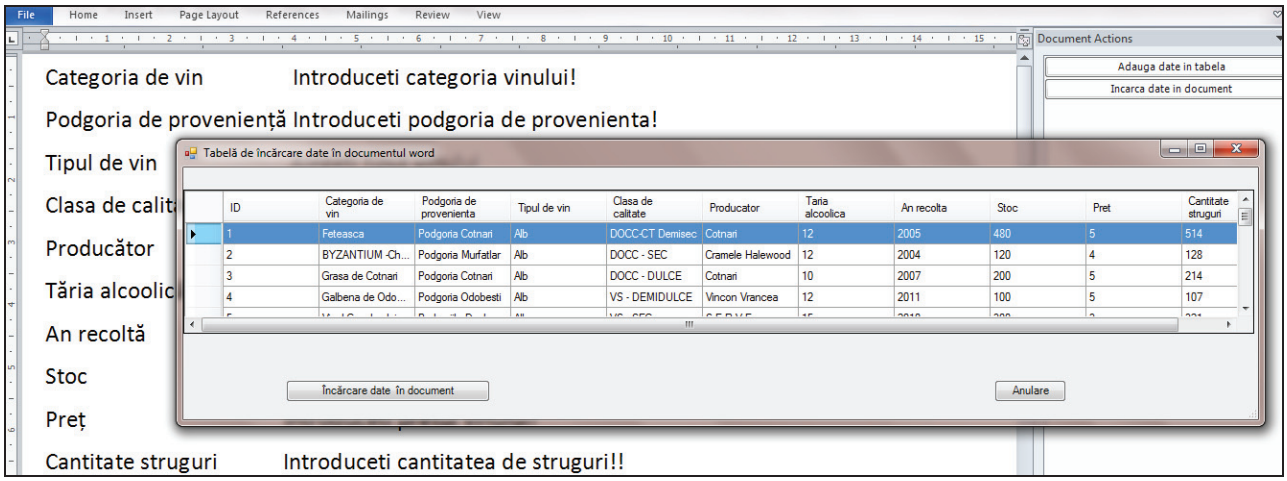

Figure 12

Therefore, by pressing the *Upload data into the document (Încărcare date în document)*, we will load the selected data in the Word document and by clicking Cancel *(Anulare)*, we can cancel the operation.

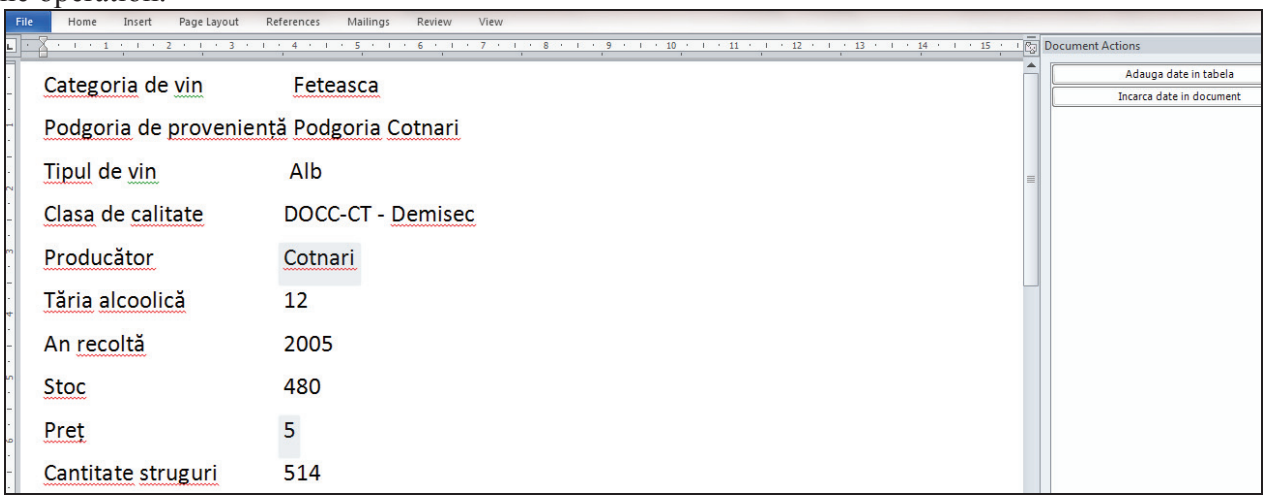

Figure 13 Uploading selected data in document

In Figure 13 we can see that after we have pressed the *Upload data into the document (Încărcare date în document)*, the selection window that contain the selected grid data will be closed and will be loaded in the appropriate fields from the Word document.

CONCLUSIONS

The software application develops an application based on Word and it has characteristic that custom code sequence will run in the same process with the Office application. Code associated documents does not need to be registered, however it is associated with the document by adding specific properties which are stored in the document file. When the Word document is loaded then we consult properties and we load the associated code of document. We have developed an application on .NET platform using the Visual Studio Tools for Office framework (VSTO). VSTO programming supports. NET for Word, Excel, Outlook, PowerPoint, Project, Visio and InfoPath in Visual Studio, and in our application we have used for a Word application. Within our application

for a company that produce and sells wine, I noticed how Visual Studio Tools for Office framework, allows Word documents to use programming features from .NET platform.

Through this method we combine text formatting in a document, application-specific attribute which is performing for text editing, with facility layout, that allow users to customize the document. Therefore software application presented in this paper is different from software used in online virtual space of commerce, being a new marketing concept, in that if we add such a code to a document, it will convert the document from a static document, in a dynamic document and thus its user can implement specific functionality.

The application developed is important in wine sales, because after entering data relating to wine in the Word file, i.e. after the introduction of wine ten features of wine: *wine category (categoria de vin), origin vineyard (podgoria de proveniență), wine type (tipul de vin), grade (clasa de calitate), manufacturer (producator), alcoholic strength (taria alcoolică), crop year (an recoltă), stock (stoc), price (preţ) and quantity of grapes (cantitate de struguri)* this data will be inserted automatically into the *Wines (Vinuri)* database. By querying this database we can get different information, such as stocks come from a particular producer, the grapes needed to be purchased from a vineyard to produce a certain quantity of wine.

Another advantage would be an information extraction from our database that comes directly to our Word document, where it can be further processed. All the information that can be extracted from the application software can optimize costs and can make good profits a company that produces and sells wine.

ACKNOWLEDGEMENT

This work was co-financed from the European Social Fund through Sectorial Operational Programme Human Resources Development 2007-2013, project number POSDRU/89/1.5/S/56287 „Postdoctoral research programs at the forefront of excellence in Information Society technologies and developing products and innovative processes", partner University of Oradea.

REFERENCES

[1] Carter, E., Lippert, E. (2009). Visual Studio Tools for Office 2007, Addison-Wesley
[2] Dârdală Marian. (2012). Medii de dezvoltare şi framework-uri pentru dezvoltarea aplicațiilor software, Curs on-line, http://www.posdru56287.org/elms/mod/resource/view.php?id=250
[3] Novak, I.et all. (2010). Visual Studio 2010 and .NET4 SIX-IN-ONE, Wiley Publishing, Inc.
[4] Pruteanu Alexandru. (2002). Dezvoltare de aplicatii in Visual Studio .NET Iasi
[5] Serban Iulian, Brezoi Dragos, Radu Tiberiu, Ward Adam. (2006). GDI+ Custom Controls with Visual C# 2005, PACKT Publishing, ISBN 1-904811-60-4
[6] http://dotnet.xhost.ro/
[7] http://tminfo.ro/upload/files/410682583Manual_final.pdf
[8] http://cursuri.cs.pub.ro/~cpl/Curs/CPL-Curs01.pdf
[9] http://www.scribd.com/doc/55621239/44/Introducere
[10] http://ti.utm.md/upload/profs/files/762.pdf