Software instrument used as interface in the design of technical installations

L Popa 1* and V Popa 2

1 University of Transilvania, Brașov, Department of Automation and Information, Transilvania University of Brașov, Mihai Viteazul Street No.5, 500036, Romania
2 Regional Department for Defense Resources Management Studies, Brașov, Mihai Viteazul Street No.130 500183, Romania

E-mail: mluminita2001@yahoo.com

Abstract. Technical installations can account for up to half of the building costs and it is crucial to have detailed projects for the assembly and maintenance of these systems. This means that workflow can be improved more efficiently and faster. With the help of the installation engineer, the beneficiary can track and control the correct execution of the works. The installation engineer is the link between the beneficiary and the contractor or those who deliver the materials and equipment, people who generally follow works and additional sales. As regards the design of technical installations, the development of versatile low-response software interfaces becomes necessary and appropriate. The article is focused on the interface that has been developed as a software instrument to increase design process productivity and optimization of technical installations by reducing the required work time and the number of clicks made. Inside the application, the Revit design software was used and the number of clicks needed to access the element linking and node troubleshooting of the technical installations was reduced. Two new functionalities have been created within the software application: "connect and copy" and "connect and rotate".

1. Introduction

Today, around the world, there are several technical installations in almost every house. Whether the industrial environment (compressed air installations, cooling water installations, technological water installations, technical gas installations, low pressure steam installations, etc.) or residential settlements (potable water installations, heating installations, sanitary facilities) are taken into account, more than 90% of the buildings are equipped with technical piping systems for cultural and administrative buildings, shopping centers, hotel buildings, hospital buildings, residential complexes and industrial buildings. The design of the technical installations for buildings (interior and exterior) must ensure the essential requirements of the norms and laws applicable in the EU countries on hygiene, health and environment, energy saving, fire safety and in-service safety.

The design department has a great importance in our campaigns, being the brain of each project, the first step that was made at the beginning of any collaboration. The success of a work begins with intelligent design and varied technical solutions. It is also known that the homogeneity and flowability of the project stages depend on correlations and details, and installation companies must therefore provide complete solutions. The problem with designing installations is to ease the designer's work and thus provide a tool for it to be able to easily make connections between the nodes (figure 1 shows some types of nodes in a system of tubing) made by drawing the pipes. Since an element within a
system (here the tubing system) has different properties such as inner / outer diameter, connection type etc., the manual search action of the required elements is very difficult.

![Figure 1. Examples of nodes in a tubing system.](image)

2. Methods
Within this work it is intended to perform automated testing of the implemented methods. It is used a static method (a method that belong to a class definition) that checks and retrieves commands returns a value of the "Result" type. In the method for the current call ("ExecuteWrappedCommand"), a "MessageScreenManager" type object is initiated that retrieves messages that are subsequently displayed for the corresponding situations.

3. The piping systems
Copy / rotate commands should normally be executed manually by the installer, but with the developed application automatically insert, the installer only needs to verify the correctness of the data. This functionality can be used in any application where automatic insertion (copying / rotation) is desired, with different input data, but it will be necessary to write the code to implement this functionality. In this case, it will be used in Stabicad, where the data will come from making calculations of elements in a CAD drawing. In order to be able to design indoor building systems, a variant proposed by Stabiplan is through an Autodesk Revit extension made internally and offered to consumers in various formats: Free, Trial or Subscription. This extension is in the form of an add-in (a tool that can be loaded by Autodesk Revit environment) and is called the Product Line Placer for Piping App, with different variants that are specially branded for different manufacturers. Through Stabicad, the engineer has advanced tools to design, visualize, manage and optimize technical and electrical installations inside building. Stabicad is an add-in that works over AutoCAD or Revit, and the user interface is a palette of commands and a ribbon. The term add-in is mainly used by Microsoft for a software or program that is added to the base program. With this application, users can graphically design their data connections.

4. Autodesk Revit software applications
Revit contains a lot of tools to plan and track different stages in the building's lifecycle, from concept to construction and later maintenance and/or demolition. Revit was designed to allow architects and other professionals in the construction field to design and document a building by creating a three-dimensional parametric model that includes both geometry and non-geometric construction information, which is also known as Modeling Building Information or BIM. The concept of bi-directional associativity between components, opinions and annotations was a distinctive feature of Revit for many versions. The ease of making changes inspired the name Revit, being an abbreviation of the Revision it. At present, the software has a double role. It is a product and, at the same time, it is the vehicle to deliver a product. In the first case, administer, acquire, modify, distribute or transmit information; In the latter case, the software acts as the basis for computer control, information communication, and the creation and control of other programs. For our project “Software instrument used as interface in the design of technical installations”, the Revit design software was used with the Stabicad platform, the decision was based on Revit's professionalism and the important benefits of
Stabicad for the vulnerabilities found in Revit. For the Revit design software, well developed for many types of users, Stabicad is an extension of standard CAD platform features and aims to deliver important functionality to optimize the design program. Another important and basic argument in this choice is that Stabicad for Revit is not just about designing installations, but also connecting symbols from drawings to article plugs and technical data. This greatly facilitates computation or interfacing with division software applications, thus leading to efficient management.

The drawing in Stabicad is done in three ways such as: drawing systems, creating diagrams and insert symbols. Stabicad's functionalities are related to the joining of elements, the extensive use of gripping points, projections and views on space-based drawing, and quoting. In addition, it has been decided to develop productivity features by adding tools that facilitate work with installations in connection, rotation, and copying of the drawing elements. The need to develop productivity-enhancing functionality comes from the need to more efficiently solve the problem of linking elements in the Revit software. The Revit software solution involves elongation of pipes and is not applicable in all cases encountered by the user. It also raises the issue of inefficiency in the actual design by changing the pipeline dimensions, which results in extra cost and inadequately occupied space in the building [1]. As a way of linking elements in the Revit software, the pipe extension is used, but it is not possible for any element in the different families. This solution also involves changing the size of the pipeline, which brings both an extra cost to the plant and a larger space occupied in the building. This necessitates the realization of the functionalities developed in this article. Stabicad is a drawing program and is an extension of the standard functions of an AutoCAD platform. Autodesk Revit is rapidly emerging as a powerful international design platform, and Stabicad complements Revit's standard functionality to ensure design efficiency and a useful end product. Stabicad supports the user flow from concept to detail, providing at any time the level of information required for consultants, installers and building managers. Scripts can be used to automate part of the Revit process; Geometry in a drawing file imported or linked directly into a Revit project has no category assigned, and therefore will not be cut. A script contains a series of AutoCAD command lines that are saved to a text file and can then be executed on CAD drawings (figure 2).

Most anything that can be typed on the command line can be copied and saved to a script. Scripts can be modified using Notepad and are saved with a .scr extension.

5. Autodesk Revit platform
Autodesk Revit is a BIM (Building Information Modeling) software platform for design and construction specialists, design engineers and architects. It addresses architects through the effective ideas they offer, from conceptual design to construction documentation. It is appointed to a structural engineer for using tools specific to intelligent structure design in coordination with other building components and for building regulations and safety. It is recommended to mechanical engineers, electricians and sanitary installers for greater precision in coordinating architectural and structural components. Autodesk Revit allows users to design buildings or structures whose components are represented in three-dimensional space, and also provides sketches of element designs in two-dimensional space. The database can be accessed with information about the structures built. At the
same time, it is important to mention the program's link with computer assisted design, called Computer-aided design, as well as the particular feature of having the tools needed to plan and track different stages of a building's life cycle, from construction to demolition [2]. In Revit's terminology, the object categories are called “families”. There are system-type families that include elements such as walls, floors, roofs, ceilings and that are built inside the project and families and components that are brought out of distinct projects for later use. An essential and defining feature of Revit software is the creation of large-scale drawings of reality by tracking the precision of the dimensions, properties, and model parameters. In this way, the users can modify components by changing predefined parameters such as height, width, position of object coordinates. In this way a family defines a geometry that is controlled by parameters, each combination of parameters can be saved in a type [3].

5.1. Contextual tab - Context sensitive ribbon
Contextual tab appears when an item is initially placed or when it is selected and displays the tools for selection in the context of that tool or item. The contextual ribbon closes when it comes out of the selected tool or when the item is no longer selected. Such a ribbon is shown in figure 3 [4].

\[ \text{Figure 3. Contextual Ribbon in Revit.} \]

5.2. The "Connect & Copy" functionality
The "Connect and Copy" functionality solves the problem of connecting and copying the item placed for its connection to other necessary connectors. This applies when the same selection is to be connected in multiple locations without having to repeat the "Connect Items" function repeatedly. For example, if pipelines need to be placed in 100 new positions, the utility of the newly developed feature will facilitate the design process by removing 99 clicks by copying the items in real time so that there is a new copy available to any new fixed position. Stopping the function is done with the "Escape" key. This is represented in figures 4 and 5. The main cases are given by the initial move of the selection, if it has all the free connectors, instead of its copying so that for the first fixed connector the functionality of the "Connecting elements" command is preserved. If the selection contains at least one already linked connector, it will require the user to select the mobile connector, and then select the fixed connector, and then connect the elements.

\[ \text{Figure 4. The initial stage before using the "Connect and Copy".} \quad \text{Figure 5. Final status after using the "Connect & Copy".} \]

The "CmdConnectAndCopy" class architecture purpose is to copy and connect the items to a new location. The class inherited from the base class "CmdConnectBase" and from the
"IExternalCommand" interface. Instances of an object that filters user selection are instantiated. The selected item must meet one or more of the following conditions: it must fit into a particular category; it must belong in the given category to the preselected elements; The connectors of the selection must meet the required criteria.

5.3. The "Connect and Rotate" functionality
The “Connect and Rotate” functionality has developed into the contextual ribbon and solves the problem of copying the elements and then rotating them around the axis of the fixed connector. This involves linking a selection and rotating it right after the connection. It is useful if the user wants to connect and then rotate the items in a shorter time and with fewer clicks, thus achieving the goal of increasing productivity in the plant design process. The addition of functionality has been suggested by users at the request of Stabicad representatives to meet their needs and efficiently fulfill them in order to efficiently optimize the Stabicad platform. The user selects the objects, displays the contextual tab, accesses the copy and rotate command by pressing the “Connect and rotate” button, the user selects the mobile connector (if the selection contains one free connector it will be considered by default as the mobile connector and will skip the request stage) and then the fixed one. The elements will be linked and the dialogue will appear in which the user will choose the angle of rotation. The function is illustrated in figures 6 and 7.

![Figure 6. Initial position for “Connect and rotate”](image1)
![Figure 7. Selection copied and rotated via the “Connect and Rotate”](image2)

Thus, the functionality comprises two representative stages: connection and rotation. To make the connection, it is necessary to study several possible situations. Retrieving the connection and rotation command is performed from the command class. It is noted that it is inherited from the basic abstract class for connection and rotation (referred to as the “CmdConnectAndrotateBase” code) which in turn inherited from the base class “CmdConnectBase” whose functions were studied in the presentation of the connection and copying functionality. Thus, the connection and rotation of the elements is based on the connection algorithm of a selection, plus the rotation. The base class with common functionality for login, connect and rotate tools inherited from the base class responsible for the connection and from the “IexternalCommand” interface in the “Autodesk.Revit.UI” namespace.

The class builder is set by the Boolean variable “rotateConnectedElements” if the selection is only to be connected or if it is to be rotated. Also, the execution command is taken through an integer variable. Many problems in software systems can be traced back to missing or incorrect requirements [5]. This makes the difference between the “Connect elements” and the “Connect & rotate elements” button. These are shown in figure 8.

![Figure 8. The button for connecting the elements and the connection and rotation respectively](image3)
5.4. Taking the commands

Keyword "enum" is used in C# to declare an enumeration and represents a distinct type that contains a set of constant value names, also called an enumeration list. In the same way, “Result” enumeration in the "Autodesk.Revit namespace UI" contains the set of values for the result that a function can perform: failure (with integer "-1"), success (with full value "0") and waiver (full value "1"). Thus, the static method that checks and retrieves commands returns a value of the "Result" type. Its input parameters are a "ExternalCommandData" external command in Revit API, a message, a set of elements, a command identifier, and a String parameter vector. Initially, the current work directory is set on the path of the Stabicad bin folder (Win32 or x64). This is necessary because Revit changes this path when a new project opens and when a command is executed Stabicad no longer locates the creation file used to resolve the dynamic library on CLSID (a "Class Identifier" is a unique global identifier that identifies a COM class object). If the server or container allows binding to embedded objects, a CLSID must be recorded for each supported object class.) Set the current project file name to Stabicad. Checks the command, and if it matches it creates an instance of the appropriate class and returns the function that executes the ribbon command. Figure 9 highlights the command checks and entry on the "if" branch for the connection and rotation command in the debug execution of the program.

```csharp
} else if (commandID == "RV_BTN_CONNECT INTO") {
    CmdConnectElements command = new CmdConnectElements();
    return command.Execute(revit, ref message, elements);
} else if (commandID == "RV_BTN_CONNECT_COPY") {
    CmdConnectAndCopy command = new CmdConnectAndCopy();
    return command.Execute(revit, ref message, elements);
} else if (commandID == "RV_BTN_CONNECT_AND_ROTATE") {
    CmdConnectAndRotate command = new CmdConnectAndRotate();
    return command.Execute(revit, ref message, elements);
} else if (commandID == "RV_BTN_ROTATE") {
    CmdRotateElements command = new CmdRotateElements();
    return command.Execute(revit, ref message, elements);
```

**Figure 9.** Execute the "Connect and rotate" command from the ribbon.

6. Results

The work aims to increase productivity by improving the design of building installations. This was done by adding the contextual ribbon together with already existing functionality, thus optimizing working time by reducing the number of clicks required to access element linking and node solving functions. Also, two new functionalities have been created: "connect and copy" and "connect and rotate". Particular cases related to the number of free connectors of the component or set of elements selected to be coupled and subsequently copied or rotated have been analyzed. In the case of connecting and copying the elements, if the selection is free (with all the unoccupied connectors), then it is initially carried out to move it, respecting the model of the element's connection functionality. One copy is made per transaction, and pressing the "Escape" key allows you to exit. When fixed components have only one available connector, the user is not required to select it, thus omitting an intermediate step in the process. If the connection and rotation functionality is called, the selected item will disconnect from any adjacent components and connect it via the fixed connector. A dialog box containing rotation information is displayed, namely direction and angle of rotation. It has conditioned the selection of the fixed position so that it does not coincide with the movable position.

Conclusions

Automatic testing was performed through integration and regression tests. This is to verify that the previously developed and tested software continues to function properly after it has been modified or interfaced with other software applications.
References

[1] Autodesk Revit development resources, www.autodesk.com/developrevit
[2] Tammik J 2019 Revid Visual Studio, https://thebuildingcoder.typepad.com/blog/2018/09/revit-2019-visual-studio-net-add-in-wizards.html, accessed 09.01.2019
[3] Faraz R 2014 C# School, ed Tore Nestenius, Jonathan Worthington, Lee Addy, First edition
[4] Autodesk, Inc. Revit Architecture 2011 User's Guide
[5] *** 2014 Software testing, ed Hamling B, Edition, BCS Learning & Developmented