

# An Example of Application Design Using SolidWorks Application Programming Interface

Andrijana Bocevska<sup>1</sup>, Stefan Krsteski<sup>2</sup> and Snezana Savoska<sup>3</sup>

<sup>1,3</sup> Faculty of Information and Communication Technologies,  
St. Kliment Ohridski University, ul. Partizanska bb, 7000 Bitola, North Macedonia  
{<sup>1</sup>andrijana.bocevska, <sup>3</sup>snezana.savoska}@uklo.edu.mk  
<sup>2</sup>krsteski.stefan@gmail.com

**Abstract.** In this paper a developed software application for car rim design and its CAD models updating by automating repetitive tasks using SolidWorks application programming interface (API) are presented. An application is developed using Microsoft Visual Studio and C# programming language. Such developed application is integrated with SolidWorks CAD package through application programming interface (API). Developed application consists of front end and back end. Front-end implements GUI (Graphical User Interface) through which type of models, number of rays and number of elements in one ray are taken as input from user. Back end is represented by the internal program, which obtains a car rim design as output. Output design result is passed to SolidWorks CAD package, which updates CAD models of car rim and manufacturing drawing.

**Keywords:** application programming interface (API), Microsoft Visual Studio, C# programming language.

## 1. Introduction

Computer-Aided Design (CAD) refers to the method of using computer software to create virtual models of proposed products. There is a variety of CAD software available, each with special features suitable for a particular application. The design of a product must be ready down to the last detail before putting it into production. This ensures that no setbacks through necessary changes to the product can occur afterwards. Thus, a CAD system is much more than a replacement for manual drawings. It brings a completely new level of possibilities that engineers can utilize for faster creation of better products with less mistakes.

CAD systems provide interfaces via Application Programming Interface (API), which enables the users to interact with the model environment with the help of Macros. Macro stands for rule or pattern that specifies how a certain input sequence should be mapped to an output sequence according to defined procedure. The interaction with the model environment is important to improve

the design automation process. The most common advantages from the use of the API present the increasing need for sharing data among various disciplines in order to find solutions from problems. Sometimes there is also a need to implement specific algorithm to perform computation, which cannot be found in commercial software. If software without API is applied, the user can only access to single model entity and the direct access to internal database is not permitted [1]. Using API database of entity a direct access is provided, which leads to saving time and interlace of model entities with math and user defined ones [2].

SolidWorks is a 3D mechanical CAD program that runs on Microsoft Windows. SolidWorks is a parasolid-based solid modeler and utilizes a parametric feature-based approach to create models and assemblies. Complete and intuitive 3D solutions for product design, analysis, and data management are provided. It is easy to learn, easy to use and easy to navigate, letting users concentrate on their designs. Using SolidWorks, the user is able to quickly plot a sketch, execute feature modeling and dimensioning. SolidWorks files use the Microsoft Structured Storage file format. In other words, there are various files embedded within each SLDDRW (drawing files), SLDPRT (part files), SLDASM (assembly files) file, including preview bitmaps and metadata sub-files.

This paper introduces the advantages of the applicable programming interface of CAD system in creating custom programs (macros) that can assist users in automating many of the tasks with the help of simple graphical user interface (GUI). This contributes to reduce design time and eliminate any potential errors.

For developing software application, Microsoft Visual Studio is used. Integration of Microsoft Visual Studio with CAD package resulted in exchanging data between these two software packages. There are many CAD software packages having Application Programming Interface functionality. In this paper SolidWorks CAD package is used, because of its user-friendly features and the support of Application Programming Interface functionality.

The paper is organized as follows. Related works are described in Section 2. Explanation about API in SolidWorks is depicted in the next section. A common way to get to know the API and automate operations, which repeat constantly, is the Macro Recorder. Reasons to write a macro are provided in Section 4. Section 5 provides screenshots of the developed application. Last section gives a brief review of the research, providing concluding remarks and directions for further work.

## **2. Related Works**

Many researchers have used the API of various CAD systems in order to develop applications and tools, and always the conclusions represent that the benefits and advantages of the API are huge. Some researchers [2] present an application of

Visual Basic Application Programming Interface (API) to develop numerical and procedural algorithm into CAD software. The presented example of API and SolidWorks interaction is about the acquisition and processing of surfaces acquired by 3d laser scanner. Paper [3] presents another use of the API regarding the manufacturing technologies that are designed for the assembly of a single cylinder, four strokes, and air-cooled engine block. An application that has been developed with the API of a general purposes CAD system is presented by [4]. This application automates the design process of a standard pneumatic double acting cylinder based on the appropriate inserted parameters (ISO 15552). The light tie rod hydraulic cylinder design process is studied by [5]. The study is based on detailed analysis of the structure design, VB programming environment with the usage of computer and SolidWorks as platform. Developed software application for product design and its CAD model updating by automating repetitive tasks using SolidWorks application programming interface (API) can be found in [6]. The authors implemented a case study of winding machine, which is a mechanical product for which a program using Visual basic language i.e. Vb.Net Application is developed and integrated with SolidWorks CAD package through application programming interface (API). The work done by [7] simplifies task regarding hundreds of CAD models needed to be open and verified using Commercial CAD software's API (Application Programming Interface).

### **3. Advantages of SolidWorks API**

The SolidWorks Application Programming Interface, or API, is used to automate and customize the SolidWorks software. Using API can help you automate design processes, reduce design time, and eliminate any potential errors.

The SolidWorks Application Programming Interface (API) is a COM programming interface to the SOLIDWORKS software. The API contains hundreds of functions that can call from Visual Basic (VB), Visual Basic for Applications (VBA), VB.Net, C++, C#, or SolidWorks macro files. These functions provide the programmer with direct access to the SolidWorks functionality, which implies that you can automate SolidWorks to carry out specific functions or tasks.

Recent SolidWorks developments have improved the method supported by native object and they have been interlaced with very powerful mathematical utility. The following three kinds of objects can be manipulated when using API into SolidWorks [2]:

- SolidWorks (model native entities),
- Math utility database (math entities) and
- User defined entities

The native geometrical objects concern the sketch entities (point, line, circle, spline, etc.) and their constraints, the features (extrusion, revolution, loft, etc.),

the assembly management (mating, inserting, moving, etc.), whereby the mathematical objects concern points, vectors and transformations for manipulation of entities (projecting from model space to sketch space and vice versa, performing basic operation on vectors, etc.).

Performed operations are recorded with the SolidWorks user interface and can be replayed using SolidWorks macros. A macro contains calls to the Application Programming Interface (API) that are equivalent to operations performed in the user interface. A macro can record mouse clicks, menu choices and keystrokes. The macro toolbar contains shortcuts to the macro recording commands, which can also be accessed from the Tools→Macro. The automation of specific operations using macro recorder especially makes sense if operations repeat constantly. Macro recorder also helps to get to know the commands and objects, when program is made outside of the CAD software tool over the API. Advantages of using macro include: tasks automation, enablement of design optimization, improvement of visualizations of CAD designs, retrieval or export of data that already exist in the CAD solid modeler's document, increase of productivity, time and money saving, procedures standardization.

#### **4. Screenshots of the developed applications**

In this paper an application using SolidWorks API is developed. Microsoft Visual Studio is implemented as an integrated development environment. Microsoft Visual Studio can be used to develop desktop and mobile applications, web pages and web applications, as well as web services. It incorporates an advanced code editor, compiler, linker, and graphical tools for creating a user interface. Visual Studio can be programmed in a number of programming languages, including: C, C++, VB.NET, C# and F#. The applied programming language in this paper is C#. C# is a powerful programming language that offers support for many types of programming such as imperative, declarative, functional, generic and object-oriented programming.

In Visual Studio, an automatic generated code is located in the code window when the option to create a Windows Forms project is chosen. This code is automatically generated by Visual Studio. It adds the necessary references to the libraries to create the form.

Next, references to SolidWorks should be added. For this purpose, Add Reference and tab COM are selected. The necessary libraries are chosen from the list. Once this step is done, the full structure of the C# program exists, after which typing the code for SolidWorks can be started.

Macro Record is a built-in feature in SolidWorks that allows us to record macro directly from SolidWorks. To do this, SolidWorks is launched, and

then Macro Record from the toolbar is selected. Macro Stop sends an order to SolidWorks to stop recording the stock and save the recorded macro.

The purpose of the application is to automate the design process of a car rim. Fig. 1 shows the user interface of the application.

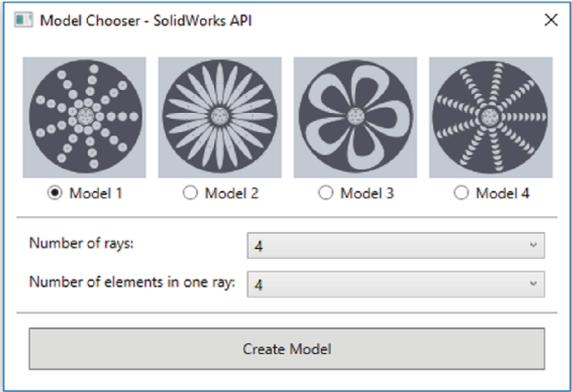


Figure 1. User interface of the application.

For the graphic interface, a group of four radio buttons is created and can be used in order to choose model of the car rim. Subsequently two drop-lists for the number of rays and the number of elements in one ray are created. A Create Model button at the bottom of the interface in SolidWorks is also added. It allows the user to start the program by executing the model creation algorithm.

The user can select one of four offered models. Design of car rim of Model 1 is shown in Fig. 2.

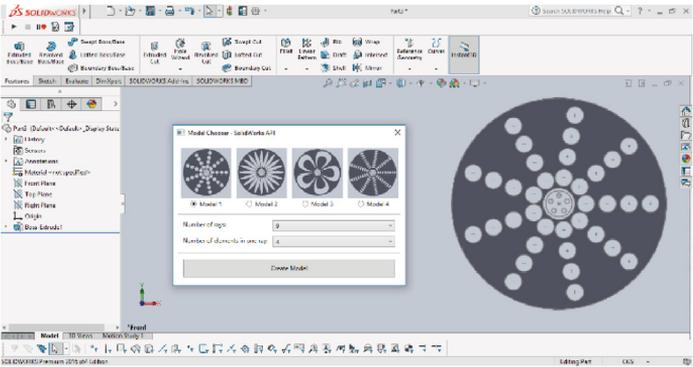


Figure 2. Creating Model 1 in SolidWorks.

When the user chooses the required parameters and selects the Create Model button, the code calls a function that checks which of the models is selected and depending on the model, the corresponding function for its creation in SolidWorks is called, as shown in Fig. 3.

```
private void button_Click(object sender, RoutedEventArgs e)
{
    if ((bool)rbModel01.IsChecked)
        RimModel01();
    else if ((bool)rbModel02.IsChecked)
        RimModel02();
    else if ((bool)rbModel03.IsChecked)
        RimModel03();
    else if ((bool)rbModel04.IsChecked)
        RimModel04();
}
```

**Figure 3.** Functions that check selected model.

If the user has selected the Model 1, then the RimModel01 function is called. In that function, we first take the other parameters, i.e. the number of rays and the number of elements in one ray, as it is shown in Fig. 4.

```
int numOfRays = int.Parse(comboBoxNumberOfRays.Text);
int numOfElements = int.Parse(comboBoxNumberOfElements.Text);
```

**Figure 4.** Parameters while RimModel01 function is called.

If an error occurs (for example, SolidWorks has not started yet), the user will be informed by a message with an explanation, Fig. 5.

```
Sldworks.Sldworks swApp = new Sldworks.Sldworks();
swApp.Visible = true;

ModelDoc2 swDoc = null;
PartDoc swPart = null;
DrawingDoc swDrawing = null;
AssemblyDoc swAssembly = null;
bool boolstatus = false;
int longstatus = 0;
int longwarnings = 0;
swDoc = ((ModelDoc2)(swApp.ActiveDoc));
if (swDoc == null)
{
    MessageBox.Show("Please open a new part in SolidWorks");
    return;
}
```

**Figure 5.** Message if an error occurs.

If all goes well, SolidWorks selects the front plane and creates a drawing. Then one element of the ray is drawn and a linear pattern is made, i.e. one ray is

created with the corresponding number of elements that was previously chosen by the user, Fig. 6.

```
skSegment = ((SketchSegment)(swDoc.SketchManager.CreateCircle(0.0503, 0, 0, 0.060421, -0.012805, 0)));
boolstatus =
swDoc.SketchManager.CreateLinearSketchStepAndRepeat(numOfElements, 1, 0.04, 0.01, 0, 1.5707963267949, "", false, false, false, true, false);
```

**Figure 6.** Creating one element of the ray and linear pattern.

Then it makes a circular pattern of all elements in the ray around the center of the car rim, Fig. 7.

```
boolstatus =
swDoc.SketchManager.CreateCircularSketchStepAndRepeat(arcRadius, 3.1415993717028678, numOfRays, 1.5707963267949, true, "", false, false, true);
```

**Figure 7.** Creating circular pattern.

Finally, Extrude feature is made in order to create a 3D model, Fig. 8.

```
myFeature = ((Feature)(swDoc.FeatureManager.FeatureExtrusion2(true, false, false, 0, 0, 0.02, 0.01, false, false, false, false, 0.017453292519943334, 0.017453292519943334, false, false, false, false, true, true, true, 0, 0, false)));
```

**Figure 8.** Creating 3D model.

## Concluding Remarks

This paper represents the advantages of the SolidWorks Macro/API (Application Programming Interface) that allows users to automate the designing tasks and eliminate the conventional and repetitive process of designing for different sizes and conditions. With the use of the SolidWorks API, it is possible to develop applications in short time that can be used to analyze various engineering situations. The main advantage of such applications is that there is no need of any physical model in order to simulate and analyze any engineering processes.

API functions, in this paper are implemented in order to achieve the car rim design in SolidWorks environment, which not only simplify the design process and improve the design efficiency, but also represent foundation for further strength analysis, motion simulation, and finite element analysis of the

car rim. Time required for overall design process is also reduced hence significant amount of saving in cost arises. SolidWorks CAD package is used, because of its user-friendly features and the support of Application Programming Interface functionality. Microsoft Visual Studio is integrated with SolidWorks due to the possibility of exchanging data between these two software packages.

The front end of the developed application implements GUI (Graphical User Interface) through which type of models, number of rays and number of elements in one ray are taken as input from user. An internal program represents the back end by means of which a car rim design is obtained as output.

This kind of applications have a wide applications prospect in the field of mechanical design. Using mentioned technologies applications with other designs could be developed. There is no need for the user to have programming knowledge or CAD modelling skills to use this application.

Directions for further work are to upgrade the existing algorithm, so that the user can select the material, color and texture of the car rim, which would significantly improve this application.

## References

- [1] Pezzuti E., Piscopo G., Ubertini A., Valentini P.P., Milana M., Di Leginio R., Una metodologia per l'analisi e l'archiviazione di reperti archeologici basata sul rilievo mediante scanner laser tridimensionali a non contatto, *Archiviazione e Restauro di Reperti Archeologici Mediante Tecniche CAD-RP* (in italian), Napoli, 2004.
- [2] Gattamelata D., Pezzuti E., Valentini Pier Paolo, Using application programming interface to integrate reverse engineering methodologies into SolidWorks, In: XVIII Congreso Internacional de Ingeniería Gráfica de INGEGRAF, 2006.
- [3] Haba S.A, Oancea G., *Int. J. Adv. Manuf. Technol.* 80,747 (2015)
- [4] Tzotzis A., Hernandez C. G., Huertas-Talon J. L, Tzetzis D., Kyratsis P., Engineering applications using CAD based application programming interface, *MATEC Web Conf.* Vol. 94, 2017, <https://doi.org/10.1051/mateconf/20179401011>
- [5] Hongyan W., Guohui H., Xuemei J., Hong W., The Parametric Design for Hydraulic Cylinder Based on SolidWorks, *Applied Mechanics and Materials Online: 2013-08-30* ISSN: 1662-7482, Vols. 380-384, pp 132-135, Trans Tech Publications, Switzerland, 2013 doi:10.4028/www.scientific.net/AMM.380-384.132. © 2013
- [6] Abhishek C. L., Rao A.S., Design and Drawing Automation Using Solid Works Application Programming Interface, *International Journal of Emerging Engineering Research and Technology*, Volume 2, Issue 7, October 2014, PP 157-167 ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)
- [7] Sung-Hwan J., Assessment of Three Dimensional CAD Models Using CAD Application Programming Interface, *ASME International Mechanical Engineering Congress and Exposition, Pittsburgh, Pennsylvania, USA, November 9–15, 2018, Volume 5: Engineering Education*, ISBN: 978-0-7918-5206-4