

An Example of Application for Custom Design Automation using SolidWorks Application Programming Interface

Hristijan Stojceski¹, Andrijana Bocevaska², Igor Nedelkovski³ and Nikola Rendevski⁴

^{1,2,3,4} *University St Kliment Ohridski, 1 Maj nn, 7000 Bitola, R. North Macedonia*

*hristijan.stojceski@uklo.edu.mk; andrijana.bocevaska@uklo.edu.mk; igor.nedelkovski@uklo.edu.mk
nikola.rendevski@uklo.edu.mk*

Abstract:

In this paper an application which allows users with minimal CAD knowledge to interact with the software and make custom design automation is presented. The user-interface was developed using the Visual Basic for applications (VBA) programming language and the CAD software's built in application programming interface (API) command structure. This process allows the automation of selected embedded CAD productivity tools whereby the user is able to modify specific parameters to manipulate the shape of their design and automatically generate a 3D computer-aided model reflecting their specific modifications. This paper utilizes a solid body of four types of tool designs as an example. We describe the creation of the user interface, as well as specific limitation ranges and/or constraints that were placed on the parameters of the designs. The result was the development of an interactive design template that prevents design engineers from needing to design the same model's family repeatedly to suit the needs of the customer, thereby reducing the design time and mistakes while enhancing the design consistency.

Keywords:

Custom design automation, SolidWorks API, VBA

1. Introduction

With today's emerging markets and product variety, it is very important for industrial companies to explore product customization to capture customer attention and deliver true customer value. Product customization refers to enabling customers to personalize a product according to their needs and preferences [1]. Add-ons, exclusive functionalities, templates, and flexibility with product design all count as different forms of personalization.

Product customization is essential because it allows customers to feel like a part of the product development process. They can choose the features they want, the color they want, and even the style of the product. In addition, product customization helps businesses learn more about their customers. By understanding what customers want and need, businesses can create products that are more likely to sell. Customization also helps companies target specific markets.

A few of the benefits that businesses can enjoy when they offer product customization [2]:

Because Customers Like It - Product customization adds a personal touch to the product. When customers design it their way, the end product does not just remain a product but a feeling gets attached to it no matter whether they are doing it for their own or for someone else. Since customers like the personalization concept, it provides brands with an excellent opportunity to excel in the e-commerce market.

Product Customization builds Customer Loyalty - Customer loyalty is one of the significant benefits of product personalization. And there's no better way to earn a customer's loyalty than to give them complete control over the product's design. Customer success is the most crucial factor in gaining that loyalty [3]. Product customization creates a bond between customers and brands and significantly increases customer retention [4].

Higher Sales - Customers are willing to spend more for a personalized item because they regard it as more than simply a product; it is something unique to them. This can be backed up with a Deloitte

study claiming that one out of every five consumers would be willing to spend 20% more on an exclusive product [5]. Also, as per Invesp, 59% of the marketers are getting a good ROI after offering the product personalization facility [6].

Word-of-Mouth Marketing - According to the Harvard Business Review, buyers who can customize their products buy more and are more inclined to suggest the brand to their friends [7]. People tend to notice something that stands out. If customers are loyal and satisfied with the products, they'll probably tell others about them, resulting in more people buying them.

Get Better Insights into your Customers - Adding to the benefits of product customization is that a personalized purchase provides a more detailed analysis of your customer's preferences, likings, and taste than a normal purchase. When customers purchase customized products, brands can collect that information and utilize it to offer products based on their previous preferences. Efficient research and development lead to a competitive advantage [8].

Product Customization module by KnowBand - Knowband offers a Product Customization module for offering customization services to the customers. It adds a customization tab on the product pages and allows the customers to personalize a product as per their desire. From adding the desired images to putting up a customized text and QR code, the Product Designer module comes with a variety of features to help customers personalize the products exactly the way they want.

Customer involvement in new product development is currently a thriving activity implemented by companies in order to fulfill customers' needs [9].

In modern conditions, it is important for any company to reduce the time required to design new products. One way to do this is through the automation of design processes. Automation in SolidWorks is a huge opportunity to speed up and make design process dynamic. Custom applications using the SolidWorks API can be as simple or as complex as desired, for instance, the API can be used to create simple macros.

Therefore, the purpose of this paper was to create an application for producing customized products for customers with poor or no skill in CAD software. This paper focused on developing an interface for the customers to use to modify product appearance by integrating several software applications such as Visual Basic, SolidWorks and Excel. Visual Basic creates the user-interface form and integrates SolidWorks and Excel together using each software's API command structure, whereas SolidWorks for 3D solid model creation and Microsoft Excel are used for customer's parameter values. SolidWorks Macro/API allows the user to automate the designing tasks and eliminate the conventional and repetitive process of designing for different sizes and conditions.

The paper is organized as follows. Previous work is described in Section 2, whereas Section 3 discusses the features of the used technologies. Description and screenshots of the developed application are given in Section 4. Last section gives a brief review of the research, providing concluding remarks and directions for further work.

2. Previous work

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.

An alternative methodology for real time object customization in a CAD system is suggested in [10]. The proposed program modifies the CAD model through Visual Basic programming and operates by controlling and manipulating the instructions. It realizes a real time customization where it allows users to adjust and change the parameters on the GUI instead of editing the model manually, thereby simultaneously modifying the shapes in the CAD interface.

The author of [11] elaborates how a complete design automation procedure can be made based on examples. The paper considers the exponential growth in design methodologies and depicts works on design automation of a cube on certain parameters. SolidWorks designing and modelling was used for writing the macro code for automation. Parameters like dimensions, colors and materials have been considered for automation. After the conceptual development and algorithm build up, an application has been made in order to make the product user-friendly.

The work presented in [12] uses Visual Basic to develop corrugated box parametric drawing system on SolidWorks platform on the basis of the established corrugated box access database. The users only

need to select the desired corrugated box carton and the key parameters, then can get the 2D expansion diagrams and 3D stereoscopic diagrams in accordance with the national standard, which significantly improve the efficiency of the corrugated box drawing.

A procedure to automate the design of metal scrap balers by integrating Visual Basic, Excel and SolidWorks is presented in [13]. Firstly, a prototype model of the mechanical system under development was designed in SolidWorks by GUI. The geometric dimensions considered for the iterative work were then entered to an Excel file. The data in the Excel file can be edited and changed for the iterative work. The modifying program developed in Visual Basic reads the data from the Excel file and automatically modifies the SolidWorks model using API. The modified model was analyzed and the analysis results were evaluated.

The authors of [14] investigate how design automation can be used and implemented to automate and improve sections of the order-to-delivery process for customized products at an industrial manufacturing company. Their objective was to develop a product configurator to automatically generate 3D models and documentation for production and sales support. This was achieved through the development of a product configurator which decreases ODP lead-time, prevents errors, reduces material waste and adds customer value.

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) are outlined in [15]. This procedure contributes to reduce design time and eliminate any potential errors. An application was developed using Microsoft Visual Studio and C# programming language and was integrated with SolidWorks CAD package through API. Output design result is passed to SolidWorks CAD package, which updates CAD models of car rim and manufacturing drawing.

3. Features of the used technologies

Key element for creating the application was a thorough understanding of the CAD tools available in SolidWorks to generate multiple configurations. SolidWorks allows the development of part configurations through the use of a design table. A design table uses an Excel spreadsheet to allow a user to enter part feature information to create a new configuration of a product by changing selected part dimensions or by suppressing part features. The Excel spreadsheet can be automatically linked to the SolidWorks software, so when a user edits the data in Excel spreadsheet externally, SolidWorks will create the new model or models based on the entered data. Each software uses a common programming structure called an application programming interface (API) which enables software users to create tools using Visual Basic for applications (VBA) programming language to interact with the software. This interaction could be accomplished using two basic approaches depending on the type of tool being developed and the type of user interaction desired. Those approaches included: VBA, Add-Ins (DLL or EXE), and standalone exe.

There are two ways to make design changes: customize a model from within SolidWorks or customize the model outside of the SolidWorks software. Customizing the model from within the CAD software wouldn't be very helpful since the user would need some level of functional knowledge of the CAD software. For those reasons the standalone exe. application was created as independent program that is capable to control other software and has its own interface. In this regard, Visual Basic software was used for the standalone exe. to interactively work within the SolidWorks and Excel software to customize a model. Once the user has submitted parameter values into the user-interface form, the parameter values will automatically be entered in Microsoft Excel, which in turn updates the SolidWorks 3D model automatically, and the result will be displayed back to the user-interface form for the customer to compare. The role for Visual Basic was to be able to control Microsoft Excel and SolidWorks using their API commands to work internally as if it were part of Visual Basic codes. To automatically enter the customer's parameter value from the user-interface form into the Excel, Visual Basic must use Excel API commands. Once the model is customized, the picture box will show the updated model with the current values of the customized model. All the textboxes in the user-interface form are protected with error checking codes to make sure that the customer cannot enter any invalid input. The customers are only allowed to enter numeric points into the textbox parametric values. In

addition, the application creates an automatic drawing of the model if the checkbox for creating drawing is selected.

4. Description of the developed application

In the paper, an application is developed using SolidWorks API and VBA programming language. SolidWorks provides an API that allows external programs to interact with and control SolidWorks functionality. The API exposes a set of objects, methods, and properties that developers can use to access and manipulate SolidWorks models, assemblies, drawings, and other elements.

VBA is event driven language, embedded in SolidWorks that enables users to write custom macros and automation scripts. These VBA macros can be used to create custom features, automate repetitive tasks, and enhance the capabilities of SolidWorks.

With the help of VBA, users can create macros that automate various aspects of the design process in SolidWorks. For instance, you can write a VBA macro to generate complex geometric shapes, perform design validations, or automate the creation of drawings and reports. VBA also allows users to customize the SolidWorks user interface by adding custom toolbars, menus, and buttons. This level of customization enables users to have a more tailored and efficient workflow based on their specific needs. Lastly, through VBA, SolidWorks can communicate and exchange data with other Microsoft Office applications, such as Excel and Access. This integration allows for seamless data transfer between the design process and other parts of the workflow.

SolidWorks includes a built-in feature called "Macro Record", which allows users to record macros directly within the software. To use this feature, simply launch SolidWorks and select "Macro Record" from the toolbar. Once the recording is complete, you can stop the macro recording by clicking on "Macro Stop". The recorded macro will then be saved.

The application's primary objective is custom design automation of tools, such as spanners. By leveraging the recorded macros, users can streamline repetitive tasks and efficiently generate designs for various tools, enhancing productivity and reducing manual effort. Figure 1 shows the user interface of the application.

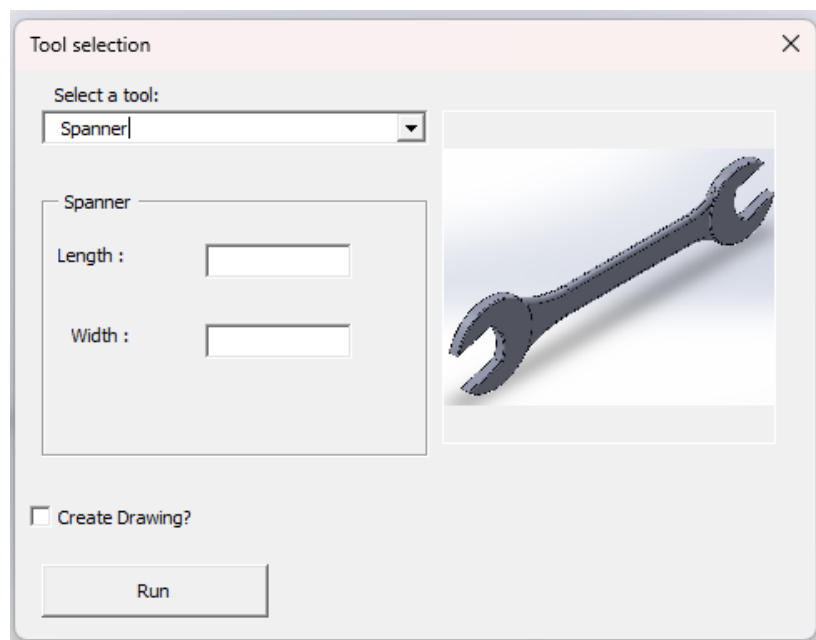


Figure 1: User interface of the application

The proposed graphic interface consists of a drop-down list, two textboxes, a checkbox and a "Run" button. A preview of the model is shown on the right side of the interface. Using the drop-down list, one of the following four tools can be chosen:

- Spanner

- Screwdriver
- Allen key
- Adjustable wrench

By default, spanner is selected, and the tool has couple of parameters that can be modified. Users can input desired tool dimensions through the textboxes, which will be utilized in the modelling process. By clicking the "Run" button, an algorithm executes, generating the tool based on the provided dimensions.

The interface's checkbox triggers the automatic creation of drawings for the newly designed tools. Users can opt to generate these drawings during the modelling process, providing comprehensive documentation for the models. To facilitate future reference and data analysis, an algorithm is implemented to store all tool dimensions in a Microsoft Excel spreadsheet. This feature ensures easy access to and review of the design specifications of previously created tools. The user can choose one out of four models which are available, as shown in Figure 2.

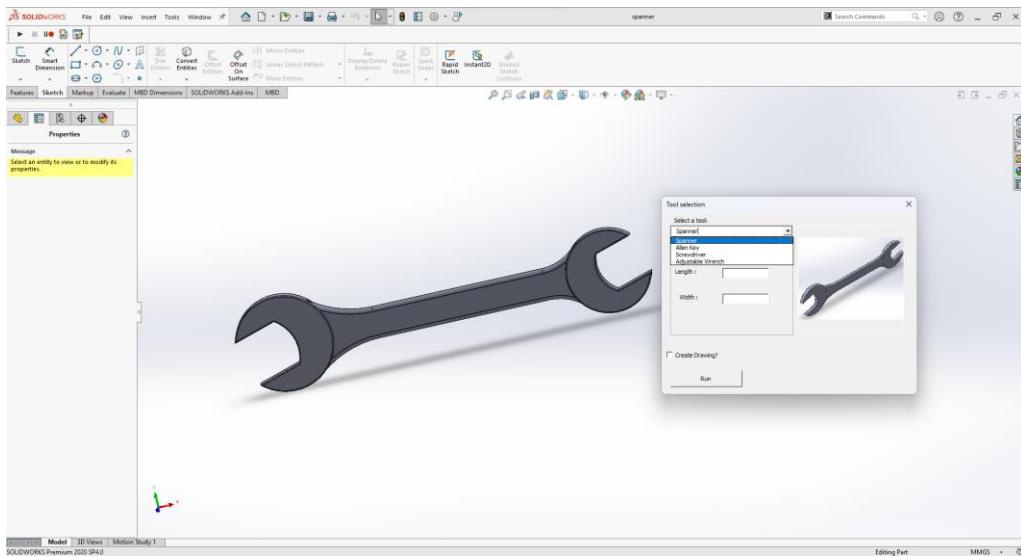


Figure 2: Available models

For each tool there are textboxes available which must be utilized before clicking on the "Run" button. Each textbox has validations for the minimal and maximal integer number for the selected tool - this prevents the user of creating a miniature spanner or an absurdly big one. If the user enters a faulty value, they will be warned by the system and informed what needs to be changed. They will also be provided with the min/max values or that particular dimension, as illustrated in Figure 3. The code for the spanner validation is shown in Figure 4.

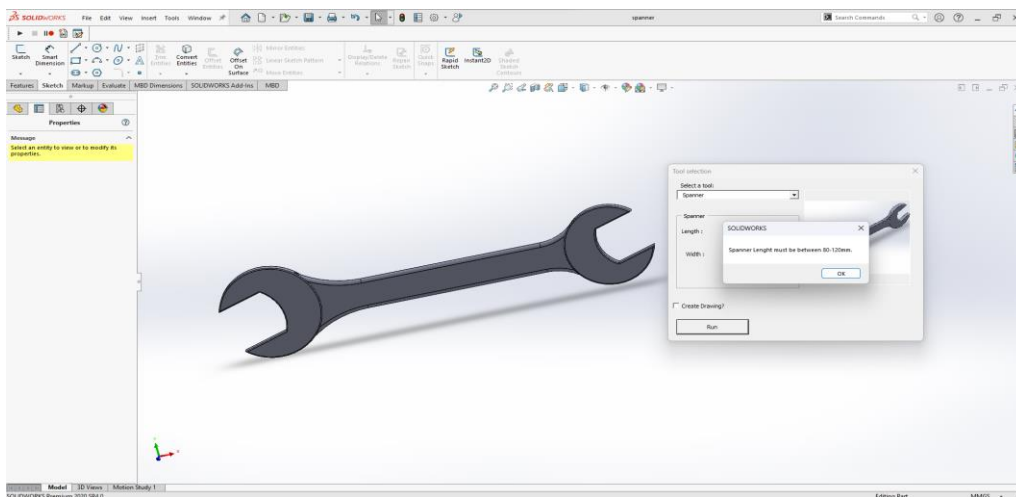


Figure 3: Validation for Spanner length

```
'Spanner Controls
If txtSpannerLenght.Value < 80 Or txtSpannerLenght.Value > 120 Then
MsgBox "Spanner Length must be between 80-120mm."
ControlValues = False
Exit Function
End If
```

Figure 4: Code for Spanner validation

After successful insertion of the values, the system will adjust the models as per requirement and it will call the function that writes the dimensions in a separate sheet for each model in an external Excel file, as given in Figure 5 and Figure 6.

```
Sub WriteToExcelFile()
Dim strDrawStatus As String

    If chkUpdateDrawing.Value = True Then
        strDrawStatus = "YES"
    Else
        strDrawStatus = "NO"
    End If

    Dim xlApp As Object
    Dim xlWorkbook As Object
    Dim xlWorksheet As Object

    Set xlApp = CreateObject("Excel.Application")
    Set xlWorkbook = xlApp.Workbooks.Open(excelPath)
    Dim lastRow As Long
    Select Case cmbModels.Text
    Case "Spanner":
        Set xlWorksheet = xlWorkbook.Sheets(1)

        lastRow = xlWorksheet.Cells(xlWorksheet.Rows.Count, "A").End(xlUp).Row + 1
        xlWorksheet.Cells(lastRow, "A").Value = spannerLenght
        xlWorksheet.Cells(lastRow, "B").Value = spannerWidth
        xlWorksheet.Cells(lastRow, "C").Value = strDrawStatus
```

Figure 5: Code for writing to external Excel file

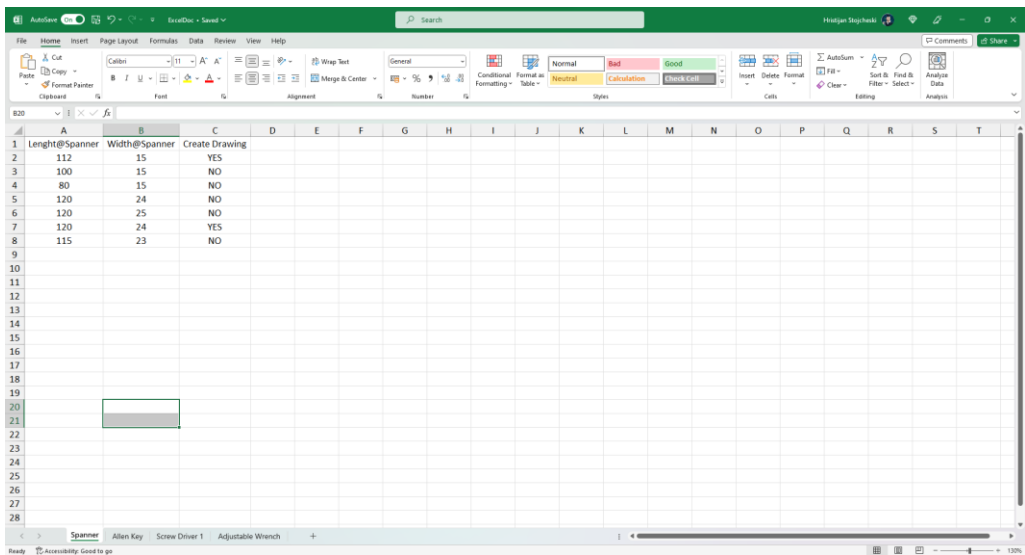


Figure 6: Excel data

Moreover, if the checkbox for creating drawing is selected by the user, the algorithm will create an automatic drawing of the model with the newly entered values by calling the method in Figure 7 in VBA. The created drawings of the spanner are illustrated in Figure 8.

```
Sub OpenAndUpdateDrawings ()  
Dim swDrawingModel As Modeldoc2  
Dim drawingName As String  
drawingName = folderPath & cmbModels.Text & "\" & cmbModels.Text & ".slddrw"  
Set swDrawingModel = swApp.OpenDoc6(drawingName, swDocDRAWING, swOpenDocOptions_Silent, "", 0, 0)  
swApp.ActivateDoc3 swDrawingModel.GetTitle, False, 0, 0  
swDrawingModel.ForceRebuild3 True  
'swApp.CloseDoc swDrawingModel.GetTitle  
Set swDrawingModel = Nothing  
End Sub
```

Figure 7: Code for creating drawing for a model

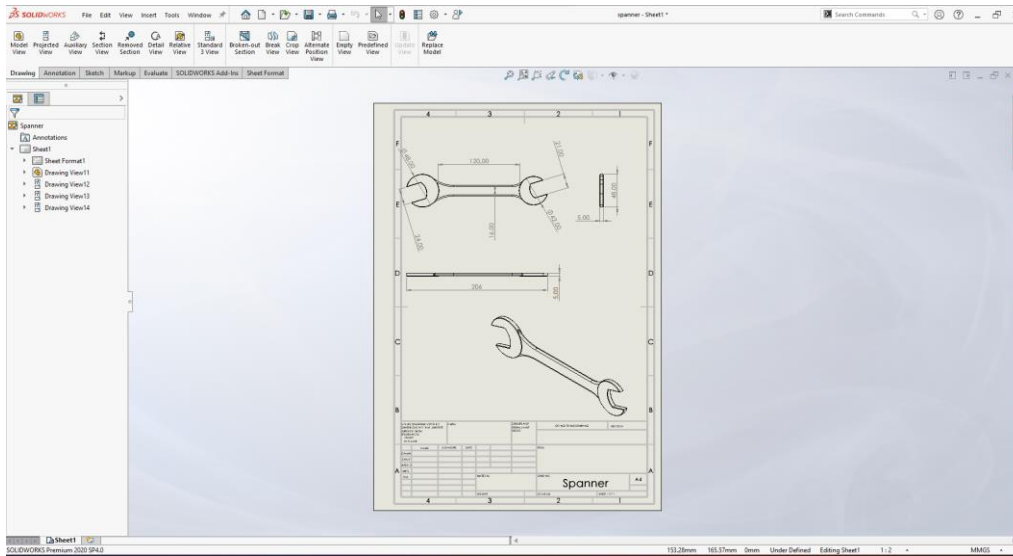


Figure 8: Drawing of a spanner

Finally, SolidWorks will rebuild the model and refresh the application so that the last entered values for the model are shown, Figure 9.

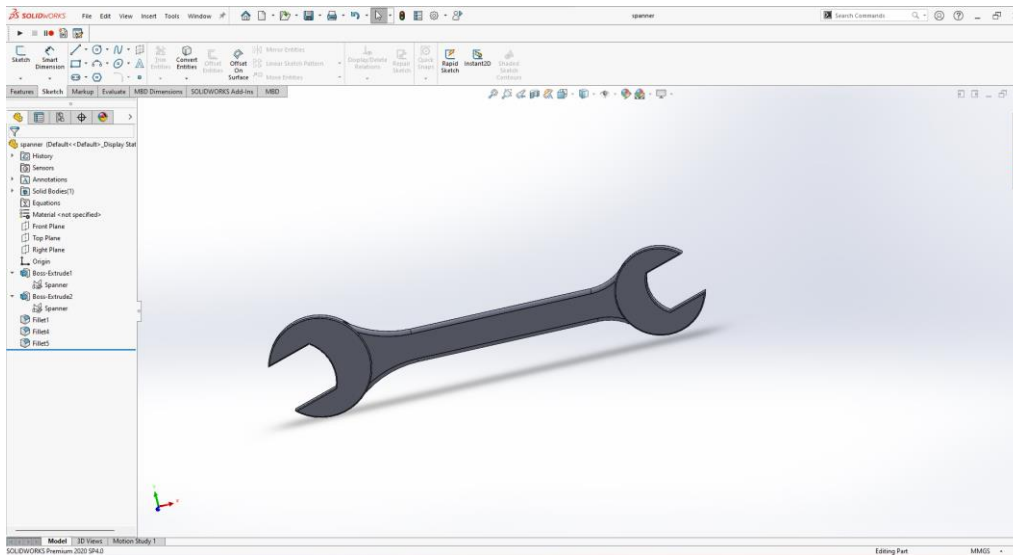


Figure 9: Final model

The process of creation of the three other models is the same as for the spanner. Dimensions are appropriately validated similarly as for the spanner and drawings can be created in the same manner.

5. Conclusions

Product customizations are significant for today's manufacturing industry, in order to gain a competitive edge among their competition. A better understanding regarding product customization and its impact on the customers can lead to a superior manufacturing process, which in turn will improve customer satisfaction. In this paper, an application was created to allow customers to be more involved in the design phase for product customization through the use of an interface that is seamlessly integrated with CAD's system API. The application provides customers with the ability to make design changes to a product via a user-interface form without the need to possess any type of CAD software skills. In the simplest sense, customers input values for pre-identified feature dimensions for a product model and the developed software program will automatically generate a new model based on the parameters. A VBA macro was written for design automation of repetitive and tedious tasks within SolidWorks. By involving customer in the new product development process, companies aim to get concepts and insights that allow them to improve an existing product or launch a new product in the market. Customer involvement using same application in new product development is considered important for successful product development. The concepts developed for this paper can be used for any products with a broad scope of design opportunities. Directions for further work are to upgrade the existing algorithm, so that the user can select the material, color and texture of the tools, which would significantly improve this application.

References:

- [1] A. Srikanth, Product Customization: Benefits, Examples, & Tips, Freshdesk Blog, April 2022, <https://www.freshworks.com/freshdesk/general/product-customization-for-customer-satisfaction-blog/>
- [2] M. Barthwal, 6 Benefits of Product Customization, Knowband, August 2022, <https://www.knowband.com/blog/ecommerce-blog/benefits-product-customization/>
- [3] The Ultimate Guide to Customer Success (2023 Updated), Chisel blog, 2023, <https://chisellabs.com/blog/customer-success/>
- [4] J. Parker, 7 Effective Strategies to Improve Customer Retention, Knowband, August 2021, <https://www.knowband.com/blog/ecommerce-blog/customer-retention-strategies/>
- [5] The Deloitte Consumer Review, Made-to-order: The rise of mass personalisation, Deloitte, 2019, <https://www2.deloitte.com/content/dam/Deloitte/ch/Documents/consumer-business/ch-en-consumer-business-made-to-order-consumer-review.pdf>
- [6] K. Saleh, Online Shopping Personalization – Statistics and Trends, investp, <https://www.invespro.com/blog/online-shopping-personalization/>
- [7] W. C. Shih, The New Era of Industrial Policy Is Here, Harvard Business Review, August 2023, <https://hbr.org/>
- [8] What is Competitive Advantage?, Chisel blog, July 2021, <https://chisellabs.com/glossary/what-is-competitive-advantage/>
- [9] I. P. Nwaokorie, Customer Involvement in New Product Development: Investigation of customer involvement in the development of a product in a clean-tech company in Sweden, Master's Thesis, Uppsala Universitet, September 2020, <https://uu.diva-portal.org/smash/get/diva2:1529472/FULLTEXT01.pdf>
- [10] Z. F. Z. Abidin, M. N. O. Zahid, "Real time object customization in CAD system", Indonesian Journal of Electrical Engineering and Computer Science, Vol. 15, No. 2, August 2019, pp. 1066-1075, ISSN: 2502-4752, <https://ijeecs.iaescore.com/index.php/IJECS/article/view/19076>
- [11] D. R. Chowdhury, "Design Automation Methodology of a Cube using Solidworks and VB.net", International Journal of Engineering Research & Technology (IJERT), Vol. 9 Issue 06, June-2020, pp. 378-383, ISSN: 2278-0181, <https://www.ijert.org/research/design-automation-methodology-of-a-cube-using-solidworks-and-vbnet-IJERTV9IS060352.pdf>
- [12] W. Mao, X. Zhang, S. Tantai, S. Shen, Z. Zhang, J. Wang, "Development of Parametric Drawing System of Corrugated Boxes Based on SolidWorks", IOP Conference Series: Materials Science

- and Engineering, vol. 768, no. 7, 2020, p. 72053, <https://iopscience.iop.org/article/10.1088/1757-899X/768/7/072053/pdf>
- [13] H. Karagulle, “Design Automation of Metal Scrap Balers by Integration of SolidWorks with Excel and VisualBASIC Language”, International Journal of Engineering Research & Technology (Ahmedabad), vol. 7, no. 8, 2018, <https://doi.org/10.17577/IJERTV7IS080095>.
- [14] M. Olander, A. Lüning, “Design automation in industrial order-to-delivery processes: Enabling mass customization of made-to-order products”, 2021, <http://liu.diva-portal.org/smash/get/diva2:1564234/FULLTEXT01.pdf>
- [15] A. Bocevaska, S. Krsteski, S. Savoska, “An Example of Application Design using SolidWorks Application Programming Interface”, 2020, pp. 159-166, <https://ceur-ws.org/Vol-2656/paper16.pdf>