Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ

УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**РАЗРАБОТКА ПЛАГИНА «ТОПОР» ДЛЯ «КОМПАС-3D»**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил:

студент гр. 581

\_\_\_\_\_\_\_\_\_ Устинов В.А

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

Проверил:

к.т.н., доцент каф. КСУП \_\_\_\_\_\_\_\_\_\_ Калентьев А.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024

**1 Описание САПР**

* 1. Описание программы

КОМПАС-3D – это российская импортонезависимая система трёхмерного проектирования, ставшая стандартом для тысяч предприятий и сотен тысяч профессиональных пользователей.

КОМПАС-3D широко используется для проектирования изделий основного и вспомогательного производств в таких отраслях промышленности, как машиностроение (транспортное, сельскохозяйственное, энергетическое, нефтегазовое, химическое и т.д.), приборостроение, авиастроение, судостроение, станкостроение, вагоностроение, металлургия, промышленное и гражданское строительство, товары народного потребления и т. д.[1]

Данная САПР позволяет проектировать модели и сборки разного уровня сложности, благодаря разнообразному функционалу, включающего в себя работу как с 2-мерными эскизами, так и с 3D-моделями. В САПР есть возможность работать со всеми основными примитивами необходимыми для создания эскизов и моделей, а также существует достаточное количество инструментов для работы с 3D-моделями (вытягивание, вращение, вырезание и др.).

Компас 3D имеет множество прямых аналогов на рынке, среди них встречаются Autodesk Inventor, SOLIDWORKS и др.

В рамках дисциплины выбор данной САПР объясняется наличием описания API на русском языке, доступность учебной версии САПР без необходимости получать одобрения от компании, а также большим количеством информации на сторонних ресурсах на русском языке, позволяющим детальнее узнать о возможностях работы с САПР.

* 1. Описание API

API ([аббр.](https://ru.wikipedia.org/wiki/%D0%90%D0%B1%D0%B1%D1%80%D0%B5%D0%B2%D0%B8%D0%B0%D1%82%D1%83%D1%80%D0%B0) от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) application programming interface,дословно интерфейс программирования приложения) — программный интерфейс, то есть описание способов взаимодействия одной компьютерной программы с другими.[2]

Для подключения и работы с API на C# потребуется выполнить ряд следующих действий:

1. Включить в свойствах проекта функцию Register for COM Interop;
2. Создать DLL-обёртку для TLB Компас API с помощью Tlblmp.exe;
3. Подключить созданный DLL к проекту;
4. Зарегистрировать библиотеку в системе КОМПАС (а именно реализовать статический метод типа .htmSample с рядом настроек)
5. Зарегистрировать библиотеку на компьютере пользователя, воспользовавшись утилитой RegAsm.exe

Таблица 1.1 − Используемые свойства класса (интерфейса) Application

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| ActiveDocument | IKompasDocument | Свойство, содержащее текущий активный документ |
| Documents | IDocuments | Коллекция всех открытых документов в приложении |
| Math2D | IMath2D | Интерфейс 2D математики |

Таблица 1.2 − Методы класса (интерфейса) Application

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| ExecuteCompasCommand | commandId, post | bool | Выполнение команды системы КОМПАС |
| MessageBoxEx | Text, caption, flags | long | Выдача всплывающего сообщения |

Таблица 1.3 − Свойства класса (интерфейса) IDocuments

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Item | IKompasDocument | Документ, заданный по имени, ссылке или индексу |

Таблица 1.4 − Используемые методы класса (интерфейса) IDocuments

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Add | Type, Visible | IKompasDocument | Создаёт новый документ |
| Open | PathName, Visible, ReadOnly, LoadCOmbinationIndex | IKompasDocument | Открывает документ (существующий) |

Таблица 1.5 − Используемые свойства класса (интерфейса) IProcess2D

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Angle | double | Угол отклонения в градусах |
| X | double | Координата X |
| Y | double | Координата Y |

Таблица 1.6 − Используемые свойства класса (интерфейса) IProcess3D

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| MateConstraintsObjects | Variant | Выбранные объекты для сопряжения |
| Placement | IPlacement3D | Положение объекта |
| TakeProcessObject | IModelObject | Объект, создаваемый в подпроцессе |

Таблица 1.7 − Используемые методы класса (интерфейса) IProcess3D

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| RunTakeCreateObjectProcess | ProcessType, TakeObject, NeedCreateTakeObj, LostTakeObj | bool | Запустить подчинённый режим создания объектов |

1.3 Обзор аналогов плагина

Первым аналогом является приложения «Teapot Plugin» [3] для Компас-3D, который предназначен для создания различных форм чайников. Он демонстрирует, как можно расширить функциональность КОМПАС-3D для создания 3D-моделей через пользовательский интерфейс плагина. Плагин может служить примером того, как организовать создание объектов с помощью параметров, передаваемых из пользовательского интерфейса в процесс моделирования. Данный аналог является прямым для разрабатываемого плагина «Топор». Интерфейс взаимодействия представлен на рисунке 1.1.



Рисунок 1.1 − Интерфейс приложения «Teapot Plugin»

Вторым аналогом является плагин для генерации модели клинка в КОМПАС-3D. Проект написан на C# и ориентирован на моделирование лезвий.[4] Плагин "Клинок" для КОМПАС-3D предназначен для автоматизированного создания модели клинка. Он разработан на языке C# и интегрируется в среду КОМПАС-3D, предоставляя пользователю инструменты для параметрического проектирования лезвий. Проект находится в открытом доступе на GitHub, что позволяет изучить его и проанализировать любому желающему.



Рисунок 1.2 − Интерфейс приложения «Плагин для Компас 3D “Клинок”»

**2 Описание предмета проектирования**

Топор — это ручной инструмент с острым лезвием на конце, используемый для рубки древесины или других материалов.[5]

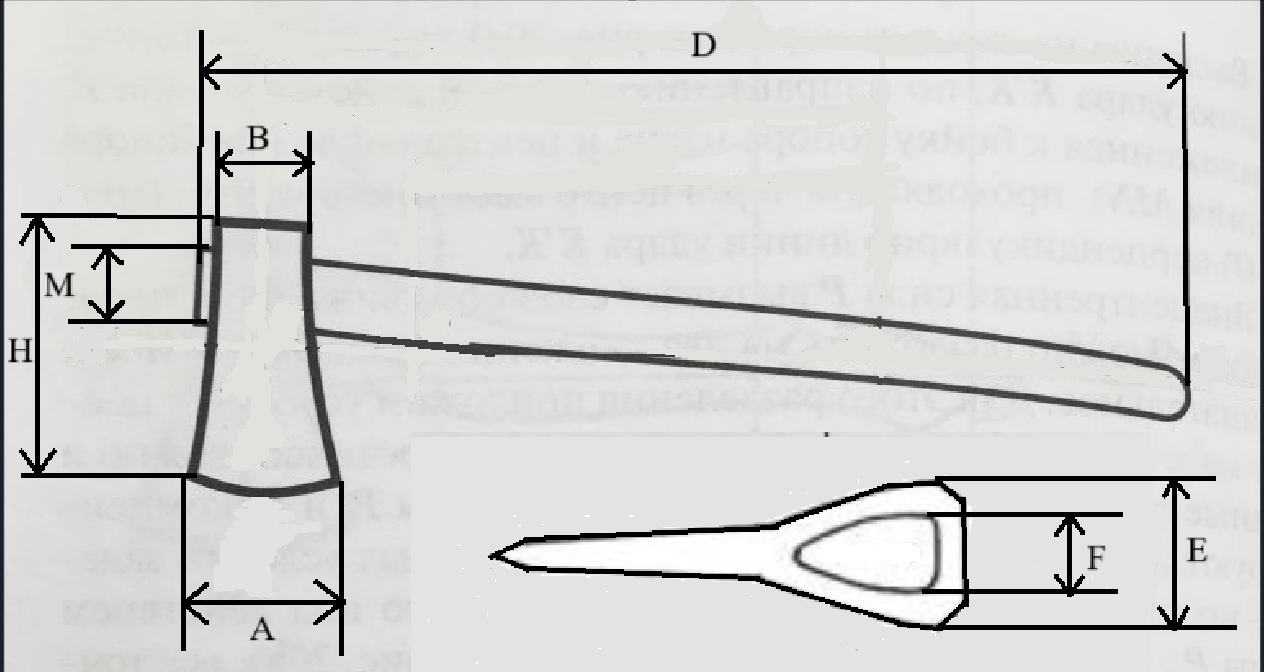


Рисунок 2.1 − Модель топора.

***Изменяемые параметры для предмета проектирования*** (также все обозначения показаны на рисунке 2.1):

− Длина лезвия топора А (100-300мм);

− Угол наклона лезвия диапазон (80-90);

− Длина топорища H (80-150мм);

− Длина ручки топора D (больше в 3 раза длинны лезвия А);

− Длинна обуха B (Составляет 80% от А);

− Ширина рукояти M (составляет 50% от А);

− Ширина топорища E (составляет 30% от длинны обуха В).

**3 Проект системы**

3.1 UML диаграмма классов

UML — это стандартный язык визуального моделирования, предназначенный для следующего использования:

− моделирование бизнеса и подобных процессов;

− анализ, проектирование и внедрения программных систем.

UML — это общий язык для бизнес-аналитиков, архитекторов и разработчиков программного обеспечения, используемый для описания, спецификации, проектирования и документирования существующих или новых бизнес-процессов, структуры и поведения артефактов программных систем.[6]

UML диаграмма классов для плагина «Топор» представлена на рисунке 3.1.

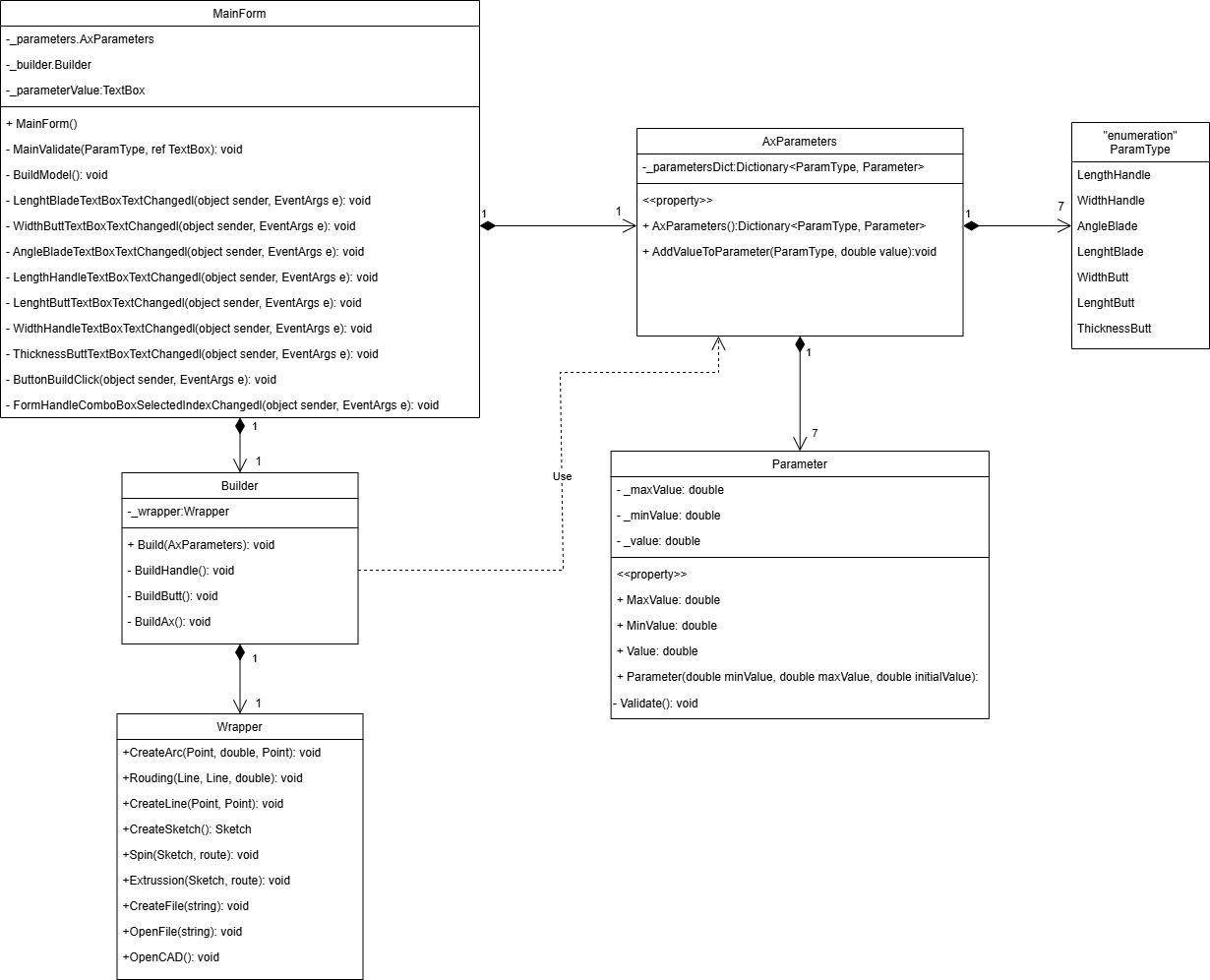


Рисунок 3.1 − UML диаграмма классов для плагина «Топор»

В таблицах ниже представлена информация о свойствах и методах каждого из классов.

Таблица 3.1 − Свойства класса MainForm

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_builder | Builder | Хранит в себе объект построения |
| \_parameters | AxParameters | Хранит в себе параметры для объекта построения |

Таблица 3.2 − Методы класса MainForm

|  |  |  |
| --- | --- | --- |
| Название | Входные параметры | Описание |
| BuildModel | − | Построение модели по заданным параметрам |
| LenghtBladeTextBox | object sender, EventArgs e | Ввод длинны лезвия |
| WidthButtTextBox | object sender, EventArgs e | Ввод длинны топорища |
| AngleBladeTextBox | object sender, EventArgs e | Ввод угла наклона |
| LengthHandleTextBox | object sender, EventArgs e | Ввод длины ручки топора |
| LenghtButtTextBox | object sender, EventArgs e | Ввод длинны обуха |
| WidthHandleTextBox | object sender, EventArgs e | Ввод длина топорища |
| ThicknessButtTextBox | object sender, EventArgs e | Ввод ширина топорища |
|  | | |
| Продолжение таблицы 3.2 − Методы класса MainForm | | |
| ButtonBuild | object sender, EventArgs e | Запуск  построения модели по заданным параметрам |
| FormHandleComboBox | object sender, EventArgs e | Выбор формы ручки |
| MainValidate | ParamType, ref textBox | Второй уровень валидации |

Таблица 3.3 − Свойства класса AxParameters

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_parametersDict | Dictionary<ParamType, Parameter> | Хранит в себе словарь с параметрами объекта построения |

Таблица 3.4 − Методы класса AxParameters

|  |  |  |
| --- | --- | --- |
| Название | Входные параметры | Описание |
| AxParameters | Dictionary<ParamType, Parameter> | Конструктор |
| AddValueToParameter | ParamType, double value | Добавление параметров |

Таблица 3.5 − Свойства класса Builder

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_wrapper | Wrapper | Хранит в себе объект обёртки API |

Таблица 3.6 − Методы класса Builder

|  |  |  |
| --- | --- | --- |
| Название | Входные параметры | Описание |
| Build | AxParameters | Построение модели по заданным параметрам |
| BuildButt | − | Построение обуха топора |
| BuildHandle | − | Построение ручки топора |
| BuildAx | − | Сборка топора |

Таблица 3.7 − Свойства класса Parameter

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_maxValue | double | Максимально допустимое значение параметра |
| \_minValue | double | Мин. допустимое значение параметра |
| \_value | double | Значение параметра |

Таблица 3.7 − Методы класса Parameter

|  |  |  |
| --- | --- | --- |
| Название | Входные параметры | Описание |
| MaxValue | double | Геттер, Сеттер |
|  |  |  |
|  |  |  |
| MinValue | double | Геттер, Сеттер |
| Value | double | Геттер, Сеттер |
| Parameter | double minValue, double maxValue, double initialValue | Конструктор |
| Validate | − | Валидация |

Таблица 3.8 − Методы класса Wrapper

|  |  |  |
| --- | --- | --- |
| Название | Входные параметры | Описание |
| CreateArc | Point, double, Point | Создание дуги по двум точкам |
| Rounding | Line, Line, double | Скругление |
| CreateLine | Point, Point | Создание линии |
| Sketch | − | Создание эскиза |
| Spin | Sketch, route | Вращение эскиза |
| Extrusion | Sketch, route | Выдавливание эскиза |
| CreateFie | string | Создание файла |
| OpenFile | string | Открытие файла |
| ModifyFile | − | Редактированиефайла |
| OpenCAD | − | Открытие Компас3D |

3.2 Макеты пользовательского интерфейса

На рисунке 3.2 и 3.3 представлены макет пользовательского интерфейса, а также валидация введенных значений, при котором пользователь вводит значение являющимся некорректным.

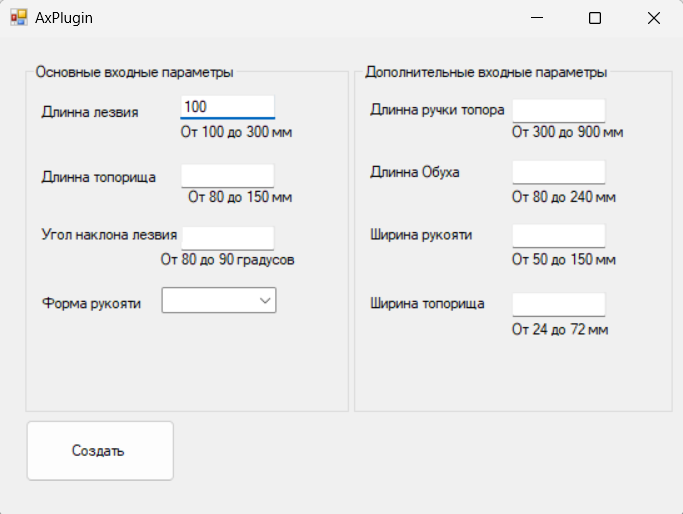


Рисунок 3.2 − Макет пользовательского интерфейса

При задании параметра значение введенное в TextBox сначала проверяется на верный тип данных, а именно double, при помощи метода MainValidate() в MainForm. Далее MainValidate вызывает метод AddValueToParameter() в переменной Parameters \_parameter для добавления значения в словарь \_parametersDict. Метод AddValueToParameter() вызывает Set для свойства Value(), задает новое значение. Метод Validate() класса Parameter вызывается в сеттере свойства Value().

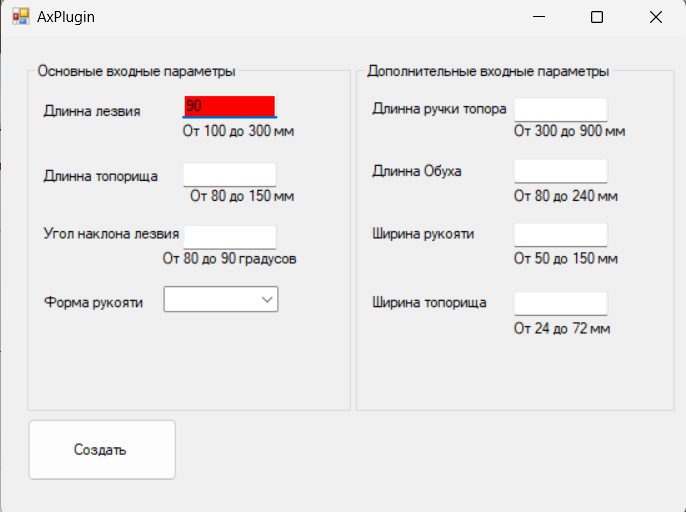


Рисунок 3.3 – Реакция системы на ошибки в введенных параметрах

**4 Список источников**

1. КОМПАС-3D [Электронный ресурс]. − Режим доступа <https://kompas.ru/kompas-3d/about/> (дата обращения 28.09.2024)
2. API [Электронный ресурс]. − Режим доступа <https://habr.com/ru/articles/464261/> (дата обращения 28.09.2024)
3. Teapot Plugin [Электронный ресурс]. − Режим доступа <https://github.com/kurocha/teapot> (дата обращения 29.09.2024)
4. Плагин для компас 3D “Клинок”. [Электронный ресурс]. − Режим доступа https://github.com/eskorpado/kompas-blade-plugin (дата обращения 05.10.2024)
5. Топор [Электронный ресурс]. − Режим доступа <https://zlatoust-air.ru/blog/sovety-pokupatelyam/topor-vidy-kharakteristiki-naznachenie-i-vybor/>

(дата обращения 18.09.2024)

6.UML [Электронный ресурс]. − Режим доступа https://www.uml-diagrams.org/ (дата обращения 07.10.2024)