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

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**РАЗРАБОТКА ПЛАГИНА «Шток» ДЛЯ САПР «КОМПАС-3D 2022»**

Проект системы по

по дисциплине «ОСНОВЫ РАЗРАБОТКИ САПР»

Выполнил:

студент гр. 589-3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_В.С. Островский

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.А. Калентьев

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

**Оглавление**

[1 Описание САПР 3](#_Toc116637014)

[1.1 Описание программы 3](#_Toc116637015)

[1.2 Описание API 4](#_Toc116637016)

[1.3 Обзор аналогов 8](#_Toc116637017)

[2 Описание предмета проектирования 9](#_Toc116637018)

[3 Проект программы 10](#_Toc116637019)

[3.1 Описание технических и функциональных аспектов проекта 10](#_Toc116637020)

[3.2 Диаграмма классов 11](#_Toc116637021)

[3.3 Макет пользовательского интерфейса 13](#_Toc116637022)

[Список литературы 15](#_Toc116637023)

# 1 Описание САПР

# Описание программы

Система «Компас-3D» предназначена для создания трёхмерных ассоциативных моделей отдельных деталей (в том числе, деталей, формируемых из листового материала путём его гибки) и сборочных единиц, содержащих как оригинальные, так и стандартизованные конструктивные элементы. Параметрическая технология позволяет быстро получать модели типовых изделий на основе проектированного ранее прототипа. Многочисленные сервисные функции облегчают решение вспомогательных задач проектирования и обслуживания производства.

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. [2]

# Описание API

API (англ. Application Programming Interface — программный интерфейс приложения) — это набор способов и правил, по которым различные программы общаются между собой и обмениваются данными. Все эти коммуникации происходят с помощью функций, классов, методов, структур, а иногда констант одной программы, к которым могут обращаться другие.

Для КОМПАС-3D созданы две различные версии API - версии 5 и версии 7. К ним разработчик прилагает справочную систему по всем включенным в эту API интерфейсам.

|  |  |  |
| --- | --- | --- |
| Название интерфейса | Описание интерфейса | |
| KompasObject | Интерфейс API КОМПАС | |
| ksEntity | Интерфейс элемента модели (оси, плоскости, формообразующего элемента) | |
| ksDocument2D | Интерфейс графического документа системы КОМПАС | |
| ksSketchDefinition | Интерфейс параметров эскиза | |
| ksDocument3D | | Интерфейс документа-модели |
| ksPart | | Интерфейс детали или подсборки в составе сборки |
| ksBaseExtrusionDefinition | | Интерфейс параметров основания - элемента выдавливания |
| ksCircularPartArrayDefinition | | Интерфейс операции копирования по окружности |

Таблица 1.1 – Интерфейсы, используемые для разработки.

В нижеописанных таблицах представлены методы, которые будут использоваться при разработке плагина, а также описание входных параметров данных методов (таблицы 1.2 – 1.15).

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| Document3D() | ksDocument | Метод для получения указателя на интерфейс трехмерного графического документа (детали или сборки) |
| ActivateControllerAPI() | bool | Метод для активации API КОМПАС-3D |
| Visible() | bool | Свойство видимости приложения |

Таблица 1.2 – Используемые методы интерфейса KompasObject

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| Create() | bool | Создать объект в модели |
| GetDefinition() | IUnkown | Получить указатель на интерфейс параметров объектов и элементов |

Таблица 1.3 – Используемые методы интерфейса ksEntity

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемое значение | Описание |
| ksCircle (double xc, double yc, double rad, long style) | Указатель на окружность – в случае удачного завершения, 0 – в случае неудачи | Создать окружность |

Таблица 1.4 – Используемые методы интерфейса ksDocument2D

|  |  |  |
| --- | --- | --- |
| Метод | Входной параметр | Описание входного параметра |
| ksCircle (double xc, double yc, double rad, long style) | xc, yc | Координаты центра окружности |
| rad | Радиус окружности |
| style | Стиль линии |

Таблица 1.5 – Описание входных параметров, используемых методов интерфейса ksDocument2D

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| BeginEdit() | bool | Войти в режим редактирования эскиза (ksDocument2D) |
| EndEdit() | bool | Выйти из режима редактирования эскиза |

Таблица 1.6 – Используемые методы интерфейса ksSketchDefinition

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| Create (bool invisible, bool typeDoc) | bool | Создать документ-модель (деталь или сборку) |
| GetPart (int type) | ksPart | Получить указатель на интерфейс компонента в соответствии с заданным типом |

Таблица 1.7 – Используемые методы интерфейса ksDocument3D

|  |  |  |
| --- | --- | --- |
| Метод | Входной параметр | Описание входного параметра |
| Create (bool invisible, bool typeDoc) | invisible | Признак режима редактирования документа (TRUE – невидимый режим, FALSE – видимый режим) |
| typeDoc | Тип документа (TRUE – деталь, FALSE – сборка) |
| GetPart (int type) | type | Тип компонента из перечисления: pInPlace\_Part – компонент, редактируемый на месте; pNew\_Part – новый компонент; pEdit\_Part –редактируемый компонент; pTop\_Part – главный компонент, в составе которого находится новый или редактируемый или указанный компонент (например, сборка, в составе которой находится редактируемая деталь) |

Таблица 1.8 – Описание входных параметров, используемых методов интерфейса ksDocument3D

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| GetDefaultEntity (short objType) | ksEntity | Получить указатель на интерфейс объекта, создаваемого системой по умолчанию |
| NewEntity (short objType) | ksEntity | Создать новый интерфейс объекта и получить указатель на него |

Таблица 1.9 – Используемые методы интерфейса ksPart

|  |  |
| --- | --- |
| Входной параметр | Описание параметра |
| objType | Тип объекта |

Таблица 1.10 – Описание входных параметров, используемых методов интерфейса ksPart

|  |  |  |
| --- | --- | --- |
| Метод | Тип объекта | Название объекта |
| GetDefaultEntity (short objType) | o3d\_planeXOY | Плоскость XOY |
| o3d\_axisOZ | Ось OZ |
| NewEntity (short objType) | o3d\_sketch | Эскиз |
| o3d\_baseExtrusion | Базовая операция выдавливания |
| o3d\_cutExtrusion | Вырезать выдавливанием |
| o3d\_circularCopy | Операция копирования по концентрической сетке |

Таблица 1.11 – Используемые типы объектов в методах интерфейса ksPart

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| SetSideParam (bool forward, short type, double depth, double draftValue, bool draftOutward) | bool | Установить параметры выдавливания в одном направлении |
| SetSketch (LPDISPATCH sketch) | bool | Задать указатель на интерфейс эскиза элемента |

Таблица 1.12 – Используемые методы интерфейса ksBaseExtrusionDefinition

|  |  |  |
| --- | --- | --- |
| Метод | Входной параметр | Описание входного параметра |
| SetSideParam (bool forward, short type, double depth, double draftValue, bool draftOutward) | forward | Направление выдавливания: TRUE - прямое направление, FALSE - обратное направление |
| type | Тип выдавливания |
| depth | Глубина выдавливания |
| draftValue | Угол уклона |
| draftOutward | Направление уклона: FALSE – уклон наружу, TRUE – уклон внутрь |
| SetSketch (LPDISPATCH sketch) | sketch | Указатель на интерфейс эскиза ksEntity |

Таблица 1.13 – Описание входных параметров, используемых методов интерфейса ksBaseExtrusionDefinition

|  |  |  |
| --- | --- | --- |
| Название | Возвращаемый тип | Описание |
| SetCopyParamAlongDir (long count, double step, bool, factor, bool dir) | bool | Установить параметры копирования |
| SetAxis (LPDISPATCH axis) | bool | Установить указатель на ось копирования |

Таблица 1.14 – Используемые методы интерфейса ksCircularPartArrayDefinition

|  |  |  |
| --- | --- | --- |
| Метод | Входной параметр | Описание входного параметра |
| SetCopyParamAlongDir (long count, double step, bool factor, bool dir) | count | Количество копий |
| step | Шаг |
| factor | Признак полного шага |
| dir | Направление |
| SetAxis (LPDISPATCH axis) | axis | Указатель на интерфейс оси ksEntity |

Таблица 1.15 – Описание входных параметров, используемых методов интерфейса ksCircularPartArrayDefinition

# Обзор аналогов

**«nanoCAD Механика»**

«nanoCAD Механика» является решением для машиностроительного проектирования, базирующимся на платформе «nanoCAD Plus» от российского разработчика – компании «Нанософт». Помимо богатого функционала для оформления конструкторской и технологической документации по ЕСКД и ЕСТД, программа располагает всеми основными инструментами цифрового проектирования. [4]

Модуль «Механика» имеет широкую библиотеку стандартных деталей, в том числе и штоков, выполненных по ГОСТ, и поддерживает возможность изменять параметры детали под необходимые нужды.

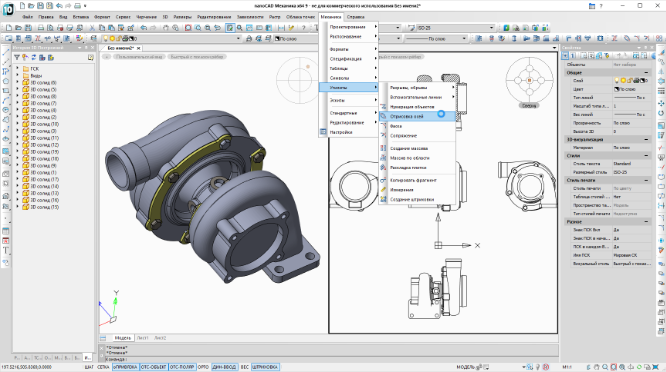


Рисунок 1.1 — Классический интерфейс «nanoCAD Механика»

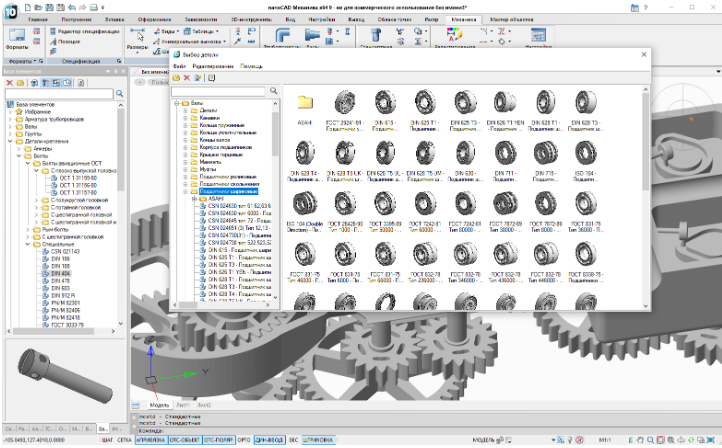


Рисунок 1.2 – Каталог стандартных деталей в nanoCAD

# 2 Описание предмета проектирования

Шток — цельный или полый стержень цилиндрической формы, соединяющий поршень и кронштейн амортизатора или других движущихся частей.[5]

На рисунке 2.1 представлен чертеж штока.

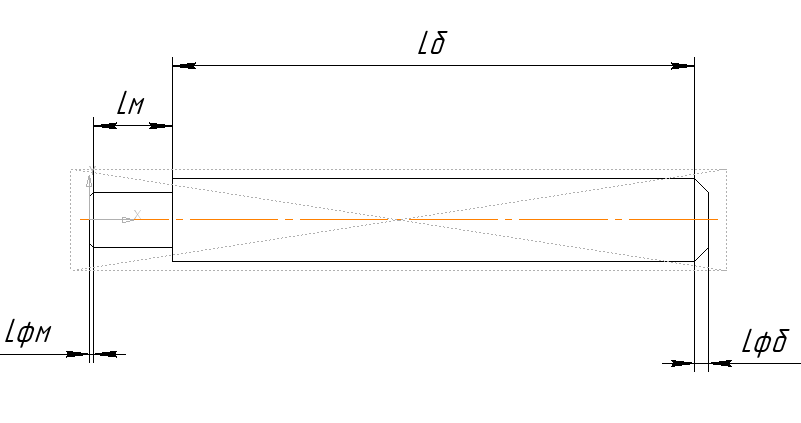


Рисунок 2.1 – Чертеж штока

Параметры штока:

1. Длина большой части Lб (от 100 мм до 180мм)
2. Длина малой части Lм – не больше 1/2 Lб, не меньше 1/4 (от 20 мм до 90мм)
3. Диаметр большой части Dб (от 10 мм до 20 мм)
4. Диаметр малой части Dм – не больше 70% Dб, не меньше 50% Dб (от 5 мм до 14 мм)
5. Длина фаски малой части Lфм – 0, либо от 5 мм до 10 мм
6. Длина фаски большей части Lфб – 0, либо от 5мм до 10 мм
7. Угол фаски малой части – 30, 45, 60 градусов
8. Угол фаски большей части – 30, 45, 60 градусов.

# 3 Проект программы

# 3.1 Описание технических и функциональных аспектов проекта

Для графического описания абстрактной модели проекта, а также пользовательского взаимодействия (сценарии действия) использован стандарт UML.

UML язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это – открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML – моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML возможна генерация кода и наоборот.[6]

При использовании UML были простроена диаграмма классов.

# 3.2 Диаграмма классов

Диаграмма классов – структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей между ними.[7]

На рисунке 3.1 представлена диаграмма классов.

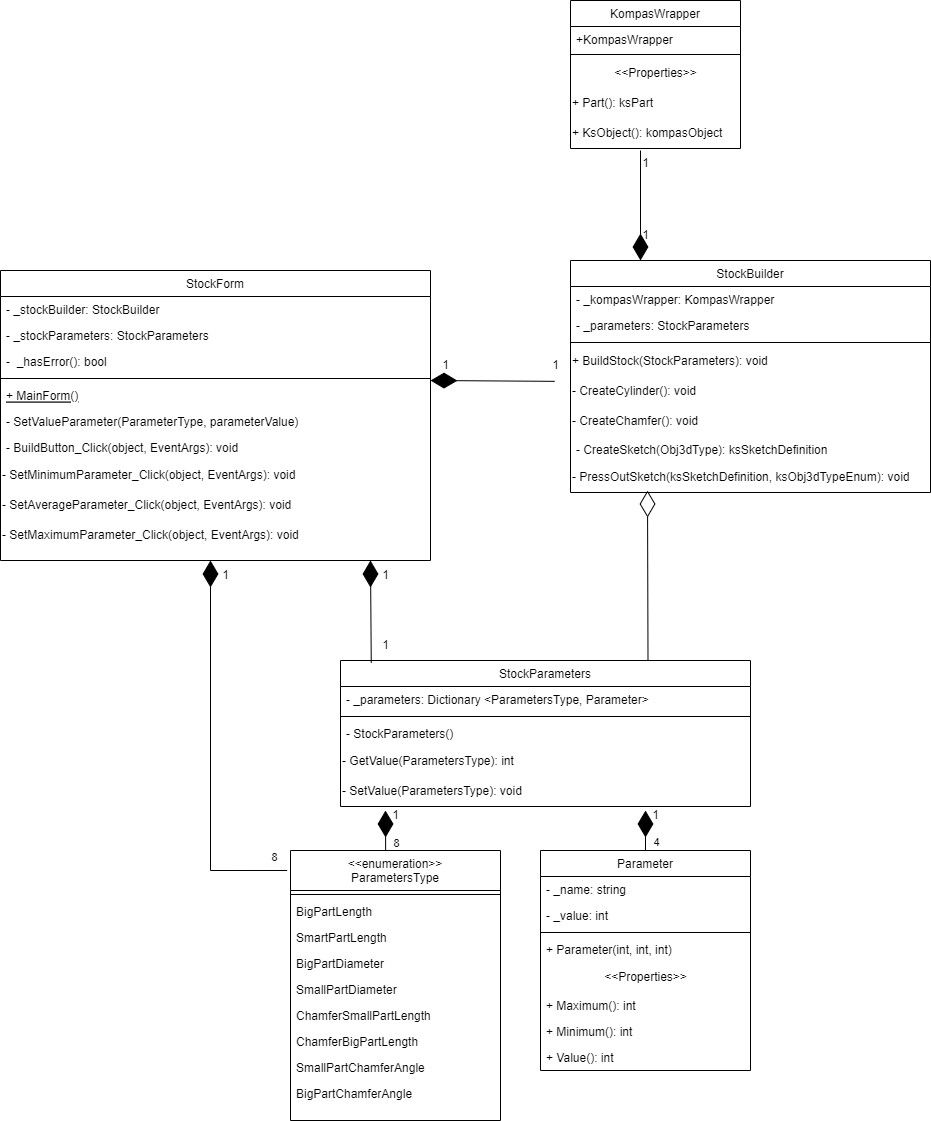


Рисунок 3.1 – Диаграмма классов

Краткое описание классов представлено в таблице 3.1.

|  |  |
| --- | --- |
| Класс | Описание |
| StockForm | Содержит в себе текстовые параметры для дальнейшей передачи их в класс StockParameters, а также кнопку для построения стола, которая вызывает метод BuildStock у класса StockBuilder. |
| KompassWrapper | Класс «обёртка» для объекта KompasAPI. Через него осуществляется доступ к API Компас-3D, и вызова методов API. |
| StockBuilder | Содержит в себе TableParameters и KompasWrapper. С помощью него осуществляется построение стола. |
| TableParameters | Содержит в себе все параметры, необходимые для построения стола |
| Parameter | Класс, содержащий в себе конкретный параметр и его основные параметры, такие как максимальное, минимальное и текущее значение |
| ParametersType | Содержит в себе все типы параметров стола |

# 3.3 Макет пользовательского интерфейса

Макет пользовательского интерфейса представляет собой форму для ввода параметров кружки. Построение модели осуществляется путем нажатия на кнопку «Build Mug». Если какое-то из полей не будет проходить проверку, то кнопка «Build Mug» будет неактивна.

На рисунке 3.2 представлен макет пользовательского интерфейса.

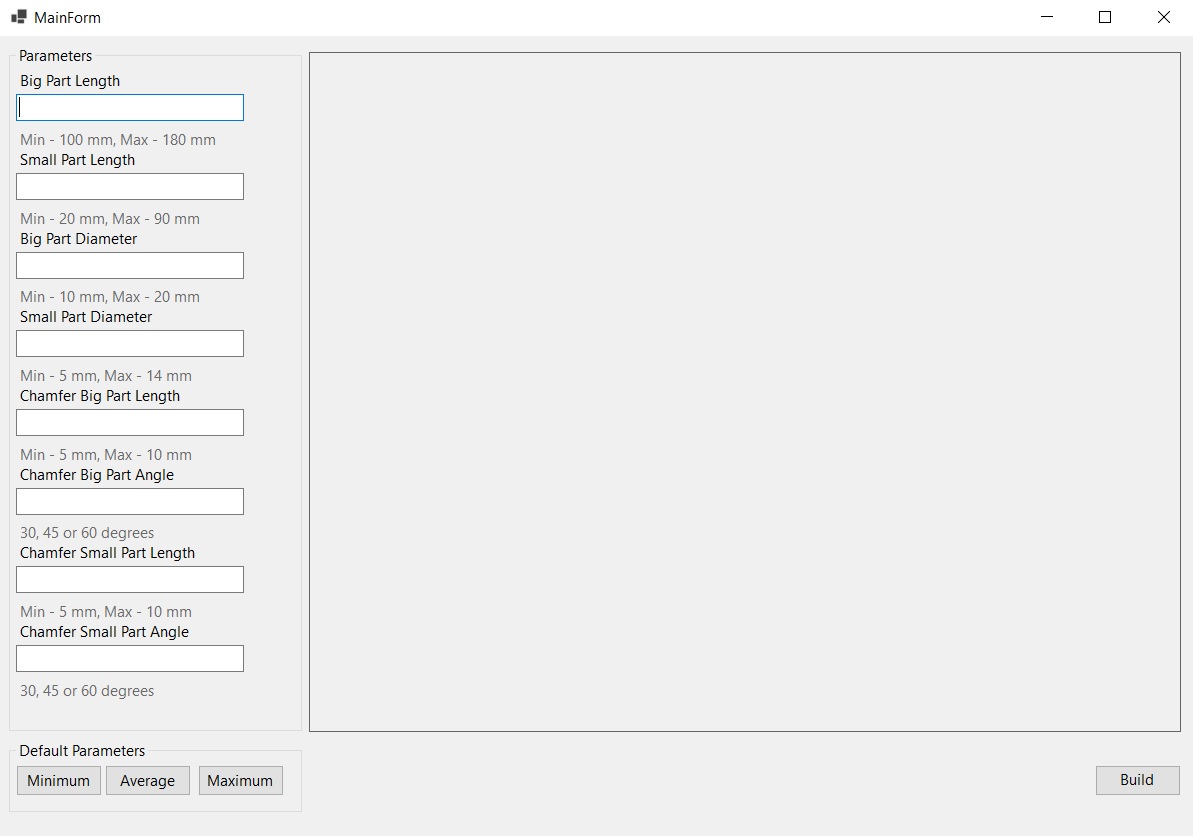


Рисунок 3.2 – Макет пользовательского интерфейса

После ввода некорректных значений, появится уведомление, пример представлен на рисунке 3.3.

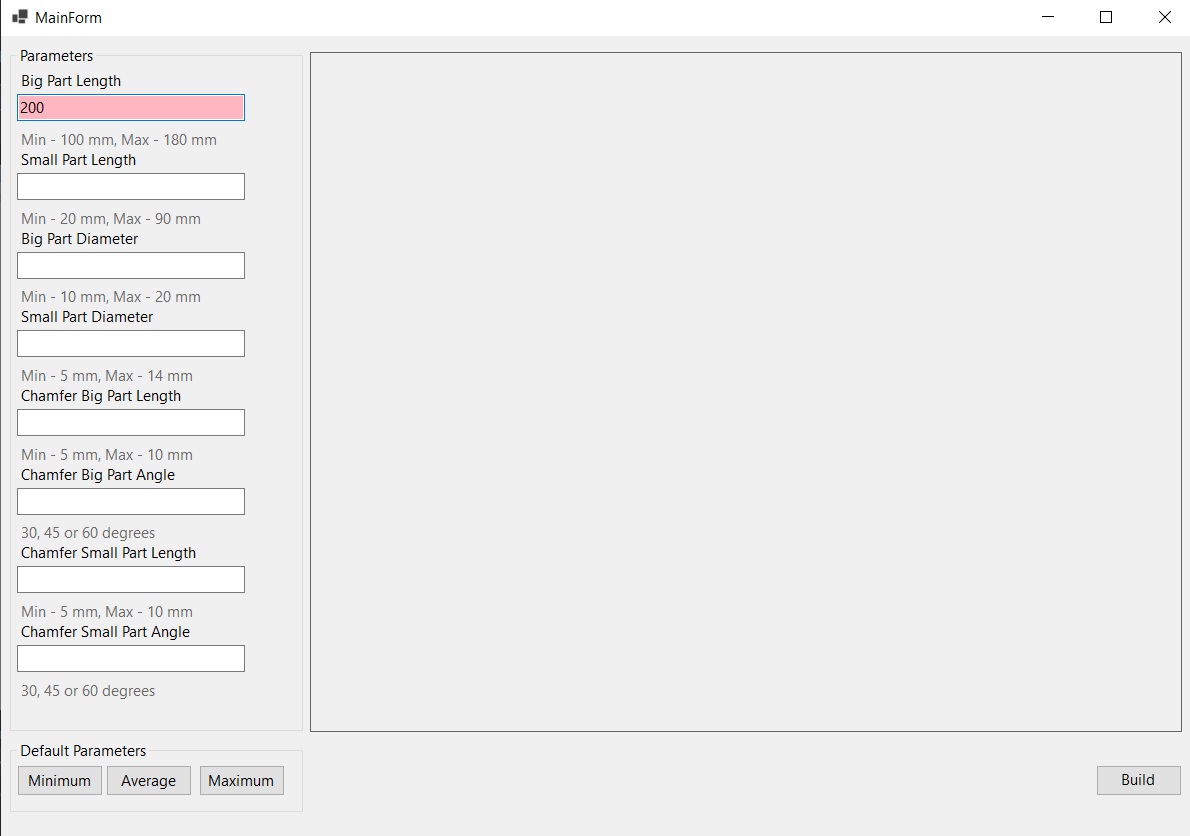


Рисунок 3.3 — Уведомление об ошибке

# Список литературы

1. САПР — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/Система\_автоматизированного\_проектирования (дата обращения 13.10.2022).

2. КОМПАС 3D — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B0%D1%81\_(%D0%A1%D0%90%D0%9F%D0%A0) (дата обращения 13.10.2022).

3. API — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/API (дата обращения 13.10.2022).

4. NanoCAD Модуль «Механика». [Электронный ресурс]. — Режим доступа: https://www.nanocad.ru/products/module\_mechanica/ (дата обращения 20.10.2022).

5. Шток — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/%D0%A8%D1%82%D0%BE%D0%BA (дата обращения 20.10.2022).

6. UML. [Электронный ресурс]. – Режим доступа: http://www.uml.org/ (дата обращения 13.10.2022).