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 «Кружка» ДЛЯ САПР «КОМПАС-3D 2022»**

Проект системы по

по дисциплине «ОСНОВЫ РАЗРАБОТКИ САПР»

Выполнил:

студент гр. 589-3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_М.В. Пахомов

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 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.

Наиболее важными для разработки любого приложения могут посчитаться следующие интерфейсы:

1 - **IKompasAPIObject**: базовый интерфейс для всех интерфейсов КОМПАС API, кроме интер­фейсов событий и некоторых вспомогательных интерфейсов;

2 - **IAp****plic****ation**: интерфейс приложения КОМПАС-3D;

3 - **IDocuments**: коллекция документов, открытых в приложении КОМПАС-3D;

4 - **IKompasError**: интерфейс информации об ошибках системы КОМПАС-3D;

5 - **IModelObject**: базовый интерфейс для всех модельных объектов.

Также, интересными для разработки конкретно лабораторного приложения можно посчитать следующие интерфейсы:

1 - **IModelContainer** – позволяет работать с коллекциями 3D-объектов, входящих в состав 3D-объекта.;

2 - **ISurfaceContainer** – устанавливает и получает коллекции операций с поверхностями;

3 - **ISketch** – интерфейс взаимодействия с эскизом;

4 - **IPart7** – интерфейс компонента 3D документа;

5 - **IPlane3D** - интерфейс плоскости 3D.

Далее будут приведены самые важные для использования в лабораторной программе методы и свойства интерфейсов. Конечно же, программа не ограничится их использованием, а будет также применять более локальные методы и свойства, предсказать использование которых будет достаточно сложно:

Таблица 1.1 - Необходимые методы/свойства класса IKompasAPIObject

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| Application | Указатель на интерфейс IApplication | Позволяет получить ссылку на приложение |

Таблица 1.2 - Необходимые методы/свойства класса IApplication

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| ActiveDocument | Указатель на интерфейс IKompasDocument | Позволяет получить текущий активный документ |
| HideMessage | Константа из перечисления ksHideMessageEnum | Позволяет скрывать/показывать сообщения |
| KompasError | Указатель на интерфейс IKompasError информации о ошибке системы КОМПАС | Позволяет получить информацию об ошибке системы КОМПАС |
| Visible | BOOL | Позволяет изменить видимость приложения |
| Quit | - | Позволяет закрыть приложение |

Таблица 1.3 - Необходимые методы/свойства класса IDocuments

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| Add | Указатель на интерфейс IKompasDocument | Позволяет создать новый документ и добавить его в коллекцию |

Таблица 1.4 - Необходимые методы/свойства класса IKompasError

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| Clear | - | Позволяет сбросить ошибку |
| Report | - | Позволяет вывести сообщение о ошибке |
| Code | Из перечисления ErrorType и ErrorType3d. | Позволяет получить код ошибки |
| Description | BSTR (строка) | Позволяет получить описание ошибки |
| Error3D | BOOL | Позволяет определить, какому из двух перечисле­ний соответствует код ошибки, полученный через свойство Code |

Таблица 1.5 - Необходимые методы/свойства класса IModelObject

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| Update | BOOL | Метод Update необходимо вызвать для вступления в силу примененных изменений объекта |
| Hidden | BOOL | Позволяет получить и установить состояние видимости объекта |
| Part | Указатель на интерфейс IPart7. | Позволяет получить компонент, владеющий элементом |

Таблица 1.6 - Необходимые методы/свойства класса IModelContainer

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| AddObject | Указатель на интерфейс IModelObject | Создает новый элемент 3D модели |

Таблица 1.7 - Необходимые методы/свойства класса ISurfaceContainer

|  |  |  |
| --- | --- | --- |
| Название метода/свойства | Тип возвращаемого значения | Описание |
| EvolutionSurfaces | Указатель на интерфейс IEvolutions | Экспорт коллекции поверхностей выдавливания |

# Обзор аналогов

**Онлайн-конструктор Mug3D**

Mug3D – это онлайн-конструктор кружки с возможностью предварительного 3D-просмотра макета. Редактор позволяет загружать фото, картинки, добавлять надписи разными цветами, перемещать, вращать и масштабировать объекты — можно создать любой принт на кружку.[4]

В Mug3D вы можете создать свой собственный индивидуальный дизайн кружки для сублимационной печати, повращать 3D-модель, скачать созданный макет для печати в формате PNG, сделать скриншот 3D-сцены в формате PNG, а также записать видео в формате WEBM.

Интерфейс онлайн-конструктора показан на рисунке 1.1.

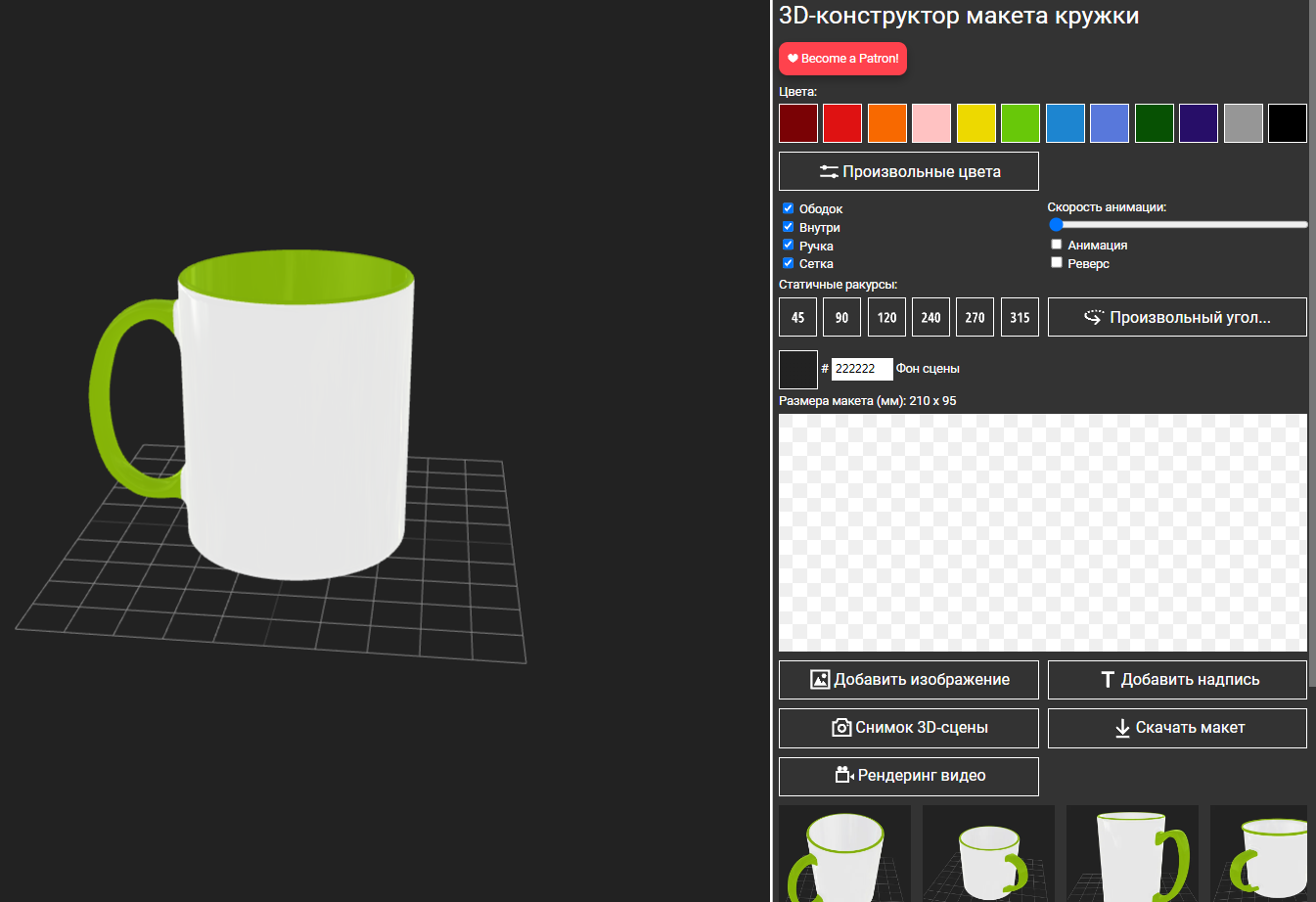


Рисунок 1.1 — Интерфейс онлайн-конструктора Mug3D

# 2 Описание предмета проектирования

Кружка – полое изделие различной формы, преимущественно с ручкой, предназначенное для питья. В общих чертах кружка имеет форму, приближающуюся к усеченному конусу или цилиндру, и варьирующуюся от одного производителя к другому.[5]

На рисунке 2.1 представлен чертеж забора.

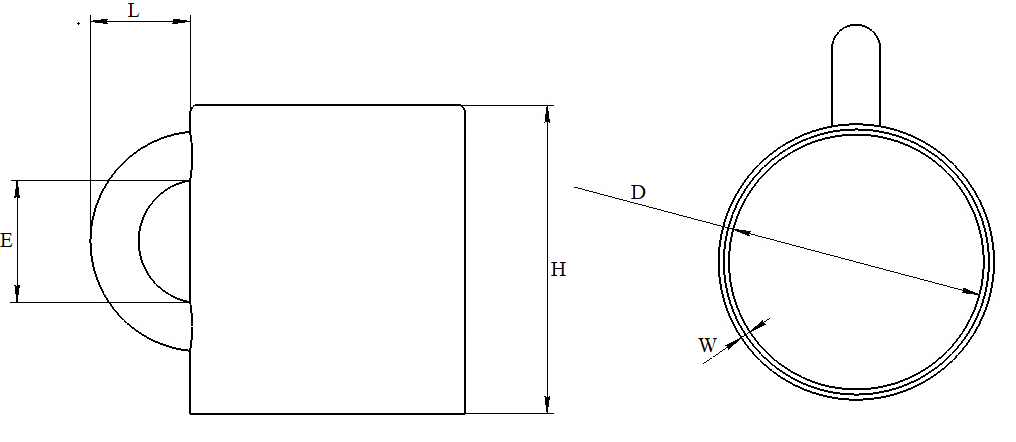


Рисунок 2.1 – Чертеж кружки

Параметры забора:

1. диаметр горла кружки ***D*** (мин - 70мм, макс - 105 мм);
2. высота кружки ***H*** (мин – 85 мм, макс – 130 мм);
3. расстояние между внешней и внутренней частями горла кружки ***W*** (мин – 5мм, макс – 10мм);
4. длина ручки кружки ***L*** (зависит от внутреннего диаметра кружки: мин – E (мин) \* 0.5, макс – E (макс) \* 0.5);
5. внутренний диаметр ручки кружки ***E*** (зависит от высоты кружки в соотношении: мин – H (мин) \* 0.7, макс – H (макс) \* 0.7);

# 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.

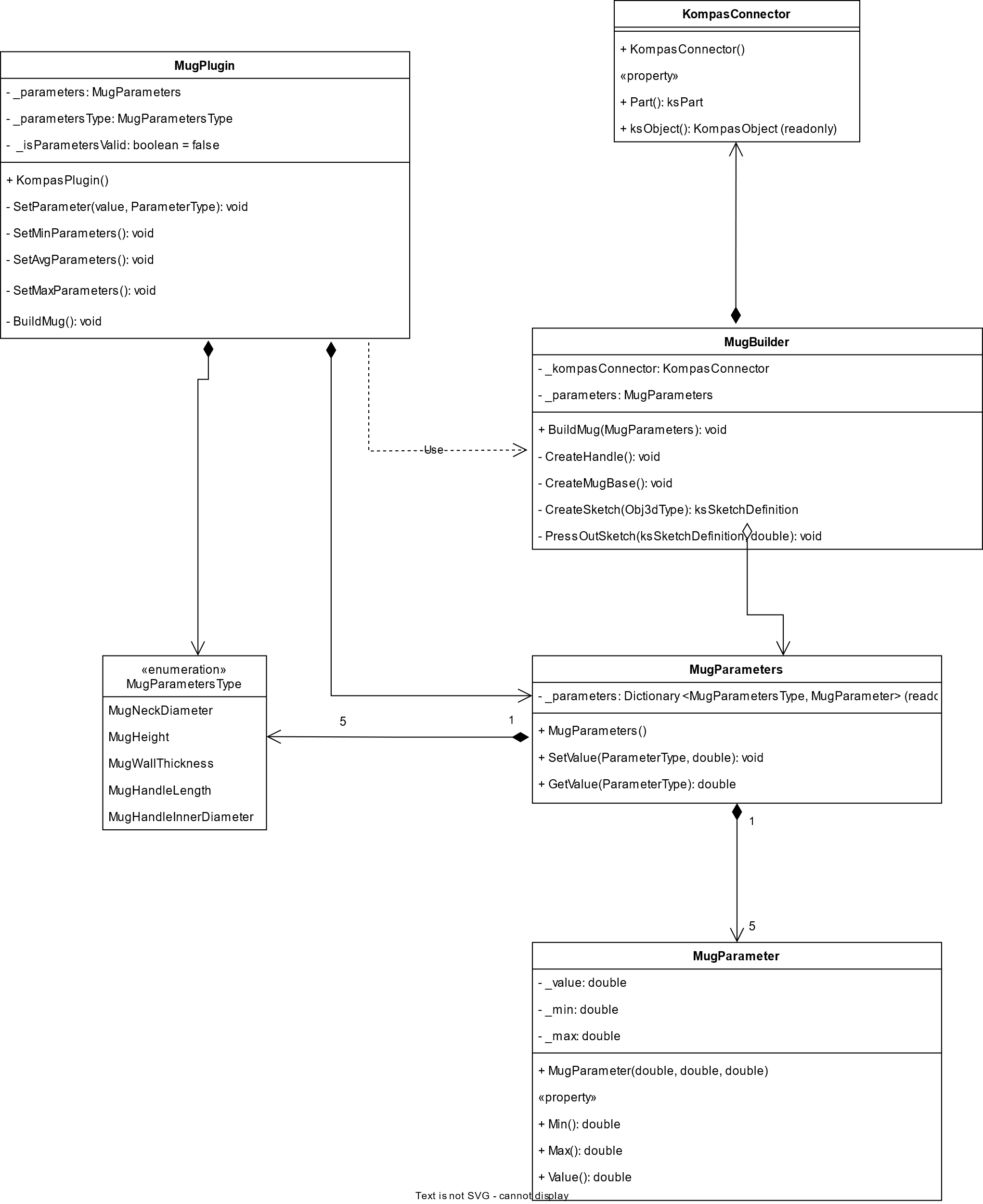


Рисунок 3.1 – Диаграмма классов

Класс KompasConnector содержит все необходимые методы для запуска плагина в КОМПАС 3D.

Класс MugPlugin связывает модели и представление.

Класс MugParameters хранит свойство parameters, которое возвращает словарь, где ключом является значение из перечисления MugParametersType, а значением параметр типа MugParameter.

Класс MugParameter хранит данные о конкретном параметре, его минимальное, максимальное и текущее значения.

Класс MugBuilder используется для построения 3D-модели кружки. Данный класс хранит в себе объект класса-обертки, используемой САПР. При вызове метода BuildMug() обращается к методам класса-обертки, чтобы построить 3D-модель на основании списка параметров, передаваемого в данный метод с объектом класса MugParameters.

# 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.

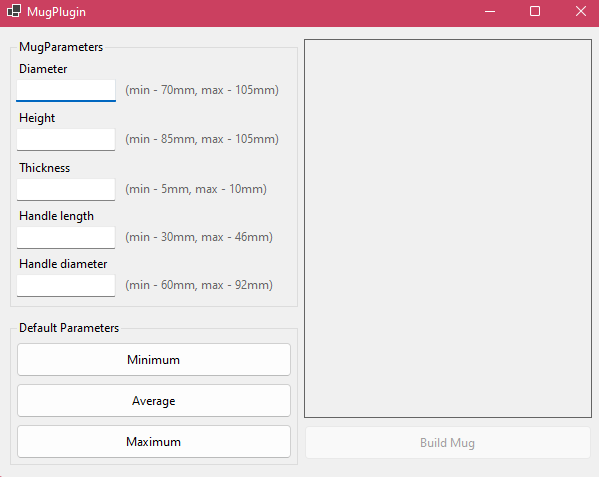


Рисунок 3.2 – Макет пользовательского интерфейса

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

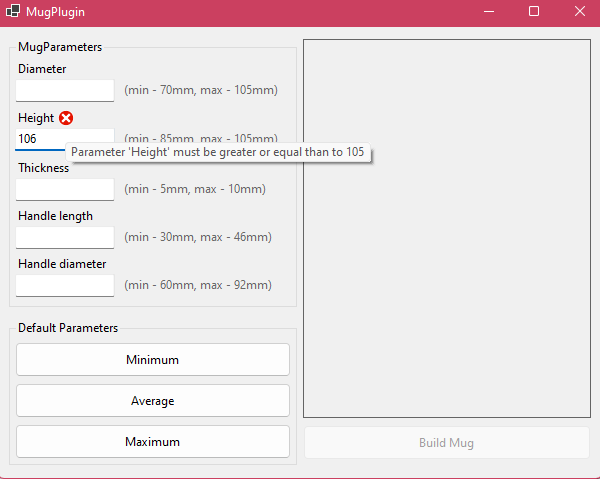


Рисунок 3.3 — Уведомление об ошибке

# 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. Mug3D. [Электронный ресурс]. — Режим доступа: https://mug3d.com/ru/?model=1 (дата обращения 13.10.2022).

5. Кружка — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D1%83%D0%B6%D0%BA%D0%B0 (дата обращения 13.10.2022).

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