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)

ПРОЕКТ СИСТЕМЫ

на тему «Разработка плагина «Форма для льда»

для САПР «Компас-3D»

по дисциплине «Основы разработки САПР»

Выполнил:

студент гр. 589-1

\_\_\_\_\_\_\_А.А. Воротова

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Руководитель:

к.т.н, доцент каф. КСУП

\_\_\_\_\_\_\_ А.А. Калентьев

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Томск 2022

**Содержание**

[1 Описание САПР 3](#_heading=h.gjdgxs)

[1.1 Описание программы 3](#_heading=h.30j0zll)

[1.2 Описание API 4](#_heading=h.1fob9te)

[1.3 Обзор аналогов 9](#_heading=h.3znysh7)

[2 Описание предмета проектирования 11](#_heading=h.tyjcwt)

[3 Проект программы 13](#_heading=h.3dy6vkm)

[3.1 Диаграмма классов 13](#_heading=h.1t3h5sf)

[3.2 Макет пользовательского интерфейса 14](#_heading=h.4d34og8)

[Список использованных источников 16](#_heading=h.17dp8vu)

**1 Описание САПР**

**1.1 Описание программы**

КОМПАС-3D — система трехмерного проектирования, ставшая стандартом для тысяч предприятий, благодаря сочетанию простоты освоения и легкости работы с мощными функциональными возможностями твердотельного и поверхностного моделирования. Ключевой особенностью продукта является использование собственного математического ядра C3D и параметрических технологий, разработанных специалистами АСКОН. КОМПАС-3D обеспечивает поддержку наиболее распространенных форматов 3D-моделей (STEP, ACIS, IGES, DWG, DXF), что позволяет организовывать эффективный обмен данными со смежными организациями и заказчиками, использующими любые CAD / CAM / CAE-системы в работе [1].

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

Система «Компас-3D» включает следующие компоненты: система трехмерного твердотельного моделирования, универсальная система автоматизированного проектирования «Компас-График» и модуль формирования спецификаций. Ключевой особенностью «Компас-3D» является использование собственного математического ядра и параметрических технологий [2].

**1.2 Описание API**

Сегодня встречаются задачи, решение которых не реализованы в CAD-системах. Чаще всего это очень узкоспециализированные задачи, которые встречаются на каком-то конкретном предприятии или подотрасли. Для решения подобных задач можно использовать КОМПАС-3D как платформу и на базе него создать свое приложение, которое позволит автоматизировать решение таких задач. Для создания таких приложений в КОМПАС-3D есть открытый API.

API — это вспомогательный интерфейс разработчика программного обеспечения, позволяющий быстро создавать программы и компоненты к ним благодаря использованию готового набора функций, методов и процедур, представленных в максимально понятной и удобной для программиста форме [3].

Физически API-функции представляются в виде отдельного программного модуля, который динамически подключается извне к основному проекту в формате DLL-библиотеки.

В КОМПАС на данный момент существуют API двух версий: API 5 и API 7. Обе версии реализуют различные функции системы и взаимно дополняют друг друга. В основном, для создания полноценных подключаемых модулей достаточно методов и свойств интерфейсов API 5 [4].

В таблице 1.1 представлены интерфейсы, которые будут использованы при разработке библиотеки.

Таблица 1.1 – Интерфейсы, используемые при разработке

|  |  |
| --- | --- |
| 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 |

Продолжение таблицы 1.1.

|  |  |
| --- | --- |
| ksDocument3D | Интерфейс документа-модели |
| ksPart | Интерфейс детали или подсборки в составе сборки |
| ksBaseExtrusionDefinition | Интерфейс параметров основания - элемента выдавливания |
| ksCircularPartArrayDefinition | Интерфейс операции копирования по окружности |

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

Таблица 1.2 – Используемые методы интерфейса KompasObject

|  |  |  |
| --- | --- | --- |
| 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 1.3 – Используемые методы интерфейса ksEntity

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

Таблица 1.4 – Используемые методы интерфейса ksDocument2D

|  |  |  |
| --- | --- | --- |
| 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 1.5 – Описание входных параметров, используемых методов 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 1.6 – Используемые методы интерфейса ksSketchDefinition

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

Таблица 1.7 – Используемые методы интерфейса ksDocument3D

|  |  |  |
| --- | --- | --- |
| 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 1.8 – Описание входных параметров, используемых методов 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) |

Продолжение таблицы 1.8.

|  |  |  |
| --- | --- | --- |
| 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 1.9 – Используемые методы интерфейса ksPart

|  |  |  |
| --- | --- | --- |
| 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 1.10 – Описание входных параметров, используемых методов интерфейса ksPart

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

Таблица 1.11 – Используемые типы объектов в методах 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 1.12 – Используемые методы интерфейса ksBaseExtrusionDefinition

|  |  |  |
| --- | --- | --- |
| 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 1.13 – Описание входных параметров, используемых методов 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 1.14 – Используемые методы интерфейса ksCircularPartArrayDefinition

|  |  |  |
| --- | --- | --- |
| 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 1.15 – Описание входных параметров, используемых методов 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 |

**1.3 Обзор аналогов**

**Autodesk Inventor 2022**

Данная программа предоставляет указанные ниже возможности 3D модели в Autodesk Inventor 2022:

* Состояния модели;
* Свойства экземпляра: назначение свойств отдельным экземплярам компонентов;
* Усовершенствования тонированного вида: использование стиля освещения модели на чертеже;
* Параметры вида чертежа: извлечение вида камеры и 3D-аннотаций;
* Форматирование текста: объединены элементы управления диалогового окна.

На рисунке 1.1 представлен пользовательский интерфейс программы «Inventor» для построения детали.

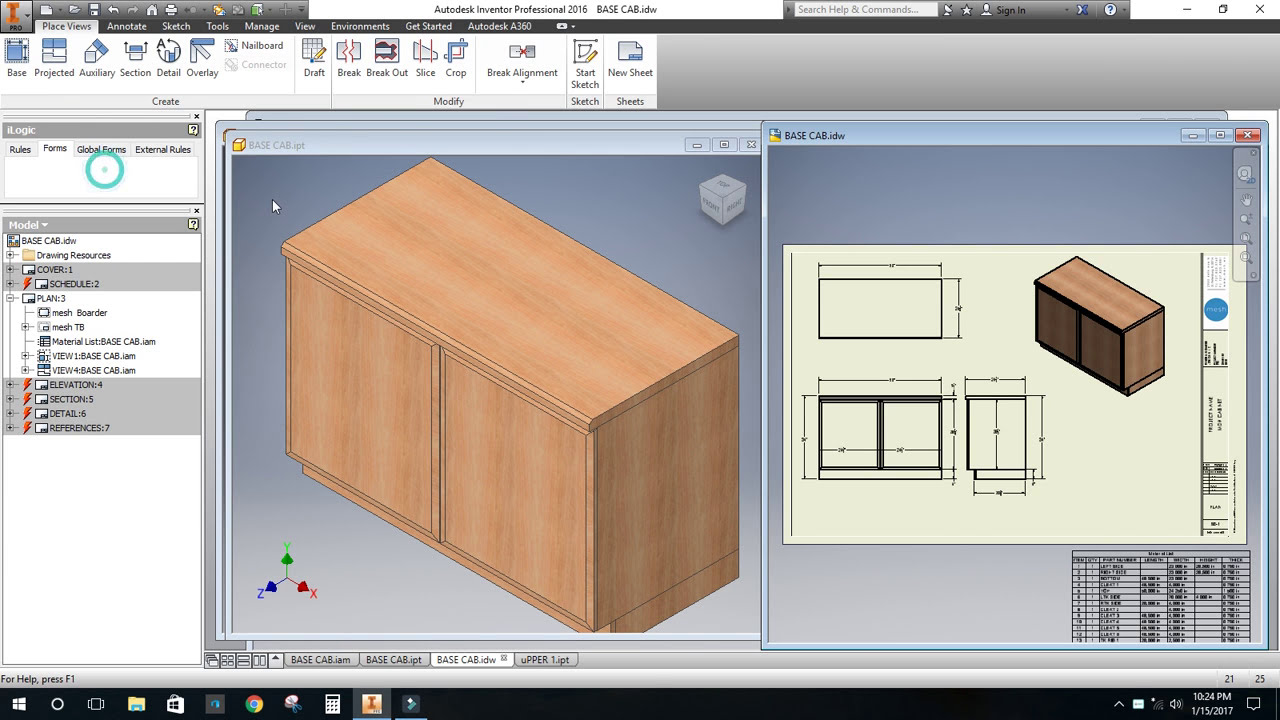


Рисунок 1.3.1 – Пользовательский интерфейс программы «Inventor» для построения ящика

**Solidworks 2022**

[SOLIDWORKS](https://my.solidworks.com/try-solidworks?lang=ru&utm_campaign=202112_rus_swks_TrySW_ru_terr&utm_medium=socialnetwork&utm_source=habr&utm_content=txt) - программное обеспечение, позволяющее решить целый комплекс задач, возникающих на производстве.

Инструментарий SOLIDWORKS позволяет не только сделать общий дизайн модели, но и:

– Провести детальное тестирование в виртуальной среде

– Доработать отдельные узлы

– Конвертировать модель в чертежи для создания реального прототипа

Таким образом, работа с проектом от стадии идеи до готового продукта осуществляется в рамках одного программного пакета. Поэтому SOLIDWORKS находит применение в самых разных отраслях.

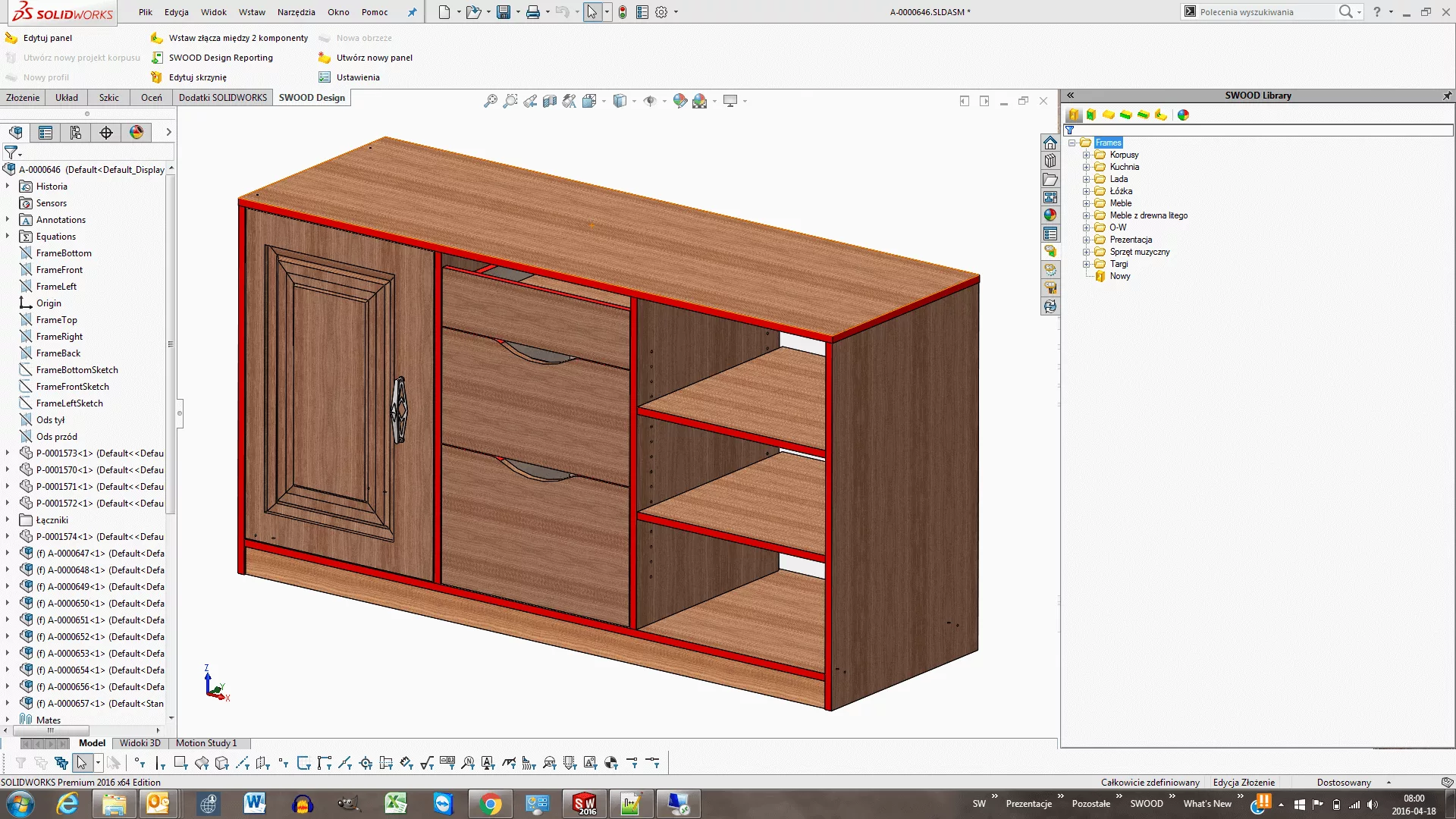


Рисунок 1.3.2 – Пользовательский интерфейс программы «Inventor» для построения ящика

**2 Описание предмета проектирования**

Предметом проектирования является форма для льда.

На рисунке 2.1 представлен чертеж форма для льда.

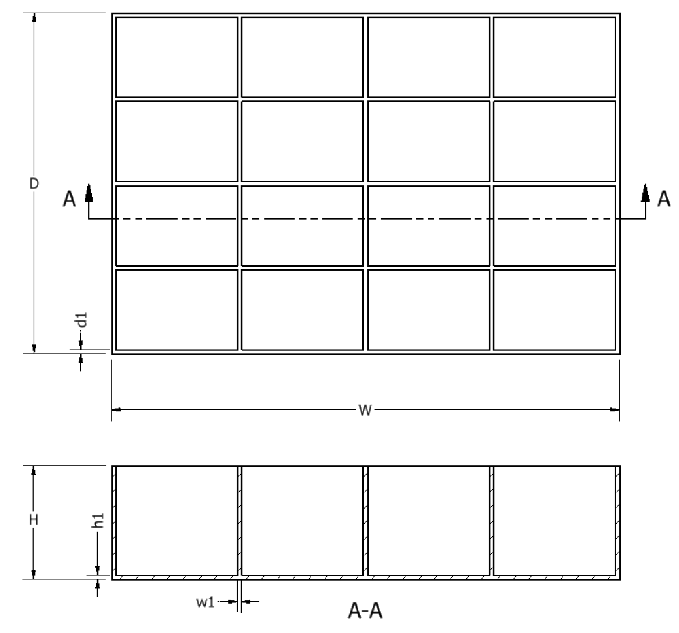


Рисунок 2.1 – чертеж формы для льда

На рисунке 2.2 представлена 3D-модель формы для льда

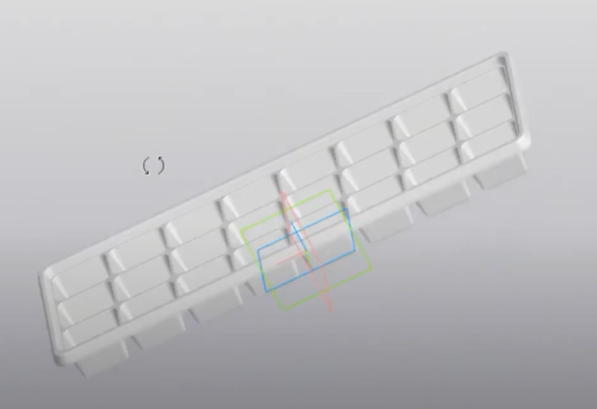


Рисунок 2.2 – 3D-модель формы для льда

**3 Проект программы**

**3.1 Диаграмма классов**

Унифицированный язык моделирования (UML) — это язык моделирования общего назначения, предназначенный для обеспечения стандартного способа визуализации проектирования системы/

Диаграмма классов UML представлена на рисунке 3.1.

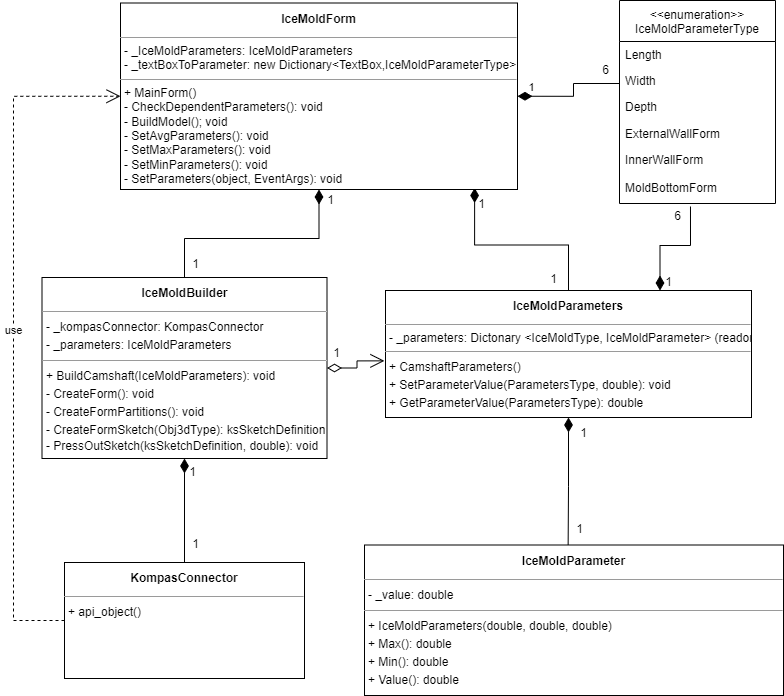


Рисунок 3.1 – Диаграмма классов UML

Далее в таблицах 3.1 – 3.4 представлено описание классов

Таблица 3.1 – Описание полей, методов, сущностей класса «IceMoldForm»

|  |  |  |
| --- | --- | --- |
| Название метода/поля | Возвращаемый тип | Описание |
| + \_parameters |  | Хранит в себе набор методов для построения формы для льда |
| - \_textBoxToParameter |  | Хранит в себе набор TextBox и соответствующий ему тип параметра из перечисления «IceMoldParametersType» |
| - SetParameter(object, EventArgs) | void | Устанавливает значение параметра |
| - SetMinParameters() | void | Устанавливает минимальное значение всех параметров |
| * SetMaxParameters() | void | Устанавливает максимальное значение всех параметров |
| - SetAvgParameters() | void | Устанавливает среднее значение всех параметров |
| - Build() | void | Строит форму по заданным параметрам |

Таблица 3.2 – Описание полей, методов, сущностей класса «IceMoldParameters»

|  |  |  |
| --- | --- | --- |
| Название метода/поля | Возвращаемый тип | Описание |
| - \_parameters |  | Хранит данные о каждом параметре модели из перечисления «IceMoldParametersType» |
| + IceMoldParameters() |  | Конструктор для создания экземпляра класса |
| +SetParameterValue(ParameterType, double) | void | Устанавливает значение определённого параметра |

Продолжение таблицы 3.2 – Описание полей, методов, сущностей класса «IceMoldParameters»

|  |  |  |
| --- | --- | --- |
| +GetParameterValue(ParameterType) | double | Возвращает значение определённого параметра |

Таблица 3.3 – Описание полей, методов, сущностей класса «IceMoldParameter»

|  |  |  |
| --- | --- | --- |
| Название метода/поля | Возвращаемый тип | Описание |
| -\_value |  | Хранит текущее значение |
| * IceMoldParameter(double, double, double) |  | Конструктор для создания экземпляра класса |
| + Max() | double | Возвращает максимальное допустимое значение параметра |
| + Min() | double | Возвращает минимальное допустимое значение параметра |
| + Value() | double | Возвращает текущее значение параметра. Задает новое значение параметра |

Таблица 3.4 – Описание полей, методов, сущностей класса «IceMoldBuilder»

|  |  |  |
| --- | --- | --- |
| Название метода/поля | Возвращаемый тип | Описание |
| * kompasConnector:KompasConnector |  | Хранит в себе методы необходимые для связи с КОМПАС 3D |
| * \_parameters: IceMoldParameters |  | Хранит данные о каждом параметре модели из перечисления «IceMoldParameter» |
| + IceMoldBuild(IceMoldParameters) | void | Построение пивной кружки по заданным параметрам |
| * CreateForm() | void | Построение основы формы |

Продолжение таблицы 3.4 – Описание полей, методов, сущностей класса «IceMoldBuilder»

|  |  |  |
| --- | --- | --- |
| * CreateFormPartitions() | void | Построение перегородок |
| - CreateFormSketch(obj3dType) | ksSketchDefinition | Возвращает интерфейс параметров эскиза |
| -PressOutSketch(ksSketchDefinition, double) | void | Выдавливает эскиз |

**3.2 Макет пользовательского интерфейса**

Пользовательский интерфейс – интерфейс, обеспечивающий передачу информации между пользователем – человеком и программно-аппаратными компонентами компьютерной сети [10].

Плагин представляет собой пользовательскую форму с полями для ввода соответствующих параметров (рисунок 3.1). Посредством кнопки «Построить форму для льда» осуществляется запуск САПР «Компас-3D», на рабочей области которой строится трехмерная модель по заданным параметрам. Если построение модели выполняется несколько раз, то плагин не запускает несколько копий программы «Компас-3D», а создает в ней новый документ. В случае ввода значений параметров, не входящих в допустимый диапазон, поле для ввода окрашивается в красный цвет и выводится окно, информирующее пользователя о не корректности введенного значения, например, со следующим текстом: «Значение параметра введено некорректно: длина форма не должна быть менее 100мм и не более 150мм».

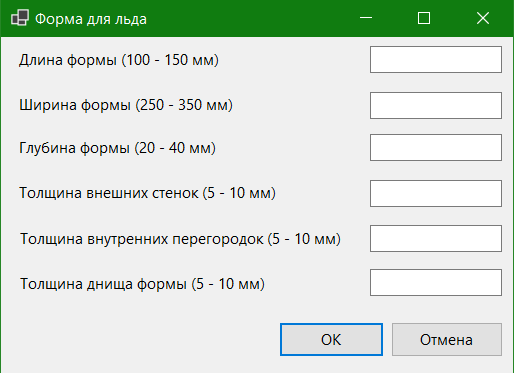


Рисунок 3.2.1 - Окно пользовательского интерфейса

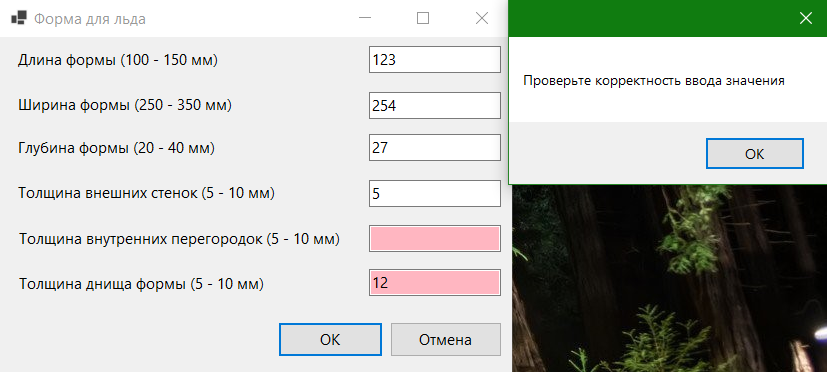


Рисунок 3.2.2 - Окно пользовательского интерфейса в динамике с отображением ошибок

**Список использованных источников**

1. Официальный сайт системы автоматизированного проектирования «Компас-3D» [Электронный ресурс] – Режим доступа: https://kompas.ru/kompas-3d/about (дата обращения: 07.10.2022)
2. Компас (САПР) [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/Компас\_(САПР) (дата обращения: 27.02.2020)
3. Интерфейс прикладного программирования геометрического ядра C3D. Его применение и главное отличие от API системы КОМПАС-3D [Электронный ресурс] – Режим доступа: https://sapr.ru/article/25210 (дата обращения: 07.10.2022)
4. Кидрук Максим. КОМПАС-3D V10 на 100% / М. Кидрук. – СПб.: Питер, 2009. – 560 с.
5. Программа автоматического построения 3D моделей и разверток по заданным значениям в AutoCAD «Inventor». [Поддержка и обучение](https://knowledge.autodesk.com/ru/support/inventor) [Электронный ресурс] – Режим доступа: https://knowledge.autodesk.com/ru/support/inventor/learn-explore/caas/CloudHelp/cloudhelp/2022/RUS/Inventor-WhatsNew/files/GUID-BAF6958D-EC21-4CC9-8A6A-0440661D77C0-htm.html (дата обращения: 13.10.2022).
6. Программа автоматического построения 3D моделей и разверток по заданным значениям в SOLIDWORKS. [Электронный ресурс] – Режим доступа: https://kb20.ru/solidworks/3d.php (дата обращения: 15.10.2022).
7. UML. Основы / Фаулер, М. – 3-е изд., пер. с англ. – СПб: Символ-Плюс, 2004. – 192 с.