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

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**РАЗРАБОТКА ПЛАГИНА «ЧАЙНИК ДЛЯ «KOMPAS-3D»**

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил:

студент гр. 581

Хайбулин Н.С.

« » 2024 г.

Руководитель:

к.т.н., доцент каф. КСУП

Калентьев А.А.

« » 2024 г.

**ОГЛАВЛЕНИЕ**

[1 ОПИСАНИЕ САПР 3](#_Toc179815024)

[1.1 Информация о выбранной САПР 3](#_Toc179815025)

[1.2 Описание API 3](#_Toc179815026)

[1.3 Обзор аналогов плагина 4](#_Toc179815027)

[2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ 6](#_Toc179815028)

[3 ПРОЕКТ СИСТЕМЫ 7](#_Toc179815029)

[3.1 Диаграмма классов 7](#_Toc179815030)

[3.2 Макеты пользовательского интерфейса 10](#_Toc179815031)

[4 СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 12](#_Toc179815032)

# ОПИСАНИЕ САПР

* 1. Информация о выбранной САПР

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

Данная САПР 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].

Прямыми аналогами «КОМПАС-3D» являются САПР «Autodesk Inventor» и «SolidWorks». К косвенным аналогам можно отнести «TinkerCAD» и «Micro–Cap».

* 1. Описание API

API (сокр. от «Application Programming Interface») – это программный интерфейс приложения, который предоставляет разработчику набор функций вместе с описанием того, что эти функции делают [2].

В «Компас–3D» API существуют двух версий: API 5 и API 7. Обе версии реализуют различные функции системы и взаимодополняют друг друга. Отсюда, обе версии программных интерфейсов в равной мере поддерживаются и развиваются с учетом самих изменений в системе. Для реализации создания плагина «Чайник» будет использоваться API «Компас–3D» версии 7 [3].

Основные используемые свойства и методы API «Компас–3D» расписаны в таблицах 1.1 – 1.6.

Таблица 1.1 – Используемые свойства интерфейса «IApplication».

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

Таблица 1.2 – Используемые методы интерфейса «IApplication».

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| CreateDocument | – | IKompasDocument | Создает новый документ |
| OpenDocument | String, filePath | IKompasDocument | Открывает существующий документ по указанному пути |
| CloseDocument | IKompasDocument, document | Void | Закрывает указанный документ |
| ActivateDocument | IKompasDocument, document | Void | Активирует указанный документ |

Таблица 1.3 – Используемые методы класса «IKompasDocument3D»

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| RebuildDocument | BOOL Result | HRESULT | Перестроить документ |

Таблица 1.4 – Используемые свойства класса «ILine»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Angle | Double | Угол между линией и осью OX в текущей системе координат |
| X1 | Double | Координата первой точки по оси X |
| X2 | Double | Координата второй точки по оси X |
| Y1 | Double | Координата первой точки по оси Y |
| Y2 | Double | Координата второй точки по оси Y |

Таблица 1.5 – Используемые свойства класса «IPoint»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Angle | Double | Угол для точки со стрелкой |
| Style | Long | Стиль точки |
| X | Double | Координата точки по оси X |
| Y | Double | Координата точки по оси Y |

Таблица 1.6 – Используемые свойства класса «IExtrusion»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Depth | Double | Глубина выдавливания |
| Direction | DirectionTypeEnum | Тип направления выдавливания |
| ExtrusionType | EndTypeEnum | Тип выдавливания |
| Sketch | ISketch | Эскиз |

* 1. Обзор аналогов плагина

«DriveWorks» – это плагин для «SolidWorks», который используется для автоматического создания параметрических моделей [4]. Он позволяет пользователю настраивать различные параметры моделей через интерфейс, а также автоматически генерирует 3D–модели на основе введённых данных. «DriveWorks» поддерживает автоматизированную валидацию параметров и создание чертежей. Интерфейс данного плагина представлен на рисунке 1.1.



Рисунок 1.1 – Интерфейс плагина «DriveWorks»

Вторым аналогом является плагин «EasyKitchen PRO». Он содержит в себе модули и компоненты для проектирования кухонной и корпусной мебели в программе «SketchUp», плагин замены компонентов и текстур, инструменты для генерации отчёта и создания деталировки и сметы. Интерфейс программы представлен на рисунке 1.2.



Рисунок 1.2 – Интерфейс плагина «EasyKitchen PRO»

# ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ

Чайник – это кухонный прибор, который представляет из себя сосуд, используемый для кипячения воды. Он состоит из нескольких элементов, включая металлический корпус, крышку, ручку, носик и дно [5, 6]. Чертёж чайника представлен на рисунке 2.1.



Рисунок 2.1 – Чертёж чайника

Изменяемые параметры для предмета проектирования (также все

обозначения показаны на рисунке 2.1):

− Диаметр дна чайника d1 (от 100 до 400мм);

− Диаметр крышки чайника d2 (от 75 до 300мм, но не больше дна d1);

− Высота чайника h1 (от 80 до 450мм);

− Высота ручки h2 (от 70 до 150мм, но не больше высоты чайника);

− Объём чайника (от 0.63 до 56.55л) – может задаваться пользователем, как и d1 или h1 и недостающий третий параметр будет автоматически заполняться в соответствии с ограничениями;

− Цвет чайника.

1. **ПРОЕКТ СИСТЕМЫ**

## 3.1 Диаграмма классов

UML–диаграмма классов – тип статической структурной диаграммы, описывающей структуру системы посредством обозначения классов, их атрибутов, методов, связей на диаграмме [7].

На рисунке 3.1 отображена диаграмма классов приложения.

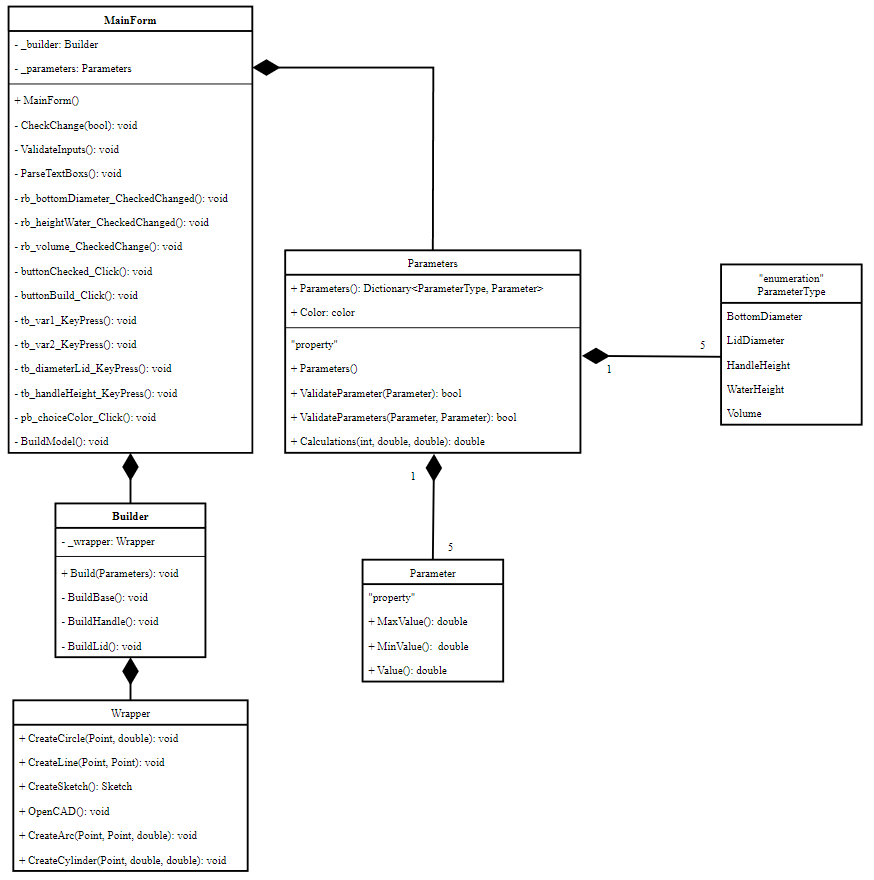


Рисунок 3.1 – Пример UML-диаграммы классов

В таблицах ниже представлена информация о свойствах и методах каждого из классов (таблицы 3.1–3.8).

Таблица 3.1 − Свойства класса «MainForm»

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

Таблица 3.2 − Методы класса «MainForm»

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| MainForm | − | − | Конструктор главной формы |
| CheckChange | bool | − | Функция для изменения полей в зависимости от выбранной радиокнопки |
| ValidateInputs | − | − | Валидация для заполненных полей |
| ParseTextBoxs | − | − | Запись значений из текстовых полей |
| rb\_diameterBottom\_ CheckedChange | − | − | Запуск проверки данных, если выбран «диаметр» |
| rb\_heightWater\_ CheckedChanged | − | − | Запуск проверки данных, если выбран «высота» |
| rb\_volume\_Checked Changed | − | − | Запуск проверки данных, если выбран «объем» |
| buttonChecked\_Click | − | − | Расчет третьего параметра по нажатию кнопки |
| buttonBuild\_Click | − | − | Построить модель по нажатию кнопки |
| tb\_var1\_KeyPress | − | − | Ограничение ввода в текстовое поле первого параметра |
| tb\_var2\_KeyPress | − | − | Ограничение ввода в текстовое поле второго параметра |
| tb\_diameterLid\_ KeyPress | − | − | Ограничение ввода в текстовое поле диаметра крышки |
| tb\_handleHeight\_ KeyPress | − | − | Ограничение ввода в текстовое поле высоты ручки |
| pb\_choiceColor\_Click | − | − | Выбор цвета |
| BuildModel | − | − | Функция запуска построения модели по заданным параметрам |

Таблица 3.3 − Свойства класса «Parameters»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Parameters | Dictionary<ParameterType, Parameter> | Хранит в себе словарь с параметрами объекта построения |
| Color | color | Переменная для хранения цвета, отдельно от числовых переменных |

Таблица 3.4 – Методы класса «Parameters»

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Parameters | − | − | Конструктор класса Parameters |
| ValidateParameter | Parameter | bool | Валидация по диапазону значений параметра |
| ValidateParameters | Parameter, Parameter | bool | Зависимая валидация параметра относительно другого параметра |
| Calculations | int, double, double | double | Функция для расчета третьего параметра по двум другим, относительно выбранной радиокнопки |

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 | Тип возвращаемых данных | Описание |
| Build | Parameters | − | Построение модели по заданным параметрам |
| BuildBase | − | − | Построение основания чайника |
| BuildHandle | − | − | Построение ручки чайника |
| BuildLid | − | − | Построение крышки чайника |

Таблица 3.7 − Свойства класса «Parameter»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| MaxValue | double | Максимально допустимое значение параметра |
| MinValue | double | Минимально допустимое значение параметра |
| Value | double | Значение параметра |

Таблица 3.8 − Методы класса «Wrapper»

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| CreateCircle | Point, double | − | Создание дуги по двум точкам |
| CreateLine | Point, Point | − | Создание линии |
| CreateSketch | − | − | Создание эскиза |
| OpenCAD | − | − | Открытие Компас3D |
| CreateArc | Point, Point, double | − | Создание дуги |
| CreateCylinder | Point, double, double | − | Создание цилиндра |

* 1. Макеты пользовательского интерфейса

Пример макета пользовательского интерфейса представлен на рисунке 3.2.

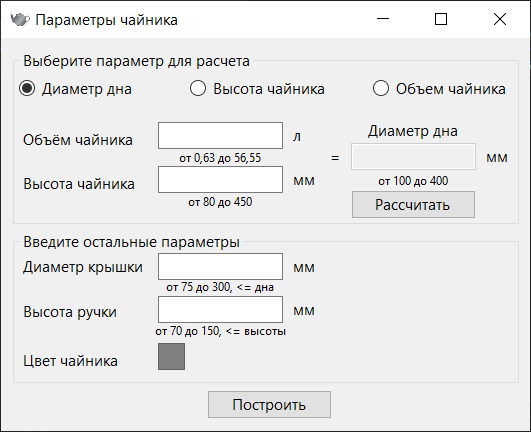


Рисунок 3.2 – Макет пользовательского интерфейса

Основой реализации макета пользовательского интерфейса стал выбор одной из трех радиокнопок – «диаметр дна», «высота чайника», «объем чайника», при изменении параметра для расчета, поля для ввода значений меняются местами. Изменение полей представлены на рисунках 3.3–3.5.

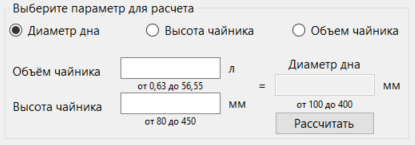


Рисунок 3.3 – Поля при выборе параметра «диаметр дна»

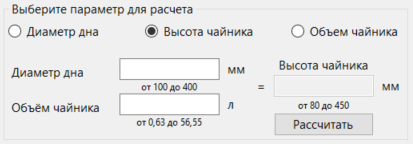


Рисунок 3.4 – Поля при выборе параметра «высота чайника»

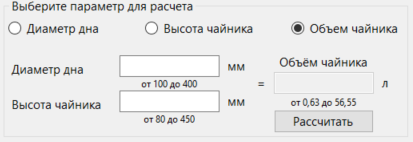


Рисунок 3.5 – Поля при выборе параметра «объем чайника»

После заполнения двух основных полей пользователь способен найти третий параметр по одной из трех формул, которая применяется по нажатию на кнопку «рассчитать». Формулы в коде представлены на рисунке 3.6.

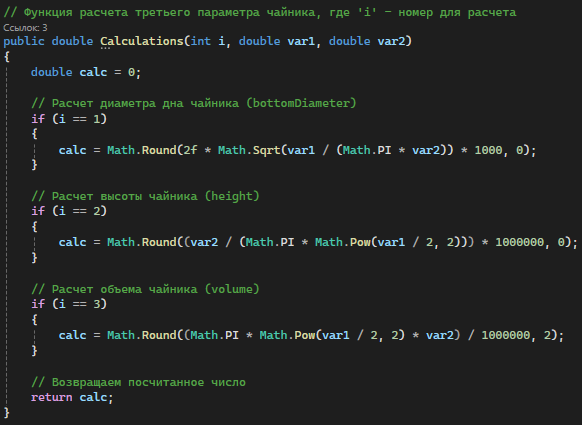


Рисунок 3.7 – Реализация функции для расчета третьего параметра

Для проверки валидации некорректных данных необходимо представить возможные действия пользователя с данным интерфейсом. Предположим, что пользователь ввел корректные данные в нужном диапазоне, но зависимые параметры, которые должны быть меньше или равны другому оказываются куда больше, чем нужно. Пример таких действий показан на рисунке 3.8.

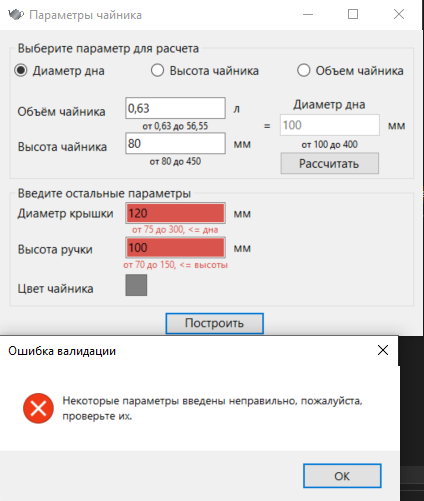


Рисунок 3.8 – Зависимая валидация параметров

Предположим, что пользователь ввел параметры, которые выходят за диапазон значений, что показано на рисунке 3.9.

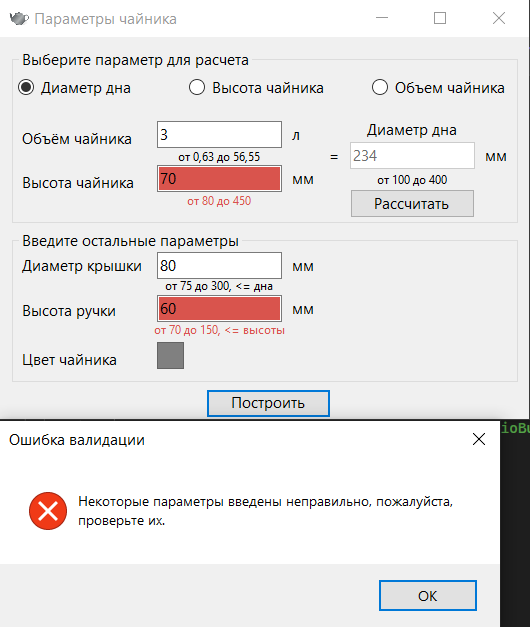


Рисунок 3.9 – Диапазонная валидация значений

Также, пользователь мог ввести некорректные данные для вычисления третьего параметра по формуле, на рисунке 3.10 продемонстрирована работа валидации на третьем расчетном параметре.

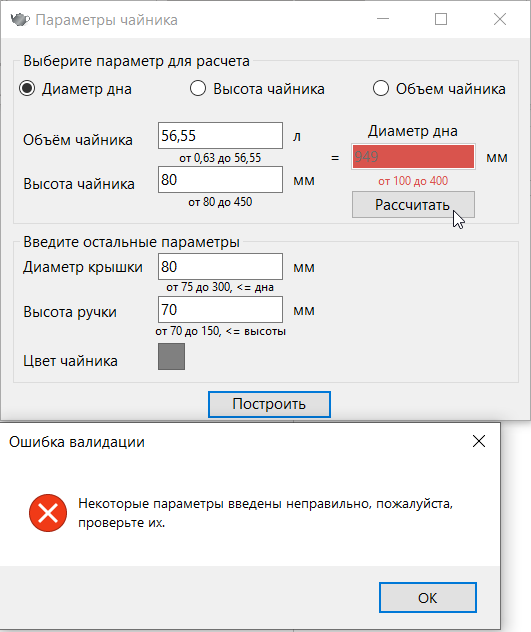


Рисунок 3.10 – Валидация третьего расчетного параметра

Валидация для проверки формата данных не требуется, так как все поля принимают только цифры и запятую для ввода (события «KeyPress» и условий для проверки введенного символа).

# СПИСОК ИСПОЛЬЗУЕМЫХ 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://ravesli.com/what-is-an-api/> (дата обращения 30.09.2024);
3. Справочная система SDK КОМПАС–3D – API интерфейсов версия 7 [Электронный ресурс] – Режим доступа: <https://help.ascon.ru/KOMPAS_SDK/22/ru-RU/applicate.html> (дата обращения 08.10.2024)
4. DriveWorks [Электронный ресурс] − Режим доступа: <https://www.driveworks.co.uk/> (дата обращения 08.10.2024);
5. EasyKitchen Pro [Электронный ресурс] − Режим доступа: <https://redkit.pro/easykitchen/easykitchen-pro/> (дата обращения 08.10.2024);
6. ГОСТ 24788−2018 «Посуда хозяйственная стальная эмалированная. Общие технические условия».
7. Фаулер M. UML. Основы, 3-е издание. – Пер. с англ. – СПб: СимволПлюс, 2004. – 192 с.