Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

РАЗРАБОТКА ПЛАГИНА «ТОРМОЗНОЙ ДИСК» ДЛЯ «КОМПАС-3D»

Пояснительная записка по дисциплине

«Основы разработки САПР»

Студент гр. 580-3 ㅤㅤㅤㅤ ㅤ

ㅤㅤㅤㅤ Д. Е. Хайрулин ㅤ

«ㅤㅤ» ㅤㅤㅤㅤㅤㅤㅤㅤ ㅤ

Проверил ㅤㅤㅤㅤㅤㅤ

Доцент кафедры КСУП, к.т.н.

ㅤㅤㅤㅤ ㅤㅤㅤㅤㅤ А. А. Калентьев ㅤ

оценка ㅤㅤㅤㅤㅤㅤㅤㅤㅤㅤㅤㅤㅤ ㅤ

«ㅤㅤ» ㅤㅤㅤㅤㅤㅤㅤㅤ ㅤ

Томск 2023

**Содержание**

[1 ОПИСАНИЕ САПР 3](#_yp6x5vic8s2c)

[1.1 Информация о выбранной САПР 3](#_tms24xr082fj)

[1.2 Описание API 4](#_f4cytl4jgq7)

[2 ПРЕДМЕТ ПРОЕКТИРОВАНИЯ 9](#_dr17vgkqr599)

[2.1 Описание предмета проектирования 9](#_hqxr7728ym1)

[2.2 Обзор аналогов плагина 10](#_4y78zd6tjjyu)

[3 ПРОЕКТ СИСТЕМЫ 11](#_pu2wh2rpgsd2)

[3.1 Диаграмма классов 11](#_to6hlteymjwr)

[3.2 Макет пользовательского интерфейса 13](#_5b59tqor801v)

[Список использованных источников 15](#_mywwljpud18f)

# 1 ОПИСАНИЕ САПР

## 1.1 Информация о выбранной САПР

КОМПАС-3D - это компьютерная система автоматизированного проектирования в трехмерном пространстве, разработанная компанией ASCON.[1]

Основные возможности КОМПАС-3D включают:

1. Создание и редактирование трехмерных моделей объектов при помощи специализированных инструментов, что включает в себя возможность проектирования сложных деталей и сборок с высокой степенью точности.
2. Генерация и редактирование технической документации, такой как чертежи, схемы, спецификации и другие документы, необходимые для производства и понимания проекта.
3. Совместимость и обмен данными с другими программами и форматами файлов, включая AutoCAD, STEP, IGES, и другие, обеспечивая возможность интеграции с другими системами и обмена информацией с коллегами и партнерами.

Аналогами КОМПАС-3D могут являться следующие САПР:

1. AutoCAD: AutoCAD является одной из самых распространенных САПР. Он обладает широкими возможностями как в области двухмерного, так и трехмерного моделирования, а также создания технической документации. Он популярен в различных индустриях, КОМПАС-3D, прежде всего, используется только для промышленных и инженерных задач.[2]
2. SolidWorks: SolidWorks - это САПР, специализирующаяся на трехмерном моделировании и инженерном анализе. Эта система также широко применяется в разработке и проектировании продуктов, особенно в машиностроении и промышленной отрасли. Является более сложным в освоении, если сравнивать с КОМПАС-3D.[3]

Выбор САПР КОМПАС-3D обусловлен следующими факторами:

1. Простота использования: КОМПАС-3D известен своим относительно простым интерфейсом и интуитивно понятными инструментами.
2. Фокус на российском рынке: КОМПАС-3D разработан в России, и у него есть специфические функции и стандарты, учитывающие потребности российских предприятий и инженеров.

## 1.2 Описание API

API (Application Programming Interface) - это набор правил и протоколов, который позволяет различным программам и компонентам взаимодействовать друг с другом. API определяет способы, как программы могут отправлять запросы, получать данные и выполнять определенные действия.[4]

Для КОМПАС-3D существует API, который позволяет разработчикам создавать свои собственные приложения и интегрировать их с функциональностью КОМПАС-3D. Это позволяет автоматизировать процессы проектирования, создавать дополнения и расширения для программы, а также интегрировать КОМПАС-3D с другими приложениями и системами. API КОМПАС-3D предоставляется в виде библиотеки или набора инструментов и функций, которые могут быть использованы разработчиками. Эти инструменты обеспечивают доступ к функциям КОМПАС-3D, таким как создание и редактирование моделей, работа с чертежами, расчеты и другие операции. Чтобы начать работу с API КОМПАС-3D, разработчику следует ознакомиться с документацией, предоставленной производителем КОМПАС-3D, которая описывает доступные функции и методы, а также предоставляет примеры использования. Далее, разработчик может использовать язык программирования (например, C++, C#, или Python), чтобы написать приложение, которое будет взаимодействовать с КОМПАС-3D через API. Применение API позволяет сделать КОМПАС-3D более гибким и адаптируемым к конкретным потребностям пользователя, а также упростить процессы проектирования и моделирования.

В таблицах ниже представлены основные методы и свойства интерфейсов KompasObject, IPart и ksDocument3D, ksDocument2D, [ksEntity](about:blank).

Таблица 1.1 - Методы интерфейса KompasObject

| Свойство | Возвращаемое значение | Описание |
| --- | --- | --- |
| Document3D | Указатель на интерфейс документа трехмерной модели ksDocument3D. | Дает возможность получить указатель на интерфейс трехмерного документа (детали или сборки). |
| ActivateControllerAPI | - | Метод для активации контроллера API. |
| ksDocument2D() | - | Интерфейс событий графического документа, события интерфейса позволяют контролировать состояние документа. |

Таблица 1.2 - Методы интерфейса IPart

| Метод | Входные параметры | Возвращаемое значение | Описание |
| --- | --- | --- | --- |
| GetDefaultEntity(short objType) | objType – тип объекта. | Указатель на интерфейс [ksEntity](about:blank) или [IEntity](about:blank). | Получить указатель на интерфейс объекта, создаваемого системой по умолчанию. |
| GetPart(int type) | type – тип объекта. | Указатель на интерфейс компонента [ksPart](about:blank) или [IPart](about:blank). | Получить указатель на интерфейс компонента в соответствии с заданным типом. |

Окончание таблицы 1.2

| Метод | Входные параметры | Возвращаемое значение | Описание |
| --- | --- | --- | --- |
| NewEntity(short objType) | objType – тип объекта. | Указатель на интерфейс [ksEntity](about:blank) или [IEntity](about:blank). | Создать новый интерфейс объекта и получить указатель на него. |

Таблица 1.3 – Методы интерфейса ksDocument3D

| Метод | Входные параметры | Возвращаемое значение | Описание |
| --- | --- | --- | --- |
| Create (bool invisible, bool \_typeDoc) | invisible –  (TRUE – невидимый режим,  FALSE – видимый режим),  typeDoc –  (TRUE – деталь,  FALSE – сборка). | TRUE – в случае успешного завершения. | Дает возможность создать пустой документ (деталь или сборку) |
| GetPart(int type) | type – тип компонента из перечисления Типы компонентов. | - | Получить указатель на интерфейс компонента в соответствии с заданным типом |

Таблица 1.4 - Методы интерфейса ksDocument2D

| Метод | Входные параметры | Возвращаемое значение | Описание |
| --- | --- | --- | --- |
| long ksLineSeg (double x1, double y1, double x2, double y2, long style) | x1, y1 - координаты первой точки отрезка, x2, y2 - координаты | Указатель на отрезок - в случае удачного завершения, 0 - в случае неудачи. | Метод для создания отрезка. |

Окончание таблицы 1.4

| Метод | Входные параметры | Возвращаемое значение | Описание |
| --- | --- | --- | --- |
|  | второй точки отрезка, style - стиль линии. |  |  |

Таблица 1.5 - Методы интерфейса [ksEntity](about:blank)

| Метод | Входные параметры | Возвращаемое значение | Описание |
| --- | --- | --- | --- |
| ksBossExtrusionDefinition(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. | TRUE – в случае успешного завершения. | Метод выдавливает эскиз в одном направлении |

Таблица 1.6 – Используемые свойства класса Application

| Название | Тип данных | Описание |
| --- | --- | --- |
| Model | ksModel | 3D-модель детали. Возвращает объект ksModel, представляющий 3D-модель детали. |

Окончание таблицы 1.6

| Название | Тип данных | Описание |
| --- | --- | --- |
| VariableTable | ksVariableTable | Таблица переменных и параметров детали. Возвращает объект ksVariableTable, который позволяет управлять переменными и параметрами детали. |

# 2 ПРЕДМЕТ ПРОЕКТИРОВАНИЯ

## 2.1 Описание предмета проектирования

Предметом проектирования является тормозной диск. Тормозной диск или же дисковый тормозной механизм или дисковые тормоза — основной элемент дисковой тормозной системы. Представляющий из себя вращающаяся часть дисковой системы, к которой при помощи привода прижимаются неподвижные тормозные колодки.

Назначение разрабатываемого плагина обусловлено быстрым моделированием тормозных дисков разных типов и размеров. Благодаря данному расширению, каждый тормозной диск можно создать индивидуально под каждого клиента. На рисунке 2.1 представлена модель тормозного диска с параметрами доступными для изменения.

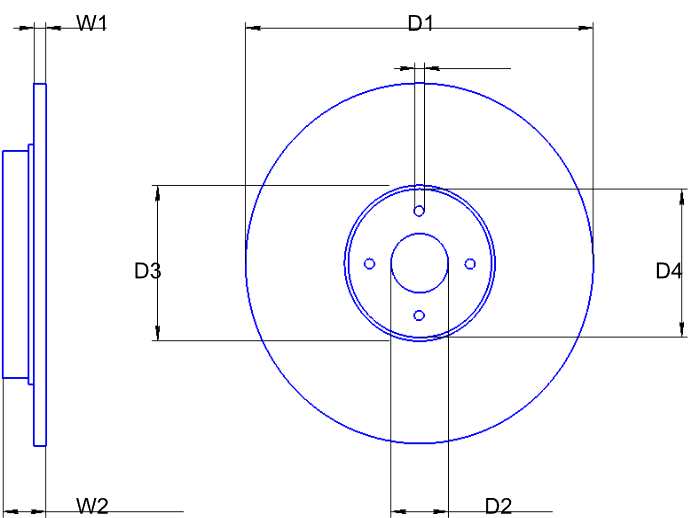


Рисунок 2.1 - Модель тормозного диска

Изменяемые параметры для плагина(также все обозначения показаны на рис. 2.1):

* W1 - Ширина рабочей поверхности диска (10 - 20 мм)
* W2 - Толщина тормозного диска (3/1 - 4/1 от W1)
* D1 - Диаметр тормозного диска (300 - 400 мм)
* D2 - Диаметр центрирования (1/3 - 1/4 от D1)
* D3 - Больший диаметр крепежа (не более 1/2 от D1, 120 - 200 мм)
* D4 - Малый диаметр крепежа (рассчитывается по формуле (D3 - (D3 \* 0,03))

## 

## 2.2 Обзор аналогов плагина

Косвенным аналогом является плагин GearTeq для SolidWorks. Плагин предназначен для автоматического проектирования шестерен различных типов, однако, шестерни и тормозные диски имеют некоторые сходства в формах, что может быть полезно при разработки плагина предназначенного для построения тормозных дисков.[5]

# 3 ПРОЕКТ СИСТЕМЫ

## 3.1 Диаграмма классов

Диаграммы UML (Unified Modeling Language) представляют собой графические инструменты, используемые для моделирования и визуализации различных аспектов системы или программного обеспечения. Они позволяют разработчикам и аналитикам представить идеи, концепции и структуру системы, а также её поведение и взаимодействие с внешним окружением. Диаграммы UML широко используются в разработке программного обеспечения и помогают в понимании, проектировании и документировании системы.[6]

На рисунке 3.1 представлена диаграмма классов:

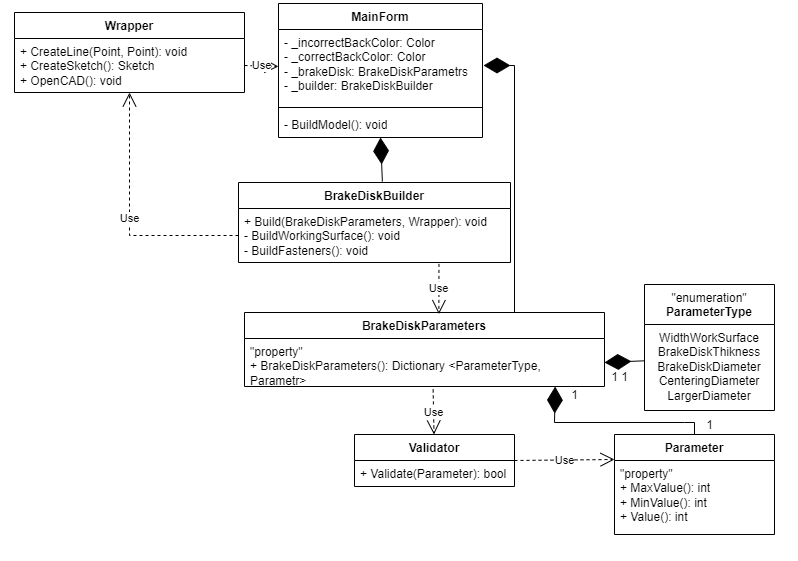


Рисунок 3.1 - Диаграмма классов

Разбор основных классов:

MainForm - является главным окном приложения. Хранит в себе параметры (Parameters) и объект класса строителя модели (Builder);

Parameter - класс, хранящий в себе параметры модели;

BrakeDiskBuilder - класс строитель модели;

Wrapper - класс обёртка API КОМПАС. В нем находятся все нужные методы создания примитивов и документов, которые пригодятся для построения модели.

Таблица 3.1 - Свойства класса Parameter

| Название | Тип данных | Описание |
| --- | --- | --- |
| MaxValue | int | Максимальное значение параметра |
| MinValue | int | Минимальное значение параметра |
| Value | int | Значение параметра |

Таблица 3.2 - Методы класса Validator

| Название | Входные параметры | Тип возвращаемых данных | Описание |
| --- | --- | --- | --- |
| Validate | Parameter | bool | Проверяет параметр на соответствие ограничений |

Таблица 3.3 - Свойств класса BrakeDiskParameters

| Название | Тип данных | Описание |
| --- | --- | --- |
| BrakeDiskParameters | Dictionary | Коллекция объектов класса Parameter, внутри которого описаны ограничения и само значение |

Таблица 3.4 - Методы класса BrakeDiskBuilder

| Название | Входные параметры | Тип возвращаемых данных | Описание |
| --- | --- | --- | --- |
| Build | BrakeDiskParameters, Wrapper | void | Построение полной модели диска. |
| BuildWorkingSurface | - | void | Построение рабочей поверхности |
| BuildFasteners | - | void | Построение крепежа |

Таблица 3.5 - Поля класса MainForm

| Название | Тип данных | Описание |
| --- | --- | --- |
| \_incorrectBackColor | Color | Цвет, который отображается при некорректном вводе |
| \_correctBackColor | Color | Цвет, который отображается при корректном вводе |
| \_brakeDisk | BrakeDiskParametrs | Экземпляр класса BrakeDiskParametrs |
| \_builder | BrakeDiskBuilder | Экземпляр класса BrakeDiskBuilder |

Таблица 3.6 - Методы класса MainForm

| Название | Входные параметры | Тип возвращаемых данных | Описание |
| --- | --- | --- | --- |
| BuildModel | - | void | Построение полной модели в Компас 3D |

## 3.2 Макет пользовательского интерфейса

Макет пользовательского интерфейса представляет собой форму для ввода параметров. При запуске программы в полях для ввода параметров отсутствуют значения. Пользователь может менять данные параметры (рисунок 3.2).

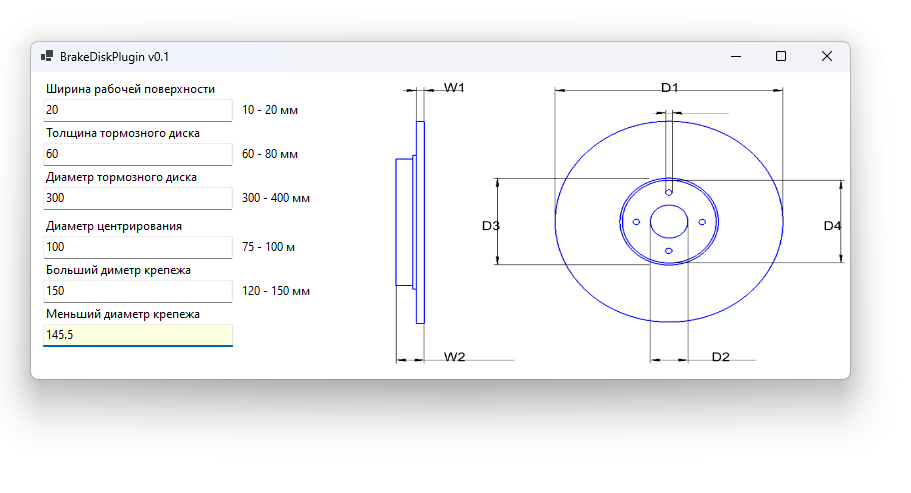


Рисунок 3.2 - Макет пользовательского интерфейса

Поле, где было введено некорректное значение изменит цвет на светло-красный (рисунок 3.3):

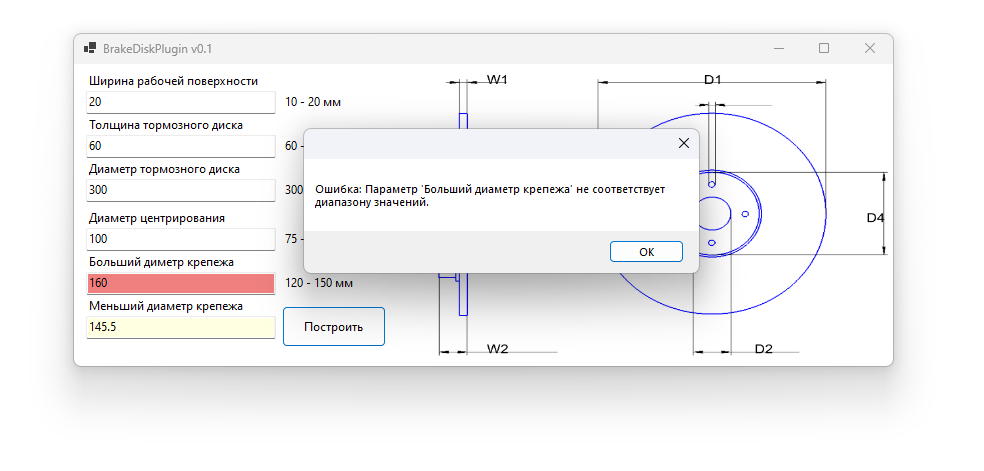


Рисунок 3.3 - Демонстрация обработки ошибок на макете

# Список использованных источников

**ЭЛЕКТРОННЫЕ РЕСУРСЫ**

**Электронные ресурсы удаленного доступа**

1. САПР КОМПАС-3D [Электронный ресурс]: Официальный сайт САПР КОМПАС. URL: <https://kompas.ru/> (дата обращения: 27.10.2023);
2. САПР Autodesk AutoCAD [Электронный ресурс]: Официальный сайт компании Autodesk. URL: <https://www.autodesk.com/products/autocad/> (дата обращения: 27.10.2023);
3. САПР SOLIDWORKS [Электронный ресурс]: Официальный сайт САПР SOLIDWORKS. URL: <https://www.solidworks.com/> (дата обращения: 27.10.2023);
4. Статья об API [Электронный ресурс]: Документация Amazon AWS. URL: <https://aws.amazon.com/ru/what-is/api/> (дата обращения: 27.10.2023)
5. Плагин для САПР SOLIDWORKS GearTeq [Электронный ресурс]: Официальный сайт компании Camnetics. URL: <https://camnetics.com/gearteq/> (дата обращения 27.10.2023);
6. Статья об UML [Электронный ресурс]: Статья на Хабр. URL: <https://habr.com/ru/articles/738428/> (дата обращения 27.10.2023).