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 «КИЯНКА» ДЛЯ САПР «КОМПАС 3D v18»

Проект системы по лабораторному проекту

по дисциплине «Основы разработки САПР»

Выполнил:

студент группы 588-3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Абдеев Т.В.

«\_\_» \_\_\_\_\_\_\_ 2021 г.

Руководитель:

к.т.н., доцент КСУП:

\_\_\_\_\_\_\_\_\_ Калентьев A. А.

«\_\_» \_\_\_\_\_\_\_\_2021 г.

# 1. **Описание САПР**

## **1.1. Описание программы**

Система автоматизированного проектирования (САПР) – это автоматизированная система, реализующая информационную технологию выполнения функций проектирования, представляет собой организационно техническую систему, предназначенную для автоматизации процесса проектирования, состоящую из комплекса технических, программных и других средств автоматизации его деятельности

«КОМПАС-3D» – это семейство универсальных систем трехмерного проектирования различных деталей и сборок, которое широко используется в строительстве и машиностроении благодаря сочетанию проектирования изделий любой сложности и простоты освоения. Особенностью продукта является использование российского геометрического ядра C3D и собственных программных технологий. Также «КОМПАС-3D» обеспечивает поддержку файлов наиболее распространённых CAD-систем, таких как Autodesk Inventor и SolidWorks, что позволяет организовывать совместную работу со смежными организациями и заказчиками, использующими другие программные продукты. [1]

## 1.2. Описание API

API (англ. Application Programming Interface) – описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой.

В «КОМПАС-3D» на данный момент существуют API двух версий: API 5 и API 7. Обе версии реализуют различные функции системы и взаимно дополняют друг друга. Обе версии программных интерфейсов в равной мере поддерживаются и развиваются с учетом самих изменений в системе.

Главным интерфейсом API системы КОМПАС является KompasObject. За получение указателя на этот интерфейс (фактически, на интерфейс приложения API 5) отвечает экспортная функция CreateKompasObject(). Методы этого интерфейса реализуют наиболее общие функции работы с документами системы, системными настройками, файлами, а также дают возможность получить указатели на другие интерфейсы (интерфейсы динамического массива, работы с математическими функциями, библиотек моделей или фрагментов и различных структур параметров определенного типа). [2]

Ниже в таблице 1.1 представлены свойства и методы интерфейса KompasObject, которые необходимы для разработки плагина.

Таблица 1.1 – Некоторые методы и свойства интерфейса KompasObject

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Document3D() |  | ksDocument | Метод для получения указателя на интерфейс трехмерного графического документа (детали или сборки) |
| GetParamStruct(short structType) | structType – тип интерфейса параметров | StructType2D | Метод для получения указателя на интерфейс графического документа (чертежа или фрагмента) |
| Visible |  | bool | Свойство видимости приложения |
| Quit() |  |  | Метод для закрытия активного окна приложения КОМПАС |

В таблице 1.2 представлены методы интерфейса ksEntity, которые будут использованы при разработке плагина

Таблица 1.2 – Методы интерфейса ksEntity

|  |  |  |
| --- | --- | --- |
| Название | Тип возвращаемых данных | Описание |
| Create() | bool | Создать объект в модели |
| GetDefinition() | IUnkown | Получить указатель на интерфейс параметров объектов и элементов |
| Update() | bool | Изменить свойства объекта (используя ранее установленные свойства) |

В таблице 1.3 представлены методы интерфейса ksDocument2D, которые будут использованы при разработке плагина

Таблица 1.3 – Некоторые методы интерфейса ksDocument2D

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| ksRectangle(  ksRectangleParam param, int style) | param – параметры прямоугольника.  style – стиль линии. | int | Получить указатель на прямоугольник на двумерной плоскости либо 0 в случае ошибки |
| ksCircle(  double xc, double yc, double rad, int style) | xc, yc - координаты центра окружности.  rad - радиус окружности.  style – стиль линии. | int | Получить указатель на окружность на двумерной плоскости либо 0 в случае ошибки |

В таблице 1.4 представлены методы интерфейса ksDocument3D, которые будут использованы при разработке плагина

Таблица 1.4 – Некоторые методы интерфейса ksDocument3D

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание | |
| Create (bool invisible, bool \_typeDoc) | invisible – признак режима редактирования документа (true – невидимый режим, false – видимый режим),  typeDoc – тип документа (true – деталь, false – сборка). | bool | Создать документ-модель (деталь или сборку) | |
| GetPart(int type) | type – тип компонента из перечисления Типы компонентов. | ksPart | | Получить указатель на интерфейс компонента в соответствии с заданным типом |

В таблице 1.5 представлены методы интерфейса ksPart, которые будут использованы при разработке плагина

Таблица 1.5 – Некоторые методы интерфейса ksPart

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| EntityCollection(short objType) | objType – тип объектов, содержащихся в массиве. | ksEnintyCollection | Формирует массив объектов и возвращает указатель на его интерфейс |
| GetDefaultEntity(short objType) | objType – тип объекта | ksEntity | Получить указатель на интерфейс объекта, создаваемого системой по умолчанию |
| GetPart(int type) | Type – тип компонента | ksPart | Получить указатель на интерфейс компонента в соответствии с заданным типом |
| NewEntity(short objType) | objType – тип объекта | ksEntity | Создать новый интерфейс объекта и получить указатель на него |

## 1.3. Обзор аналогов

**Библиотека «Инструмент» для Компас 3Д**

Сторонняя библиотека «Инструмент» обеспечивает быстрое вычерчивание различных инструментов в системе Компас-3D в соответствии с их точными размерами, согласно ГОСТ, ОСТ или ТУ и работает как отдельный модуль. Основные геометрические параметры инструментов хранятся в базе данных инструментов и полностью соответствуют ГОСТ. Для того чтобы можно было автоматически начертить инструмент любого типоразмера в графической системе, для каждого его вида разработана своя математическая модель. С помощью математических моделей и происходит расчет всех недостающих размеров инструмента, необходимых для его вычерчивания. [3]

На рисунке 1.1 показаны различные инструменты, построенные с помощью бибилиотеки.



Рисунок 1.1 – Интерфейс библиотеки «Инструмент»

**MechaniCS**

MechaniCS, созданный компанией CSoft, – это приложение к САПР AutoCAD и Inventor, предназначенное для разработки и оформления чертежей и спецификаций в соответствии с ЕСКД, проектирования изделий общего машиностроения.

Приложение MechaniCS располагает стандартным набором инструментов для оформления чертежей по ЕСКД. В нем удобно создавать форматы и штампы чертежей, проставлять размеры и шероховатость поверхностей, допуски размеров, формы и расположения, делать выноски и различные специальные обозначения. При помощи маркеров пользователи имеют возможность создавать собственные элементы оформления чертежей. Есть инструменты создания спецификаций. Специальные инструменты разработаны для проектирования типовых механических соединений. Кроме того, в приложение включены методики расчета различных механических характеристик деталей машин. База элементов содержит стандартные и унифицированные врезаемые элементы, детали и сборочные единицы, которые можно использовать при проектировании. [4]

На рисунке 1.2 приведён пример редактора технических требований MechaniCS.



Рисунок 1.2 – Редактор технических требований приложения MechaniCS

# 2. Описание предмета проектирования

Предметом проектирования является модель киянки, модель имеет 5 основных параметров:

1. Ширина бойка киянки A (от 40 мм до 80 мм);
2. Длина бойка киянки B (от 100 мм до 200 мм);
3. Высота бойка киянки C (от 40 мм до B/2 мм);
4. Диаметр ручки киянки D (от 20 мм до A-10 мм);
5. Высота ручки киянки H (от 150 мм до 400 мм).

На рисунке 2.1 представлен чертёж модели киянки.



Рисунок 2.1 – Чертёж модели киянки.

# 3. Проект программы

## 3.1 Диаграмма классов

Диаграмма классов определяет типы классов системы и различного рода статические связи, которые существуют между ними. На диаграммах классов изображаются также атрибуты классов, операции классов и ограничения, которые накладываются на связи между классами. Целью создания диаграммы классов является графическое представление статической структуры элементов системы.[5] Диаграмма классов плагина представлена на рисунке 3.1.

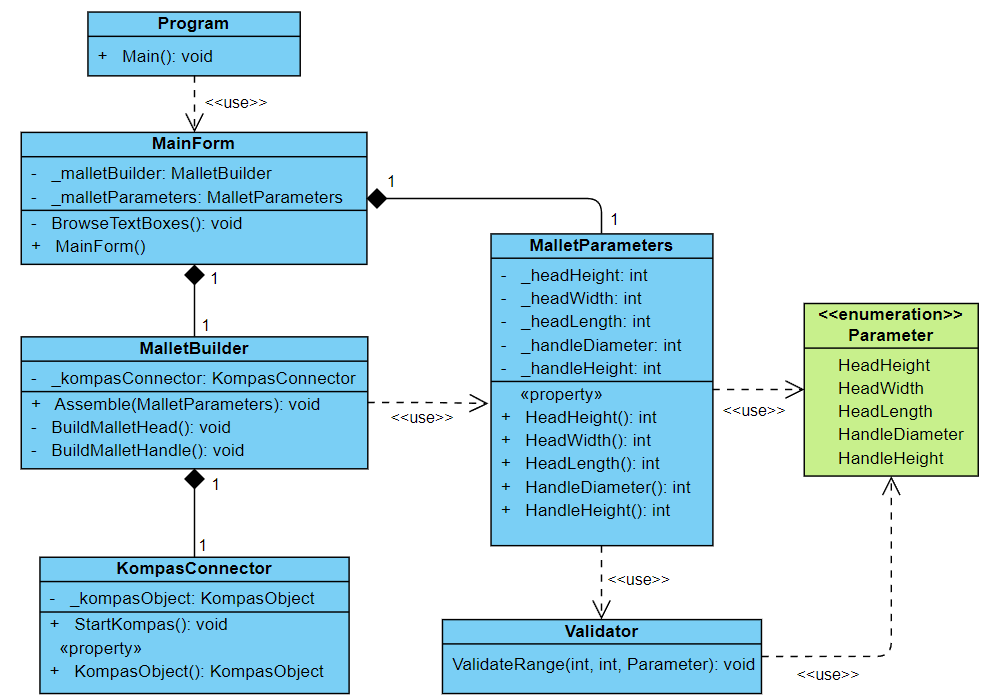


Рисунок 3.1 – Диаграмма классов плагина «Киянка»

* KompasConnector – обеспечивает связь с API «КОМПАС-3D»;
* MainForm – содержит пользовательский интерфейс плагина;
* MalletParameters – содержит параметры 3D-модели киянки;
* MalletBuilder – осуществляет вызов методов API, необходимых для постройки 3D-модели;
* Validator – содержит методы проверки диапазона вводимых значений;
* Parameter – в данном перечислении хранятся все виды параметров.

## 3.1 Макет пользовательского интерфейса

Плагин представляет собой пользовательскую форму с полями для ввода 5 геометрических параметров. Левее полей располагается описание вводимых в них параметров и их допустимые размеры. Ниже располагается кнопка для запуска построения. Справа находится изображение, демонстрирующее размеры модели для лучшего ориентирования пользователя.

Макет пользовательского интерфейса представлен на рисунке 3.1.

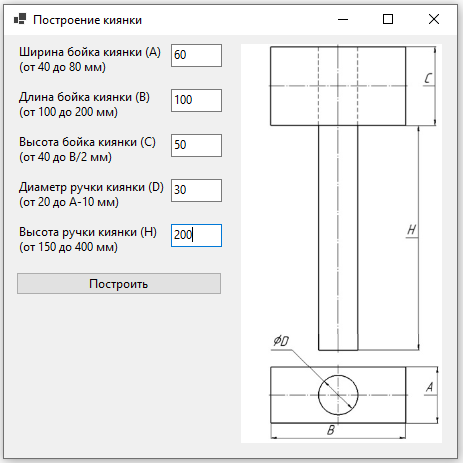


Рисунок 3.1 – Макет пользовательского интерфейса.

В случае ввода некорректных данных поля для ввода будут отмечены красным цветом, кнопка «Построить» не будет доступна, также при наведении на поле с неправильно введённым параметром будет показываться подсказка. пример этого представлен на рисунке 3.2.

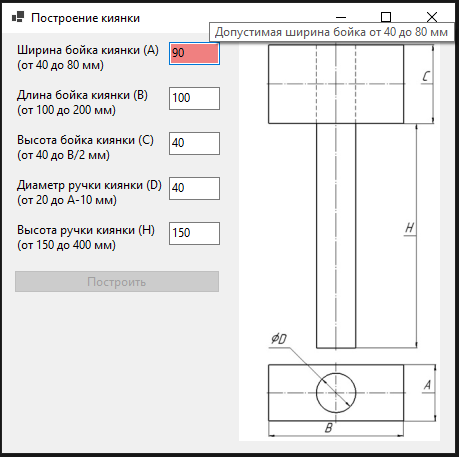


Рисунок 3.2 – Вид интерфейса при некорректном вводе данных

# Список литературы

1. Система трёхмерного моделирования КОМПАС-3D [электронный ресурс]. – URL: <https://ascon.ru/products/7/review/> (дата обращения 12.11.2021)

2. Базовые интерфейсы API системы КОМПАС [электронный ресурс]. – URL: <https://it.wikireading.ru/23741> (дата обращения 12.11.2021).

3. Библиотека «Инструмент» для «Компас-3D» [электронный ресурс]. – URL: <http://www.insoftmach.ru/Instrument.html> (дата обращения 12.11.2021).

4. CSoft MechaniCS [электронный ресурс]. – URL: [MechaniCS 2021 (csoft.ru)](https://www.csoft.ru/catalog/soft/mechanics/mechanics-2021.html) (дата обращения 12.11.2021).

5. Фаулер М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования. Изд: Символ–Плюс, 2011, с.192 (3-е издание).