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 «ЖУРНАЛЬНЫЙ СТОЛ» ДЛЯ «КОМПАС-3D V19»**

Проект системы по лабораторному проекту

по дисциплине «ОСНОВЫ РАЗРАБОТКИ САПР»

«Построение журнального стола

в системе КОМПАС-3D V19»

Выполнил:

студент гр.587-1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Монгуш С.В.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Калентьев А. А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Томск 2021

Оглавление

[1 Описание САПР 3](#_Toc36331825)

[1.1 Описание программы 3](#_Toc36331826)

[1.2 Описание API 4](#_Toc36331827)

[1.3 Обзор аналогов 8](#_Toc36331828)

[1.3.1 Плагин PDF для САПР КОМПАС-3D 8](#_Toc36331829)

[2 Описание предмета проектирования 9](#_Toc36331830)

[3 Проект программы 11](#_Toc36331831)

[3.1 Описание технических и функциональных аспектов проекта 11](#_Toc36331832)

[3.2 Диаграмма вариантов использования (Use Cases) 11](#_Toc36331833)

[3.3 Диаграмма классов 13](#_Toc36331834)

[3.3 Макет пользовательского интерфейса 14](#_Toc36331835)

[Список литературы 15](#_Toc36331836)

# 1 Описание САПР

# Описание программы

В настоящее время проектирование в своем понимании представляет собой автоматизированный процесс и в некотором роде программно-аппаратный. Проектировщику, который занимается разработкой сложного механизма, или устройства, требующего больших расчетов, математических вычислений при построении модели и высокой точности, подходят системы автоматизации проектных решений — САПР [1].

САПР позволяют уменьшить финансовые затраты на разработку макета (модели) проекта (объекта), а также сократить время, которое тратит проектировщик на создание модели объекта и составление проектной документации.

В каждой крупной САПР есть свой средства для разработки, которые предоставляются с целью дать возможность разработчикам расширить функционал данной системы под свои конкретные нужды. Данным средством является API — программный интерфейс приложения [2]. Это набор готовых средств: классов, процедур, функций, структур, констант и т.д. API позволяет определить функциональность, которую предоставляет приложение, при этом абстрагируясь от того, как она реализована.

Расширение функционала в основном подразумевает разработку плагина или библиотеки на основе предоставленного API. Плагин — независимо компилируемый программный модуль, динамически подключаемый к основной программе, предназначенный для расширения или использования ее возможностей [3].

В качестве системы, которая предоставляет API и для которой стоит задача разработать плагин, была выбрана САПР «КОМПАС-3D» V 19.

# Описание API

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

В КОМПАС-3D существуют API двух версий: API 5 и API 7. Обе версии реализуют различные функции системы и дополняют друг друга. Обе версии программных интерфейсов в равной мере поддерживаются и развиваются с учетом самих изменений в системе. В основном, для создания полноценных подключаемых модулей достаточно методов и свойств интерфейсов API 5.

Главным интерфейсом API системы КОМПАС-3D является KompasObject. Получить указатель на этот интерфейс можно при работе под управлением внешнего приложения (контроллера) – после вызова стандартной системной функции. Методы этого интерфейса реализуют наиболее общие функции работы с документами системы, системными настройками, файлами, а также дают возможность получить указатели на другие интерфейсы [4].

Ниже в таблице 1.1 представлены основные свойства и методы интерфейса KompasObject.

Таблица 1.1. – Методы интерфейса KompasObject.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| Document3D() |  | Указатель на интерфейс документа трёхмерной модели ksDocument3D | Даёт возможность получить указатель на интерфейс трёхмерного документа (детали или сборки) |
| Visible |  |  | Свойство видимости приложения |
| GetMathematic2D() |  | Указатель на интерфейс ksMathematic2D | Метод для получения указателя на интерфейс для работы с математическими функциями |
| GetParamStruct(short structType) | structType – тип интерфейса параметров | Указатель на интерфейс указаного типа из StructType2D | Метод для получения указателя на интерфейс графического документа (чертежа или фрагмента) |

Продолжение таблицы 1.1

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| GetDynamicArray(long type) | ext – расширение имени файла,  filter – фильтр поиска (0 – фильтр формируется автоматически),  preview – признак подключения окна предварительного просмотра:  с подключением окна, без подключения окна  typeDir – стартовая папка | Строка с именем файла | Возвращает указатель на интерфейс динамического массива |

Таблица 1.2 – Методы интерфейса IPart.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| EntityCollection  (short objType) | |  |  | | --- | --- | | objType | - тип объектов, содержащихся в массиве. | | Указатель на интерфейс [ksEntityCollection](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntityCollection.htm) или [IEntityCollection](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntityCollection.htm) | Формирует массив объектов и возвращает указатель на его интерфейс |
| GetDefaultEntity  (short objType) | |  |  | | --- | --- | | objType | - тип объекта. | | |  | | --- | | Указатель на интерфейс [ksEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm) или [IEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm). |   Типы объектов (objType):   |  |  |  | | --- | --- | --- | | o3d\_planeXOY | 1 | - плоскость XOY | | o3d\_planeXOZ | 2 | - плоскость XOZ | | o3d\_planeYOZ | 3 | - плоскость YOZ | | o3d\_pointCS | 4 | - точка начала системы координат | | o3d\_axisOX | 71 | - ось OX | | o3d\_axisOY | 72 | - ось OY | | o3d\_axisOZ | 73 | - ось OZ | | Получить указатель на интерфейс объекта, создаваемого системой по умолчанию |

Продолжение таблицы 1.2

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| GetPart(int type) | |  |  | | --- | --- | | type | - тип компонента. | | указатель на интерфейс компонента [ksPart](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksPart.htm) или [IPart](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksPart.htm). | Получить указатель на интерфейс компонента в соответствии с заданным типом |
| NewEntity(short objType) | |  |  | | --- | --- | | objType | - [тип объекта](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/Obj3dType_NewEntil_Part.htm). | | указатель на интерфейс [ksEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm) или [IEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm). | 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 – признак режима редактирования документа  (TRUE – невидимый режим,  FALSE – видимый режим),  typeDoc – тип документа  (TRUE – деталь,  FALSE – сборка). | TRUE – в случае успешного завершения. | Дает возможность создать пустой документ (деталь или сборку) |
| UpdateDocumentParam() |  | TRUE – в случае успешного завершения. | Активизировать измененные параметры документа |
| GetPart(int type) | type – тип компонента из перечисления Типы компонентов. |  | Получить указатель на интерфейс компонента в соответствии с заданным типом |

# Обзор аналогов

# 1.3.1 Плагин PDF для САПР КОМПАС-3D

Данный плагин позволяет производить экспорт моделей и сборок из КОМПАС-3D в формат PDF формат [5]. Основной особенностью является возможность интерактивного взаимодействия пользователя с сохраненной 3D сценой внутри PDF файла. Например, пользователь может вращать, масштабировать, передвигать детали и сборки внутри 3D PDF файла. Также доступно создание анимации сборки и разборки изделий. Это полезно для подготовки интерактивных сборочных инструкций, создания маркетинговых материалов, презентаций, а также для налаживания взаимодействия между проектировщиками и заказчиками. В подобных ситуациях традиционным подходом являлся экспорт сборки или детали КОМПАС-3D в промежуточный формат и дальнейшее сохранение в формат 3D PDF. Используемый подход в плагине исключает использование промежуточных файлов для осуществления 3D преобразования, что существенно повышает качество выходной 3D модели в формате PDF. Ключевые возможности:

* сохранение деталей и сборок в формате 3D PDF для интерактивного просмотра при помощи бесплатной программы Adobe Reader;
* создание анимаций, имитирующих естественный порядок сборки и разборки создание имитации анимации гибки листовых тел;
* вставка в существующие PDF документы, содержащие основной текст, фоновые картинки, таблицы спецификаций, эмблемы, логотипы;
* пакетный режим для поочередной конвертации всех файлов.

# 2 Описание предмета проектирования

Предметом проектирования является журнальный стол.

Журнальный столик часть интерьера дома. За ним удобно играть в настольные игры с детьми, на столешницу можно положить незаконченное рукоделие или недочитанные журналы, планшет или ноутбук, чтобы поработать в спокойной домашней обстановке. Так же удобен для чаепития.

Конструкция журнального столика представляет столешницу, к которой прикреплены из 4 ножки. У каждого столика свое индивидуальное соотношение размеров ножек и столешницы, поэтому разрабатываемый плагин имеет для каждого параметра столика определённый промежуток размера для каждого значения.

Параметры ключа:

* Длина столешницы L (от 550 до 900 мм);
* Ширина столешницы W (от 550 до 900 мм);
* Высота столешницы H (от 50 до 80 мм);
* Длина ножки L1 (от 50 до L/2 мм);
* Ширина ножки W1 (от 50 до W/2 мм);
* Высота ножки H1 (от 400 до 600 мм);

Плагин имеет зависимые параметры:

* Длина ножки L1 > L/2;
* Ширина ножки W1 > W/2

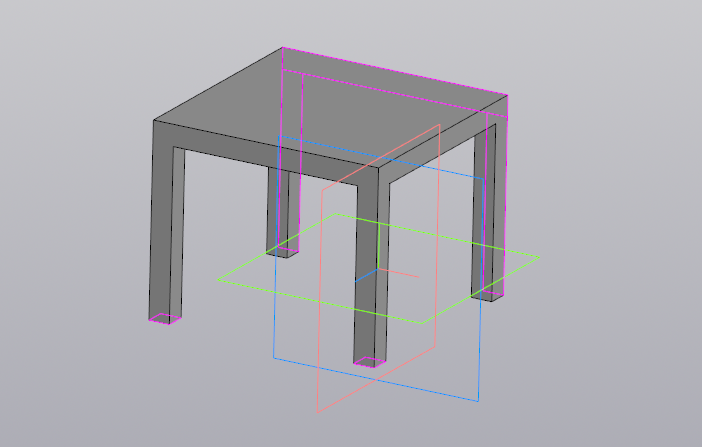


Рисунок 2.1 – Модель журнального стола в КОМПАС-3D

# 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 (Unified Modelling Language).

Для представления архитектуры программы выбирают тот или иной вид диаграмм. Язык графического описания UML позволяет представить различные аспекты функционирования программной системы с помощью определённых стандартом UML условно-графических обозначений.[7] UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем.

При использовании UML были простроены: диаграмма использования и диаграмма классов.

# 3.2 Диаграмма вариантов использования (Use Cases)

Вариант использования (use case) — это описание множества последовательных действий (включая вариации), которые выполняются некоторым субъектом с целью получения результата, значимого для некоторого действующего лица.

ВИ предполагает взаимодействие действующих лиц и системы или другого объекта. Действующее лицо представляет собой логически связанное множество ролей, которые играют пользователи системы во время взаимодействия с ней [7].

Назначение диаграммы – описание функциональности и поведения, позволяющее заказчику, пользователю, а также разработчику обсуждать проектируемую или существующую систему.

На рисунке 3.1 представлена диаграмма вариантов использования.

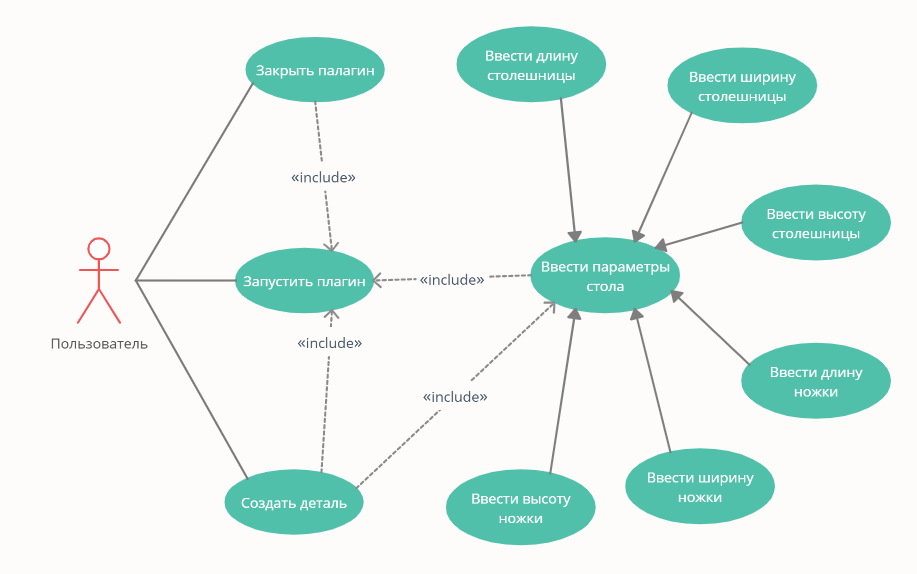


Рисунок 3.1– Диаграмма вариантов использования.

# 3.3 Диаграмма классов

Диаграмма классов описывает типы объектов системы и различного рода статические отношения, которые существуют между ними. На диаграммах классов отображаются также свойства классов, операции классов и ограничения, которые накладываются на связи между объектами [8].

На рисунке 3.2 представлена диаграмма классов.

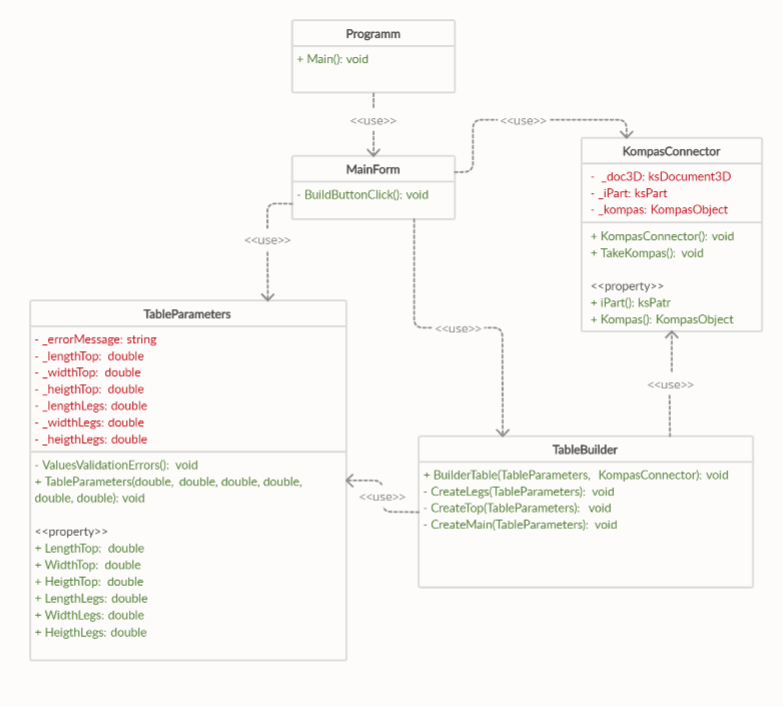


Рисунок 3.2 – Диаграмма классов

Для реализации был выбран следующий набор классов:

* MainForm – класс диалогового окна, который обеспечивает взаимодействие между пользователем и программой;
* TableParameters − класс, хранящий в себе все параметры проектируемой 3D-модели, осуществляет проверку зависимых параметров;
* KompasConnector – класс для работы с API КОМПАС 3D.
* Builder – класс, осуществляющий вызов методов API, необходимых для постройки 3D-модели.

# 3.3 Макет пользовательского интерфейса

# Список литературы