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 «ПОДВЕСНОГО ШКАФА» ДЛЯ САПР «КОМПАС** - **3D»**

Проект системы по лабораторному проекту

«Разработка плагина моделирования подвесного шкафа для системы   
Компас - 3D»  
по дисциплине «Основы разработки САПР»

Студент гр. 588-2

\_\_\_\_\_\_\_\_\_ А.А.Насонов

« » 2021

Руководитель

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_ А.А.Калентьев  
« » 2021

Томск 2021

Содержание

[1 Описание САПР 3](#_Toc38893060)

[1.1 Описание программы «Компас - 3D» 3](#_Toc38893061)

[1.2 Базовые интерфейсы API системы КОМПАС 4](#_Toc38893062)

[1.3 Обзор аналогов 13](#_Toc38893063)

[1.3.1 MechaniCS 1](#_Toc38893065)3

[2 Описание предмета проектирования 15](#_Toc38893066)

[3 Проект программы 1](#_Toc38893067)6

[3.1 UML диаграмма классов 1](#_Toc38893068)6

[3.2 Макет пользовательского интерфейса 1](#_Toc38893069)7

[Список использованных источников](#_Toc38893070) 19

1. **Описание САПР**

**1.1 Описание программы «Компас - 3D»**

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].

КОМПАС-3D учебная версия – версия, предназначенная для учебы, полностью бесплатна для образовательных учреждений. Студенческая версия не отличается по функциональности, она направленна не на коммерческую цель, а на образовательную.

КОМПАС-3D имеет несколько прикладных приложений:

* КОМПАС-3D LT – версия для ознакомления в ПО, моделированием, черчением. Можно создавать несложные модели деталей, с помощью чертежного редактора можно делать выкройки для одежды и т.п.
* КОМПАС-3D HOME – версия для моделирования моделей техники и т.п. Система поддерживает 3D-принтеры, тем самым можно легко получить реальный объект из трехмерной модели в реальной жизни [1].

**1.2 Описание API САПР КОМПАС-3D**

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. Обе версии программных интерфейсов в равной мере поддерживаются и развиваются с учетом самих изменений в системе [2].

Главным интерфейсом API системы КОМПАС является KompasObject, получить указатель на этот интерфейс (на интерфейс приложения API 5) можно с помощью экспортной функции CreateKompasObject() [2].

Методы этого интерфейса (Таблица 1.1) 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].

Таблица 1.1 – Методы интерфейса KompasObject().

|  |  |  |
| --- | --- | --- |
| **Название метода** | **Тип** | **Описание** |
| Document3D() | ksDocument | Метод для получения указателя на интерфейс трехмерного документа (детали или сборки) |
| ActivateControllerAPI() | Bool | Метод для активации API Компас 3D |
| Visible | Bool | Свойство видимости приложения |

В таблицах 1.2 – 1.3 представлены методы и параметры интерфейса ksDocument3D для сборки модели в программе КОМПАС–3D.

Таблица 1.2 – Методы интерфейса ksDocument3D.

|  |  |  |
| --- | --- | --- |
| **Название метода** | **Тип** | **Описание** |
| Create(bool invisible, bool typeDoc) | Bool | Метод для создания пустого документа (деталь или сборку) |
| GetPart(int type) | ksPart | Метод, возвращающий указатель на интерфейс детали или компонента сборки |
| Filename | String | Свойство, определяющее имя файла, из которого вставлен компонент |
| Visible | Bool | Свойство видимости приложения |

Таблица 1.3 – Входные параметры, используемых методов интерфейса ksDocument3D.

|  |  |  |
| --- | --- | --- |
| **Метод** | **Входной параметр** | **Описание** |
| Create(bool invisible, bool typeDoc) | Invisible | Признак ре­жима редакти­рования доку­мента (TRUE - неви­димый режим, FALSE - види­мый режим) |

Продолжение таблицы 1.3

|  |  |  |
| --- | --- | --- |
| **Метод** | **Входной параметр** | **Описание** |
| Create(bool invisible, bool typeDoc) | typeDoc | Тип докумен­та (TRUE - де­таль, FALSE - сбор­ка) |
| GetPart(int type) | Type | Тип компонента из перечисления.  Типы компонентов:  pInPlace\_Par – 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 |

В таблице 1.4 представлены методы, которые соответствуют свойствам трехмерных элементов модели.

Таблица 1.4 – Методы интерфейса ksEntity.

|  |  |  |
| --- | --- | --- |
| **Название метода** | **Тип** | **Описание** |
| Create() | Bool | Метод для создания объекта в модели |
| GetDefinition() | IUnkown | Метод для получения указателя на интерфейс параметров объектов и элементов |
| Create() | Bool | Метод для создания объекта в модели |

В таблицах 1.5 – 1.6 представлены методы и параметры интерфейса ksPart для создания объекта и получение на него указателя.

Таблица 1.5 – Методы интерфейса ksPart.

|  |  |  |
| --- | --- | --- |
| **Название метода или свойства** | **Тип** | **Описание** |
| GetDefaultEntity (short objType) | ksEntity | Метод для получения указателя на интерфейс объекта, создаваемого системой в трехмерном документе по умолчанию |
| NewEntity (short objType) | ksEntity | Метод, создающий интерфейс нового трехмерного объекта и возвращающий указатель на него |

Таблица 1.6 – Входные параметры, используемых методов интерфейса интерфейса ksPart.

|  |  |  |
| --- | --- | --- |
| **Метод** | **Входной параметр** | **Описание** |
| GetDefaultEntity (short objType) | objType | Тип объектов, содержащихся в массиве |
| NewEntity (short objType) | objType | Тип объектов, содержащихся в массиве |

В таблицах 1.7 – 1.8 представлены методы и параметры интерфейса ksDocument2D для создания изображения эскизов для трехмерной модели.

Таблица 1.7 – 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 |
| ksEllipse (LPDISPATCH param) | Указатель на эллипс – в случае удачного завершения,  0 – в случае неудачи | Создать эллипс |

Таблица 1.8 – 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 |
| ksEllipse (LPDISPATCH param) | param | Указатель на интерфейс [ksEllipseParam](mk:@MSITStore:D:\Program%20Files\Компас\SDK\SDK.chm::/ksEllipseParam.htm) |

В таблице 1.9 представлены методы интерфейса ksSketchDefinition для работы с эскизами.

Таблица 1.9 – Используемые методы интерфейса ksSketchDefinition.

|  |  |  |
| --- | --- | --- |
| **Название** | **Возвращаемый тип** | **Описание** |
| BeginEdit() | Bool | Войти в режим редактирования эскиза (ksDocument2D) |

Продолжение таблицы 1.9.

|  |  |  |
| --- | --- | --- |
| EndEdit() | Bool | Выйти из режима редактирования эскиза |

В таблице 1.10 представлены типы объектов документа-модели, используемые при разработке плагина.

Таблица 1.10 – Типы объектов документа-модели.

|  |  |  |  |
| --- | --- | --- | --- |
| **Метод** | **Тип объекта** | **Название объекта** | **Интерфейс параметров** |
| GetDefault Entity (short objType) | o3d\_planeXOZ | Плоскость XOZ | ksPlaneParam |
| o3d\_axisOY | Ось OY |  |
| NewEntity (short objType) | o3d\_sketch | Эскиз трехмерной операции | ksSketchDefinition |
| o3d\_cutExtrusion | Операция вырезания выдавливанием | ksCutExtrusionDefinition |
| o3d\_circularCopy | Копирование по концентрической сетке | ksCircularCopyDefinition |
| o3d\_bossExtrusion | Операция выдавливания | ksBossExtrusionDefinition |

В таблицах 1.11 – 1.13 представлены используемые методы интерфейсов для типов объектов.

Таблица 1.11 – Методы интерфейса ksCutExtrusionDefinition

|  |  |  |
| --- | --- | --- |
| **Название** | **Тип** | **Описание** |
| SetSideParam (bool forward, short type, double depth, double draftValue, bool draftOutward) | Bool | Установить параметры выдавливания в одном направлении |

Таблица 1.12 – Методы интерфейса ksCircularCopyDefinition

|  |  |  |
| --- | --- | --- |
| **Название** | **Тип** | **Описание** |
| SetAxis(LPDISPATCH axis); | Bool | Установить ось копирования |
| SetCopyParamAlongDir (long count, double step, bool factor, bool dir) | Bool | Установить параметры копирования |

Таблица 1.13 – Описание входных параметров интерфейсов.

|  |  |  |
| --- | --- | --- |
| **Метод** | **Входной параметр** | **Описание** |
| SetSideParam (bool forward, short type, double depth, double draftValue, bool draftOutward) | Forward | Направление выдавливания: TRUE – прямое направление, FALSE – обратное направление |

Продолжение таблицы 1.13

|  |  |  |
| --- | --- | --- |
| **Метод** | **Входной параметр** | **Описание** |
|  | Type | Тип выдавливания.  Виды:  etBlind – строго на глубину;  etThroughAll – через всю деталь;  etUpToVertexTo – на расстояние до вершины;  etUpToVertexFrom – на расстояние за вершину;  etUpToSurfaceTo – на расстояние до поверхности;  etUpToSurfaceFrom – на расстояние за поверхность;  etUpToNearSurface – до ближайшей поверхности. |
| Depth | Глубина выдавливания |
| draftValue | Угол уклона |
| draftOutward | Направление уклона: FALSE - уклон наружу, TRUE - уклон внутрь |

Продолжение таблицы 1.13

|  |  |  |
| --- | --- | --- |
| **Метод** | **Входной параметр** | **Описание** |
| SetAxis(LPDISPATCH axis); | Axis | Указатель на интерфейс оси [ksEntity](mk:@MSITStore:C:\Program%20Files\ASCON\KOMPAS-3D%20V16\SDK\SDK.chm::/ksEntity.htm) |
| SetCopyParamAlongDir (long count, double step, bool factor, bool dir) | Count | Количество копий |
| Step | Шаг |
| Factor | Признак полного шага |
| Dir | Направление копирования |
| SetSketch (LPDISPATCH sketch) | Sketch | Указатель на интерфейс эскиза [ksEntity](mk:@MSITStore:C:\Program%20Files\ASCON\KOMPAS-3D%20V16\SDK\SDK.chm::/ksEntity.htm) |

В таблице 1.14 представлено описание входных параметров этих методов.

Таблица 1.14 – Методы интерфейса ksBossExtrusionDefinition.

|  |  |  |
| --- | --- | --- |
| Название | 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 |

**1.3 Обзор аналогов**

**1.3.1 MechaniCS**

**MechaniCS** — 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 средства 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 [3].

Все детали общей конструкторско-технологической базы обладают интеллектом и являются объектно-зависимыми. При изменении параметров одной детали все связанные с ней объектно-зависимые детали изменятся автоматически, причем в соответствии с их параметрами в базе. Такая технология — мощный инструмент многовариантного проектирования, залог повышения качества выпускаемых проектов. Важно, что этот подход одинаково доступен пользователям AutoCAD и Autodesk Inventor [3].

На Рисунке 1.1 представлено приложение MechaniCS.



Рисунок 1.1 – Скриншот приложения MechaniCS

1. **Описание предмета проектирования**

Подвесной шкаф — это мебельная конструкция в виде негабаритного шкафа, монтируемого на стене. Для производства навесных конструкций используют пластик, натуральное дерево, фанера, древесно стружечная плита, древесноволокнистая плита средней плотноси или ламинированную древесно-стружечную плиту.

Значение геометрических параметров может быть зависимо от других

1. Высота шкафа = 540 мм ≤Z≤1000 мм;
2. Глубина шкафа = 280 мм ≤Y≤320 мм;
3. Ширина шкафа = 300 мм ≤X≤450 мм;
4. Высота первой полки шкафа = 100 мм ≤Z1=Z-Z2-Z3 мм;
5. Высота второй полки шкафа = 100 мм ≤Z2=Z-Z1-Z3 мм;
6. Высота третей полки шкафа = 100 мм ≤Z3=Z-Z1-Z2 мм;

Изображение шкафа: Z(высота шкафа), Y(глубина шкафа), X(ширина шкафа), Z1(высота первой полки шкафа), Z2(высота второй полки шкафа), Z3(высота третей полки шкафа) приведено на рисунке 2.1.

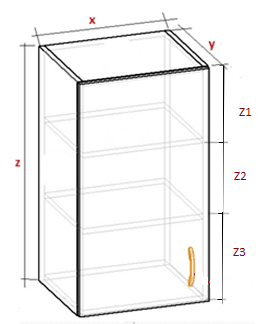


Рисунок 2.1 – Изображение шкафа

1. **Проект программы**
   * 1. **UML диаграмма классов**

Унифицированный язык моделирования (Unified Modeling Language) – это система обозначений, которую можно применять для объектно-ориентированного анализа и проектирования. Его можно использовать для визуализации, спецификации, конструирования и документирования программных систем [4].

Диаграмма классов предназначена для представления внутренней структуры программы в виде классов и связей между ними. Все сущности должны быть представлены объектами классов в программе. При этом у каждого класса должно быть только одно назначение и уникально осмысленное имя, которое будет связано с этой целью [4].

На диаграммах классов отображаются также свойства классов, операции классов и ограничения, которые накладываются на связи между объектами. В UML термин функциональность применяется в качестве основного термина, описывающего и свойства, и операции класса. Целью создания диаграммы классов является графическое представление статической структуры декларированных элементов системы (Рисунок 3.2)

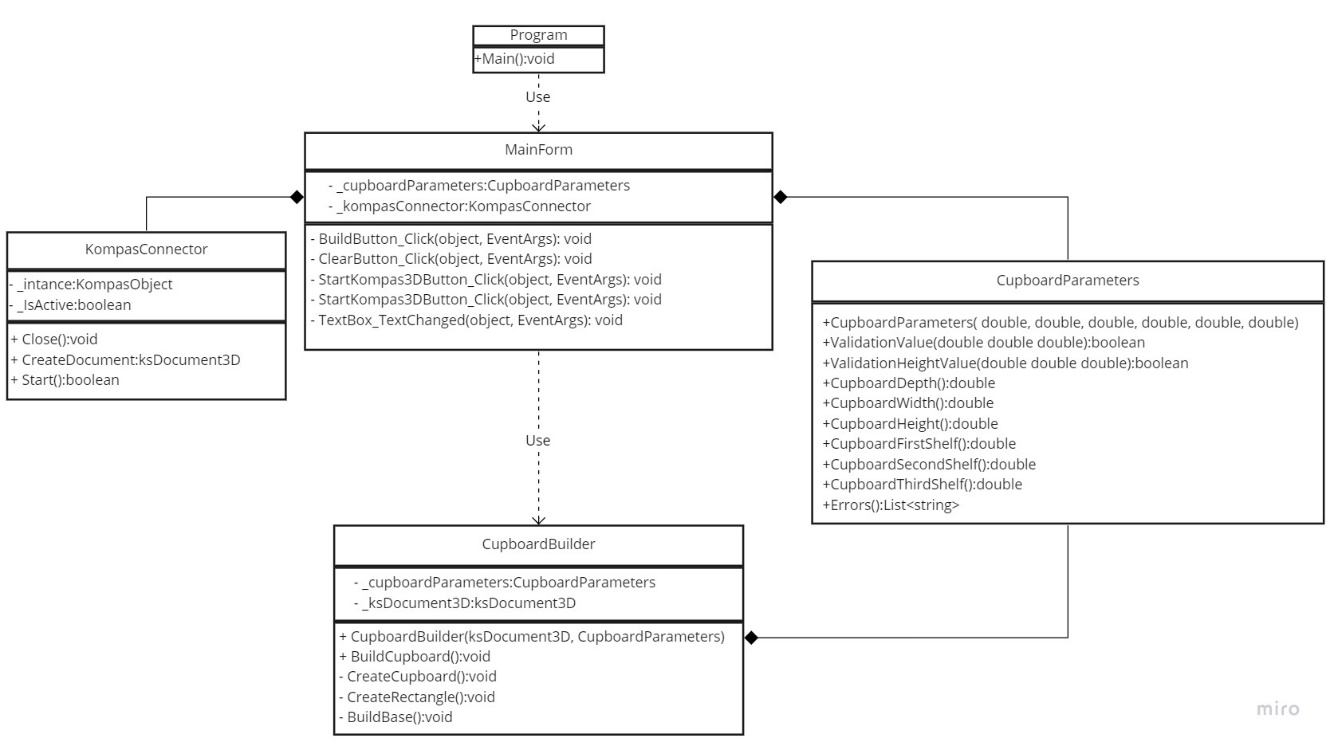


Рисунок 3.2 – Диаграмма классов

Класс «Program», использует «MainForm» для обработки действий в графическом интерфейсе.

Класс «CupboardBuilder» содержит в себе методы создания 3D модели в «Компас 3D»,

Класс «CupboardParameters» включает в себя введенные значения в графическом интерфейсе.

Класс «KompasConnector» производит запуск программы «КОМПАС-3D» и строит объект в этой программы.

* 1. **Макеты пользовательского интерфейса**

Макет пользовательского интерфейса создан с помощью Windows Form. На рисунке 3.3 представлен макет пользовательского интерфейса для ввода параметров модели. Так же на рисунке 3.3 представлено, как будет отображаться не правильный ввод параметров (поле окрашивается в красный цвет), если значения введены правильно, то окно не изменяет цвет.

Перед пользователем представлены 6 полей, предназначенный для ввода параметров (в мм) детали. Помимо этого, присутствует кнопка для построения модели, в дальнейшем при нажатии на которую будет загружаться Компас-3D и начинаться построение модели кости, и кнопка для построения модели по базовым параметрам, чтобы облегчить пользователю представление модели. Также в правой части формы пользователь видит схему будущей детали с разметкой всех параметров.

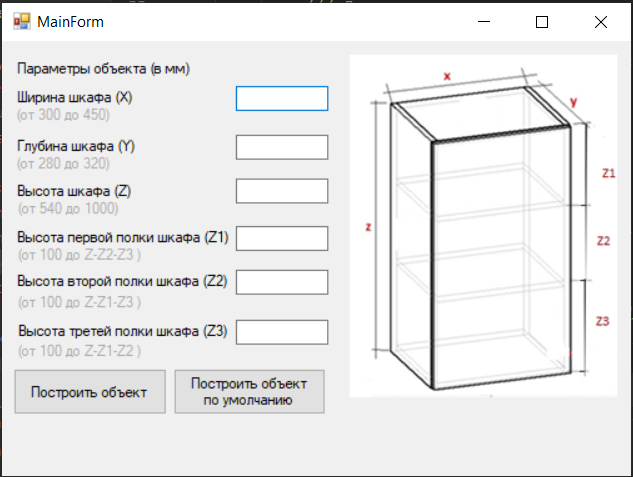


Рисунок 3.3 – Макет пользовательского интерфейса

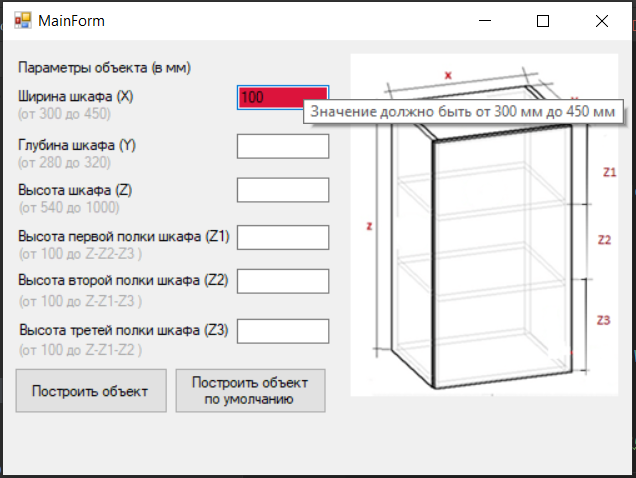


Рисунок 3.4 – Подсказка об ошибке

**Список использованных источников**

Продукты АСКОН. [Электронный ресурс]. — Режим доступа: https://incona.net/software/ascon/products (дата обращения: 19.10.2021).

Базовые интерфейсы API системы КОМПАС. [Электронный ресурс]. — Режим доступа: https://it.wikireading.ru/23741 (дата обращения: 20.10.2021)

MechaniCS. [Электронный ресурс]. — Режим доступа: https://www.csoft.ru/catalog/soft/mechanics/mechanics-2021.html (дата обращения: 20.10.2021)

Мартин Фаулер. UML. Основы. / Мартин Фаулер; пер. с англ. А. Петухова – 3-е издание. – Спб: Символ-Плюс, 2004 – 192с.