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 «КНИЖНЫЙ ШКАФ» ДЛЯ «КОМПАС-3D»**

Проект системы по лабораторному проекту

по дисциплине «ОСНОВЫ РАЗРАБОТКИ САПР»

«Построение книжного шкафа в системе КОМПАС-3D»

Выполнил:

студент гр. 588-1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Черкас О.С.

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

Руководитель:

к.т.н., доцент каф. КСУП

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

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

Томск 2021

**1 Описание САПР**

* 1. **Описание программы Компас-3D**

В настоящее время проектирование в своем понимании представляет собой автоматизированный процесс и в некотором роде программно-аппаратный. Проектировщику, который занимается разработкой сложного механизма подходят системы автоматизации проектных решений — САПР [1].

САПР позволяют уменьшить финансовые затраты на разработку макета, 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 функциональности, 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» версии 19.

КОМПАС-3D – это российская система трехмерного проектирования, ставшая стандартом для тысяч предприятий и десятков тысяч профессиональных пользователей. Система КОМПАС-3D широко используется для проектирования изделий основного и вспомогательного производств в таких отраслях промышленности, как машиностроение, приборостроение, станкостроение и т. д.

* 1. **Обзор аналогов**

K3-Мебель — это комплекс компьютерных программ для проектирования, производства и продажи корпусной мебели.

К3-Мебель позволяет автоматизировать процесс приема заказов и подготовки производственных заданий на базе компьютерных технологий. Комплекс К3-Мебель обеспечивает:

* создание дизайна помещения заказчика при продаже мебели в салоне;
* сокращение времени приема и расчета стоимости заказа;
* снижение затрат на разработку проектно-конструкторской и технологической документации;
* значительное ускорение разработки новых проектов и выпуска конструкторско-технологической документации;
* исключение ошибок при передаче информации из салона на производство;
* экономию материальных ресурсов за счет точного расчета используемых в изделии материалов;
* высокое качество оформления документации;
* взаимное увязывание приема заказов, подготовки производства, складского и бухгалтерского учета;
* повышение эффективности труда за счет коллективной работы в единой информационной среде;
* рост квалификации персонала предприятия и его готовности к дальнейшим инновациям.

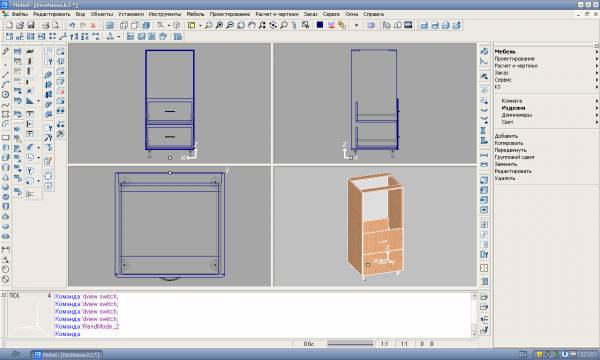


Рисунок 1.1 – Интерфейс плагина

Работа комплекса основана на принципе «От компьютерной модели к полному комплекту документов». Конструктору необходимо только создать модель будущего изделия, а все чертежи, спецификации, управляющие программы для ЧПУ программный комплекс К3-Мебель подготовит в автоматическом режиме. Автоматическое получение всего пакета документов исключает возникновение ошибок, связанных с «человеческим фактором». Программный комплекс K3-Мебель рекомендуется для использования в мебельных салонах, магазинах и предприятиях, специализирующихся на проектировании, производстве и монтаже корпусной мебели.

Комплекс К3-Мебель, обладая «открытым кодом», позволяет пользователю самостоятельно производить точную настройку программы на конкретную специфику, реализовывать свой информационный и производственный документооборот, обеспечивая тем самым информационную безопасность и независимость от разработчика.[4]

1. **Описание API Компас-3D**

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 (на интерфейс приложения API 5) 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.

Ниже в таблице 2.1 представлены основные свойства и методы интерфейса KompasObject.

Таблица 2.1 – 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) |

Продолжение таблицы 2.1

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| 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 2.2 представлены методы интерфейса ksEntity, необходимые для разработки плагина.

Таблица 2.2 – 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 2.3 представлены свойства и методы интерфейса ksDocument2D, необходимые для разработки плагина

Таблица 2.3 – 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 2.4 представлены свойства и методы интерфейса ksDocument3D, необходимые для разработки плагина.

Таблица 2.4 – Методы интерфейса ksDocument3D

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

В таблице 2.5 представлены свойства методы интерфейса ksPart, необходимые для разработки плагина.

Таблица 2.5 – Свойства и 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 |

В таблице 2.6 представлены типы объектов документа-модели, необходимые для разработки плагина.

Таблица 2.6 – Некоторые типы объектов документа-модели

|  |  |  |
| --- | --- | --- |
| Идентификатор объекта | Название объекта | Интерфейс параметров |
| o3d\_unknown | Неизвестный (включает все объекты) |  |
| o3d\_planeXOZ | Плоскость XOZ | ksPlaneParam |
| o3d\_planeYOZ | Плоскость YOZ | ksPlaneParam |
| o3d\_planeXOY | Плоскость XOY | ksPlaneParam |
| o3d\_sketch | Эскиз | ksSketchDefinition |
| o3d\_face | Грань | ksFaceDefinition |
| o3d\_baseExtrusion | Базовая операция выдавливания | ksBaseExtrusionDefinition |

1. **Описание предмета проектирования**

Книжный шкаф (шкаф для книг) — шкаф, предназначенный для хранения книг. Преимущественно книжные шкафы изготавливаются с остекленными или полностью стеклянными дверями, внутри корпуса которого расположены полки, регулируемые по высоте.

Шкафы для книг могут быть цельными или составными и состоять из одного или двух отделений. Обычно нижнее отделение имеет полки или ящики за двумя глухими дверками, а в верхнем отделении за стеклянными дверцами размещаются переставные полки на полкодержателях, позволяющих регулировать расстояние между ними в зависимости от размера книг. Нижнее отделение книжного шкафа может быть одной ширины с верхним или несколько шире него. Обычно в нижнем отделении хранят книги большего формата. Книги на полках можно размещать в один или два ряда. Дверки книжных шкафов бывают на петлях или раздвижные, застеклённые или деревянные гладкие. Книжный шкаф в составе секционной мебели представляет собой отдельные секции по вертикали, поставленные друг на друга.[6]

По условию ТЗ шкаф может менять свои размеры, не нарушая установленные рамки, приведенные ниже:

1. W – Ширина шкафа: от 50 до 300 см;
2. H – Высота шкафа: не больше 200 см;
3. S – Глубина шкафа: от 20 до 70 см;
4. H2 – Высота полок;
5. F – Толщина материала: от 2 до 10 см.

Плагин имеет следующие зависимости:

1. W не может превышать H более чем в 1.5 раза;
2. Высота шкафа не может быть меньше высоты пяти полок;
3. H2 не может превышать значение 1/5 H.

На рисунке 3.1 представлен чертеж с указанными параметрами.

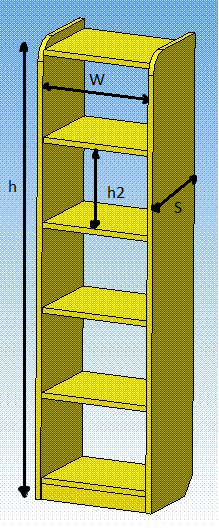


Рисунок 3.1 – Чертеж книжного шкафа

1. **Проект программы**
   1. **Диаграмма классов**

Для графического описания абстрактной модели проекта, а также пользовательского взаимодействия (сценарии действия) использован стандарт UML.

UML язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это – открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML – моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML возможна генерация кода и наоборот.

Диаграмма классов – структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей между ними.[6]

Диаграмма классов плагина представлена на рисунке 4.1.

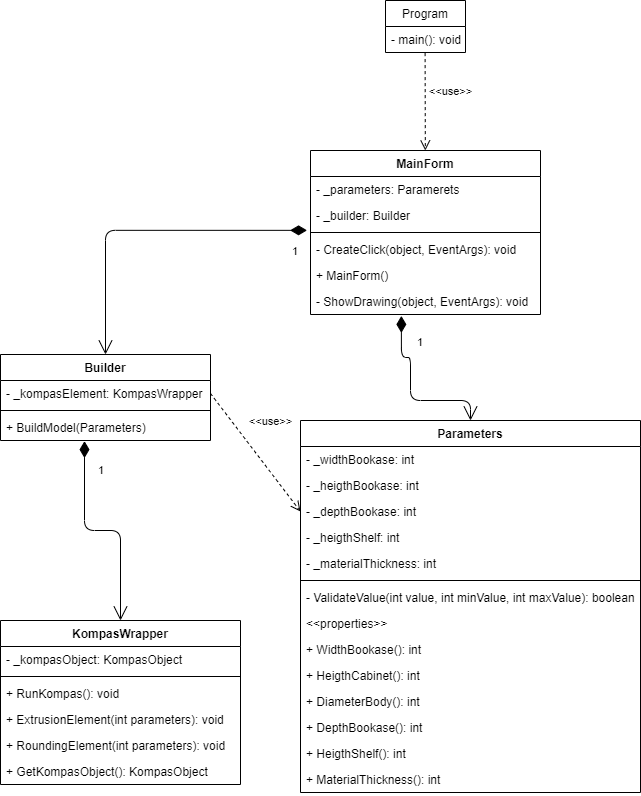


Рисунок 4.1 – Диаграмма классов плагина «Книжный шкаф»

Для реализации был выбран следующий набор классов:

Класс «Program» использует «MainForm» для обработки действий в графическом интерфейсе. «Builder» содержит в себе метод BuildModel() для создания 3D модели в «Компас 3D», которая также подключается к САПР при помощи «KompasWrapper». Класс «Parameters» содержит введенные значения в графическом интерфейсе. При передаче значений свойствам класса «Parameters» в сеттерах при помощи метода «ValidateValue» проверяется правильность диапазона значения. В случае выхода из диапазона вызывается исключение. «KompasWrapper» имеет публичные методы выдавливания (ExtrusionElement()) и скругления (RoundingElement()) для создания модели в САПР.

* 1. **Макет пользовательского интерфейса**

Макет пользовательского интерфейса представляет собой форму для ввода параметров книжного шкафа (рисунок 4.1).

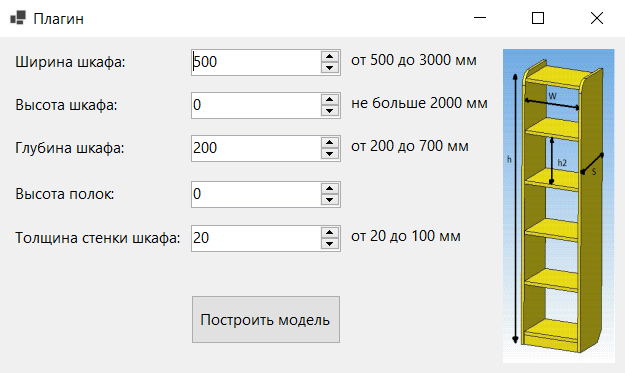


Рисунок 4.1 – Главное окно плагина

Построение модели осуществляется путем нажатия на кнопку «Построить модель». При попытке ввода недопустимых символов, они не будут вводиться в строку (если необходимо ввести цифры, то невозможно будет ввести другие символы). Также при вводе в строку цифр из недопустимого диапазона, который указан справа от каждой области ввода, при попытке построить модель значения будут сбрасываться до минимально или максимально допустимых.

**Список литературы**

Норенков И.П. «Основы автоматизированного проектирования». Издательство: МГТУ; Москва:, 2002 – 336 с.

API – Википедия. [Электронный ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/API> (дата обращения 23.10.2021)

Плагин – Википедия. [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Плагин> (дата обращения 25.10.2021)

K3-Мебель. [Электронный ресурс]. – Режим доступа: http://mebdom.my1.ru/index/k3\_mebel\_raskroj/0-47 (дата обращения 11.11.2021)

Книжный шкаф [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/%D0%9A%D0%BD%D0%B8%D0%B6%D0%BD%D1%8B%D0%B9\_%D1%88%D0%BA%D0%B0%D1%84#:~:text=%D0%9A%D0%BD%D0%B8%D0%B6%D0%BD%D1%8B%D0%B9%20%D1%88%D0%BA%D0%B0%D1%84%20(%D1%88%D0%BA%D0%B0%D1%84%20%D0%B4%D0%BB%D1%8F%20%D0%BA%D0%BD%D0%B8%D0%B3,%D1%88%D0%BA%D0%B0%D1%84%2C%20%D0%BF%D1%80%D0%B5%D0%B4%D0%BD%D0%B0%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%BD%D1%8B%D0%B9%20%D0%B4%D0%BB%D1%8F%20%D1%85%D1%80%D0%B0%D0%BD%D0%B5%D0%BD%D0%B8%D1%8F%20%D0%BA%D0%BD%D0%B8%D0%B3.&text=%D0%9E%D0%B1%D1%8B%D1%87%D0%BD%D0%BE%20%D0%BD%D0%B8%D0%B6%D0%BD%D0%B5%D0%B5%20%D0%BE%D1%82%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B8%D0%BC%D0%B5%D0%B5%D1%82%20%D0%BF%D0%BE%D0%BB%D0%BA%D0%B8,%D0%B2%20%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D0%B8%20%D0%BE%D1%82%20%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%80%D0%B0%20%D0%BA%D0%BD%D0%B8%D0%B3. (дата обращения 01.11.2021)

UML. [Электронный ресурс]. – Режим доступа: http://www.uml.org/ (дата обращения 9.11.2021)