Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждения

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»**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил:

студент гр. 588-1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Черкас О.С.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Калентьев А.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Оглавление

[1 Введение 3](#_Toc100670736)

[2 Постановка и анализ задачи 4](#_Toc100670737)

[2.1 Описание предмета проектирования 5](#_Toc100670738)

[2.2 Выбор инструментов и средств реализации 6](#_Toc100670739)

[2.3 Назначение плагина 6](#_Toc100670740)

[3 Обзор аналогов 7](#_Toc100670741)

[4 Описание реализации 9](#_Toc100670742)

[4.1 Диаграмма классов 9](#_Toc100670743)

[5 Описание программы для пользователя 16](#_Toc100670744)

[6 Тестирование программы 18](#_Toc100670745)

[6.1 Функциональное тестирование 18](#_Toc100670746)

[6.2 Модульное тестирование 20](#_Toc100670747)

[6.3 Нагрузочное тестирование 21](#_Toc100670748)

[Заключение 24](#_Toc100670749)

[Список использованных источников 25](#_Toc100670750)

# 1 Введение

Автоматизация моделирования имеет огромное значение для развития науки, техники и производства в современном обществе. В настоящее время автоматизация – основной способ повышения производительности и эффективности труда инженерно-технических работников, занимающихся моделированием сложных устройств. Использование автоматизации в проектировании позволяет создавать все более сложные технические объекты и гибко реагировать на появление новых решений и технологий в той или иной области техники. Она позволяет значительно повысить точность расчетов, выбрать наилучшие варианты для реализации на основе строгого математического анализа всех или большинства вариантов проекта с оценкой технических, технологических и экономических характеристик производства и эксплуатации проектируемого объекта, значительно повысить качество конструкторской документации, существенно сократить сроки проектирования и передачи конструкторской документации в производство, эффективнее использовать технологическое оборудование с программным управлением [1].

Таким образом, целью данной работы является разработка плагина, автоматизирующего построение модели «Книжный шкаф» для системы автоматизированного проектирования Компас 3D с помощью интегрированной среды разработки Visual Studio 2019 Сommunity. [2]

Интегрированная среда разработки Visual Studio — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения.

# 2 Постановка и анализ задачи

В рамках лабораторных работ в соответствии с технически заданием требовалось разработать плагин, который на основе входных параметров, интегрируя с системой Компас 3D, строит модель «Книжный шкаф». [3] Необходимо чтобы плагин позволял задавать параметры по умолчанию, а также изменять входные параметры книжного шкафа, такие как:

* ширина шкафа;
* высота шкафа;
* глубина шкафа;
* высота полок;
* толщина материала;
* тип ножек.

## **2.1 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.[4]

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.

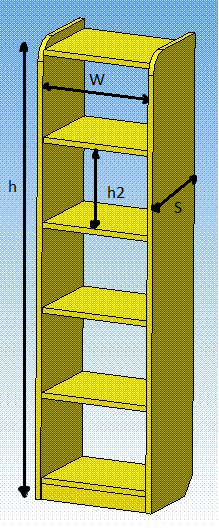


Рисунок 2.1 – Чертеж книжного шкафа

## **2.2 Выбор инструментов и средств реализации**

На основе требований к техническому заданию программа выполнена на языке программирования C# в среде Microsoft Visual Studio 2019 с использованием .NET Framework 4.7.2 [2], бибиблиотеки для Kompas 3D [5].

Инструментом тестирования и создания модульных тестов был выбран тестовый фреймворк NUnit [6] версии 3.13.2.

Для реализации пользовательского интерфейса использовалась система для построения настольные приложения WinForms [7].

## **2.3 Назначение плагина**

Назначение разрабатываемого плагина обусловлено быстрым моделированием книжных шкафов разных типов. Благодаря данному расширению, столяры могут наглядно рассмотреть спроектированную модель, при необходимости перестроить под необходимые им параметры.

# 3 Обзор аналогов

**К3-Мебель**

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.

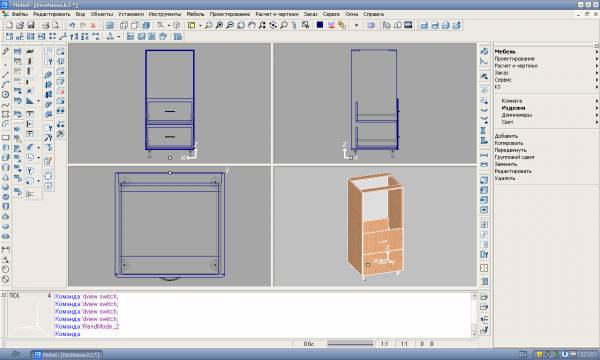


Рисунок 3.1 – Интерфейс плагина

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.[8]

# 4 Описание реализации

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.[9]

При использовании UML были простроена диаграмма классов.

## **4.1 Диаграмма классов**

Диаграмма классов – структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей между ними.[9]

На рисунке 4.1 представлена диаграмма классов.

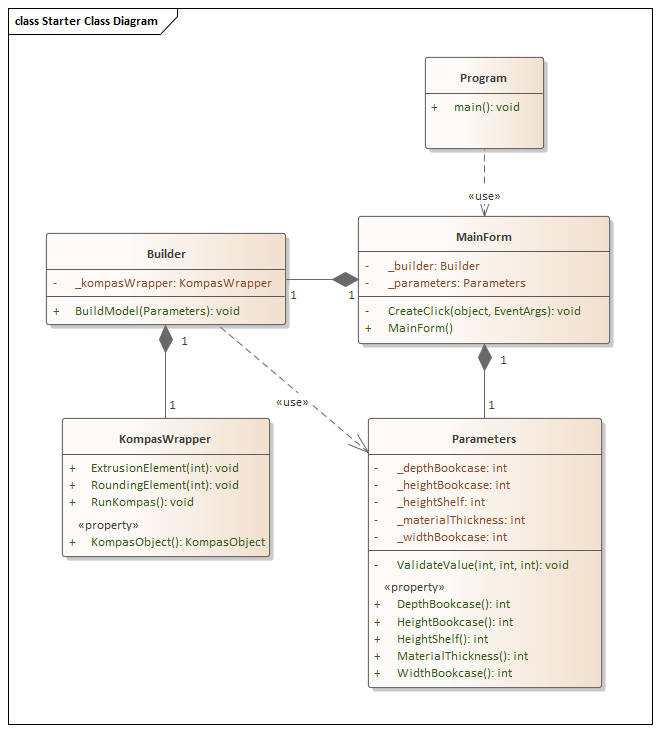


Рисунок 4.1 – Изначальная диаграмма классов

Для реализации был выбран следующий набор классов:

Класс «Program» использует «MainForm» для обработки действий в графическом интерфейсе. В таблицах 4.1 и 4.2 представлено описание членов класса «MainForm».

«Builder» содержит в себе метод BuildModel() для создания 3D модели в «Компас 3D», которая также подключается к САПР при помощи «KompasWrapper». В таблице 4.3 представлено описание членов класса «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. В таблицах 4.4 и 4.5 представлено описание членов класса «Parameters».

Таблица 4.1 – Описание полей класса «MainForm»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_builder | Builder | Экземпляр класса строителя книжного шкафа |
| \_parameters | Parameters | Экземпляр класса параметров книжного шкафа |

Таблица 4.2 – Описание методов класса «MainForm»

|  |  |  |
| --- | --- | --- |
| Название | Входные параметры | Описание |
| CreateClick | object sender,  EventArgs args | Обработчик события нажатия на кнопку «Построить модель» |
| MainForm |  | Конструктор класса «MainForm» |

Таблица 4.3 – Описание методов класса и свойств «KompasWrapper»

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Возвращаемое значение | Описание |
| ExtrusionElement | int parameterValue | void | Метод выдавливания эскиза |
| RoundingElement | int parameterValue | void | Метод скругления |
| RunKompas |  | void | Метод запуска «Компас 3D» |
| KompasObject |  | KompasObject | Свойство, возвращающее Интерфейс API КОМПАС |

Таблица 4.4 – Описание полей класса «Parameters»

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_depthBookcase | int | Глубина шкафа |
| \_heightBookcase | int | Высота шкафа |
| \_heightShelf | int | Высота полки |
| \_materialThickness | int | Толщина материала |
| \_widthBookcase | int | Ширина шкафа |

Таблица 4.4 – Описание методов класса «Parameters»

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Возвращаемое значение | Описание |
| ValidateValue | int value,  int minValue,  int maxValue | void | Проверяет, входит ли значение элемента в промежуток minValue и maxValue |
| DepthBookcase |  | int | Свойство, которое возвращает и задает значение глубины шкафа |
| HeightBookcase |  | int | Свойство, которое возвращает и задает значение высоты шкафа |
| HeightShelf |  | int | Свойство, которое возвращает и задает значение высоты полки |
| MaterialThickness |  | int | Свойство, которое возвращает и задает значение толщины материала |
| WidthBookcase |  | int | Свойство, которое возвращает и задает значение ширины шкафа |

В итоговом проекте созданы следующие классы и методы, которые отображены на итоговой диаграмме классов (рисунок 4.2).

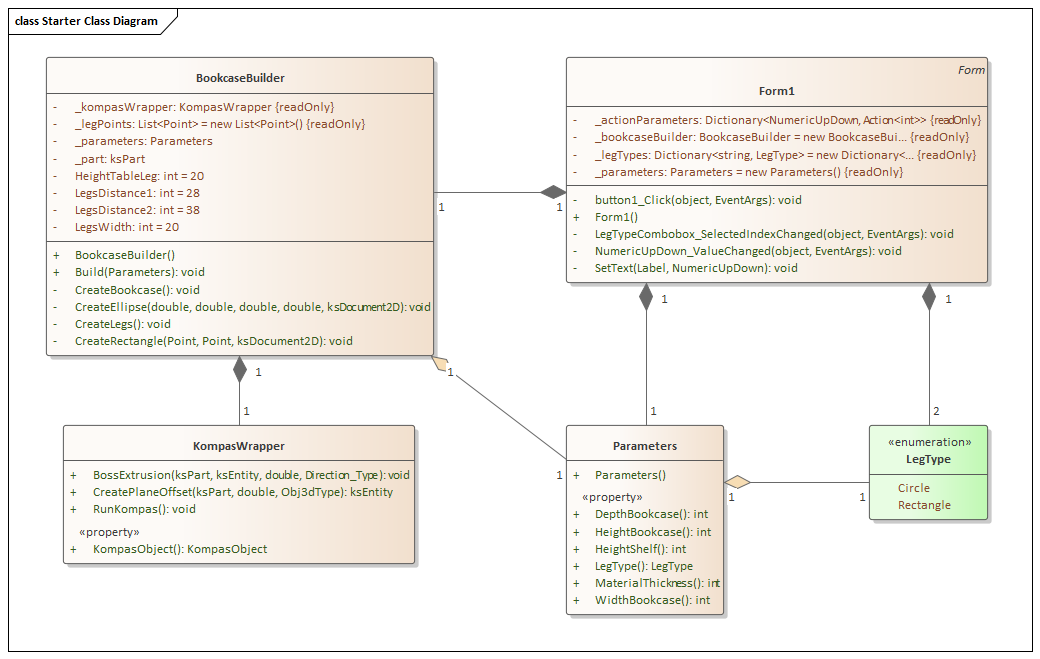


Рисунок 4.2 – Итоговая диаграмма классов

В класс Form1 были добавлены обработчик события для элементов NumericUpDown, ComboBox и Button. Метод SetText новые надписи с ограничениями для элементов Label (таблицы 4.5 и 4.6).

В класс Parameters было добавлено свойство для выбора типа ножек (LegType). Были убраны поля и метод валидации, так как валидация проводится в классе Form1 (таблицы 4.7).

В класс BookcaseBuilder был добавлены константы, а также методы для создания частей шкафа. Было добавлено поле тапа Parameters для получения значения параметров (таблицы 4.8 и 4.9).

В класс KompasWrapper были добавлены методы для выдавливания и создания эскизов (таблицы 4.10).

Таблица 4.5 – Поля класса Form1

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| \_parameters | Parameters | Параметры книжного шкафа |
| \_bookcaseBuilder | BookcaseBuilder | Строитель шкафа |
| \_actionParameters | Dictionary<NumericUpDown, Action<int>> | Словарь действий |
| \_legTypes | Dictionary<string, LegType> | Словарь типов ножек |

Таблица 4.6 – Методы класса Form1

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Параметры | Возвращаемый тип | Описание |
| Form1 | – | – | Конструктор |
| SetText | Label, NumericUpDown | void | Установить текст в Label |
| Button1\_Click | object, EventArgs | void | Обработчик события нажатия на кнопку |
| NumericUpDown\_ ValueChanged | object, EventArgs | void | Оброботчик события изменения значения в NumericUpDown |
| LegTypeCombobox\_ SelectedIndexChanged | object, EventArgs | void | Обработчик события измененрия выбора в комбобоксе |

Таблица 4.7 – Методы и свойства класса Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Параметры | Возвращаемый тип | Описание |
| DepthBookcase | – | int | Свойство возвращает и задает глубину шкафа |
| HeightBookcase | – | int | Свойство возвращает и задает высоту шкафа |
| HeightShelf | – | int | Свойство возвращает и задает высоты полки |
| MaterialThickness | – | int | Свойство возвращает и задает толщину материала |
| WidthBookcase | – | int | Свойство возвращает и задает ширину шкафа |
| LegType | – | LegType | Свойство возвращает и задает тип ножек |
| Parameters | - | - | Конструктор |

Таблица 4.8 – Поля класса BookcaseBuilder

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| HeightTableLeg | int | Высота ножек (значение 20) |
| LegsWidth | int | Ширина ножек (значение 20) |
| LegsDistance1 | int | Первое расстояние от угла (значение 28) |
| LegsDistance2 | int | Второе расстояние от угла (значение 38) |
| \_legPoints | List<Point> | Координаты ножек комода |
| \_kompasWrapper | KompasWrapper | Экземпляр класса работы с Компас 3D |
| \_part | ksPart | Часть модели |
| \_parameters | Parameters | Параметры модели |

Таблица 4.9 – Методы класса BookcaseBuilder

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Параметры | Возвращаемый тип | Описание |
| BookcaseBuilder | – | – | Конструктор |
| Build | Parameters | void | Метод строит модель |
| CreateBookcase r | – | void | Метод создать шкаф |
| CreateLegs | – | void | Метод создает ножки |
| CreateRectangle | Point, Point, ksDocument2D | void | Метод создает прямоугольник по двум точкам |
| CreateEllipse | double, double, double, double, ksDocument2D | void | Метод создает эллипс |

Таблица 4.10 – Методы и свойства класса KompasWrapper

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Параметры | Возвращаемый тип | Описание |
| KompasObject | – | KompasObject | Свойство, возвращающее экземпляр Компас 3D |
| RunKompas | – | void | Метод запускает Компас 3D |
| BossExtrusion | ksPart, ksEntity, double, Direction\_Type | void | Метод выдавливание объекта |
| CreatePlaneOffset | ksPart, double, Obj3dType | ksEntity | Создание плоскости относительно плоскости type на расстоянии на определенном расстоянии |

# 5 Описание программы для пользователя

Макет пользовательского интерфейса представляет собой форму для ввода параметров книжного шкафа. Построение модели осуществляется путем нажатия на кнопку «Построить». При попытке ввода недопустимых символов, поле с неверными значениями будет подсвечиваться красным и при наведении на нее будет показываться сообщение ошибки.

Плагин состоит из диалогового окна, которое имеет 7 полей ввода параметров, 1 кнопку.

На рисунке 5.1 представлен пользовательский интерфейс.

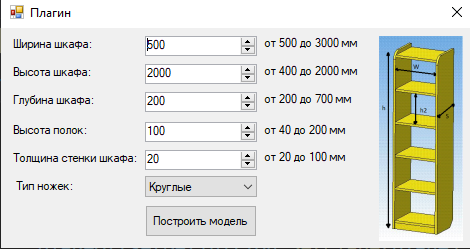


Рисунок 5.1 – Пользовательский интерфейс

После ввода необходимых параметров, построить деталь в САПР Компас 3D можно с помощью кнопки «Построить». Книжный шкаф, построенный по заданным параметрам по умолчанию в САПР Компас 3D, представлен на рисунке 5.2.

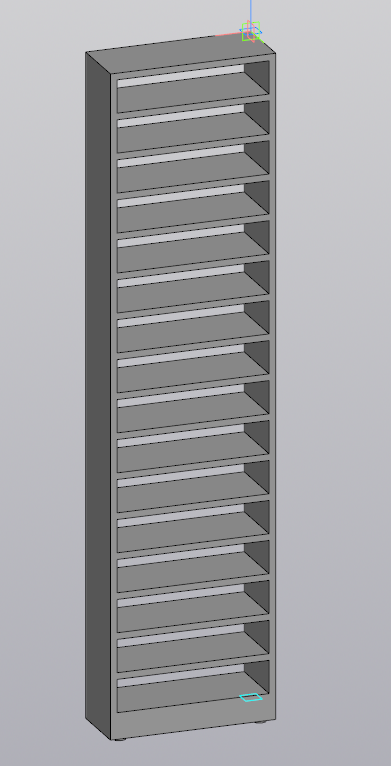


Рисунок 5.2 — Построенный книжный шкаф

# 6 Тестирование программы

Тестирование позволяет убедиться в работоспособности программы, выявлять ошибки при изменении какого-либо функционала.

## **6.1 Функциональное тестирование**

При функциональном тестировании проверялось корректность работы плагина «Книжный шкаф», а именно, соответствие полученного результата в виде трехмерной модели, с входными параметрами. [10]

Проведено тестирование максимальных и минимальных параметров модели.

На рисунках 6.1 представлен проверка размеров модели с минимальным введенными параметрами в САПР Kompas 3D (ширина 500 миллиметров, высота 2000 миллиметров, глубина шкафа 200 миллиметров, высота полок 100 миллиметров, толщина материала 20 мм).

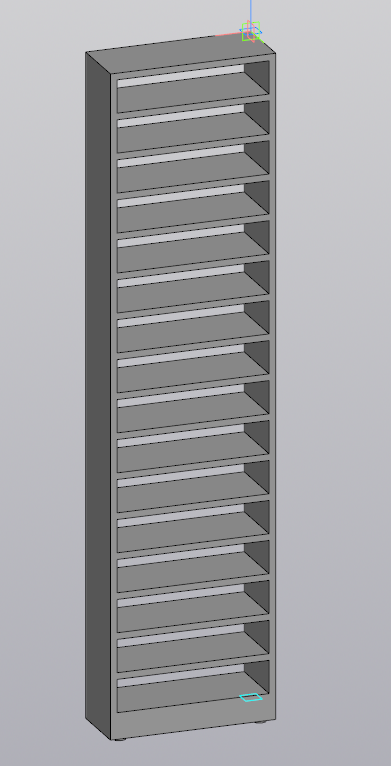


Рисунок 6.1 – Модель с минимальными введенными параметрами в Kompas 3D

Ниже на рисунках 6.2 представлена проверка размеров модели с максимальными введенными параметрами в САПР Kompas 3D (ширина 3000 миллиметров, высота 2000 миллиметров, глубина шкафа 700 миллиметров, высота полок 200 миллиметров, толщина материала 100 мм).

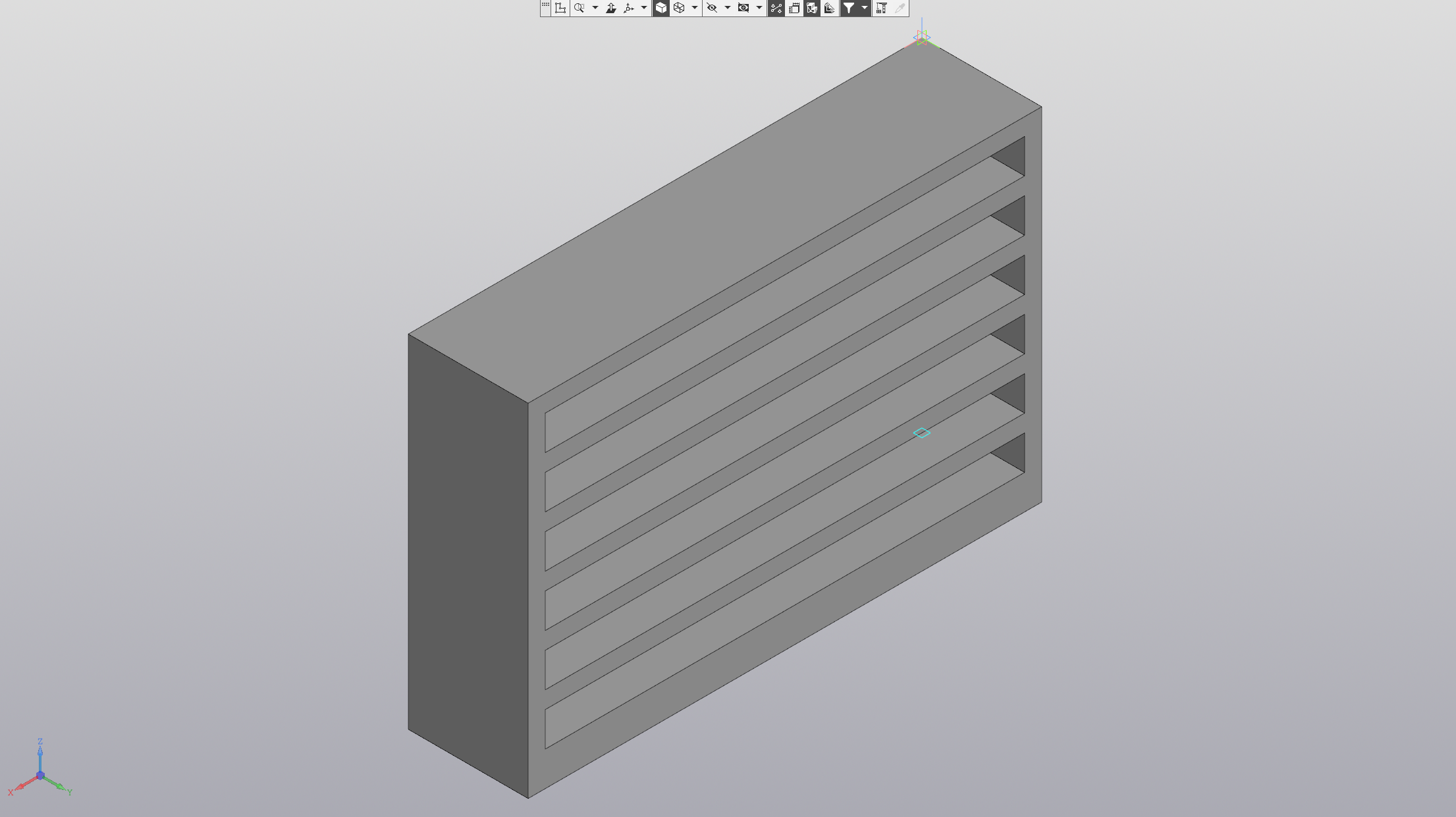


Рисунок 6.2 – Модель с максимально веденными параметрами в Kompas 3D

Ниже на рисунках 6.3 представлена проверка размеров модели с минимальными введенными параметрами в САПР Kompas 3D (ширина 300 миллиметров, высота 200 миллиметров, глубина шкафа 200 миллиметров, высота полок 40 миллиметров, толщина материала 20 мм).

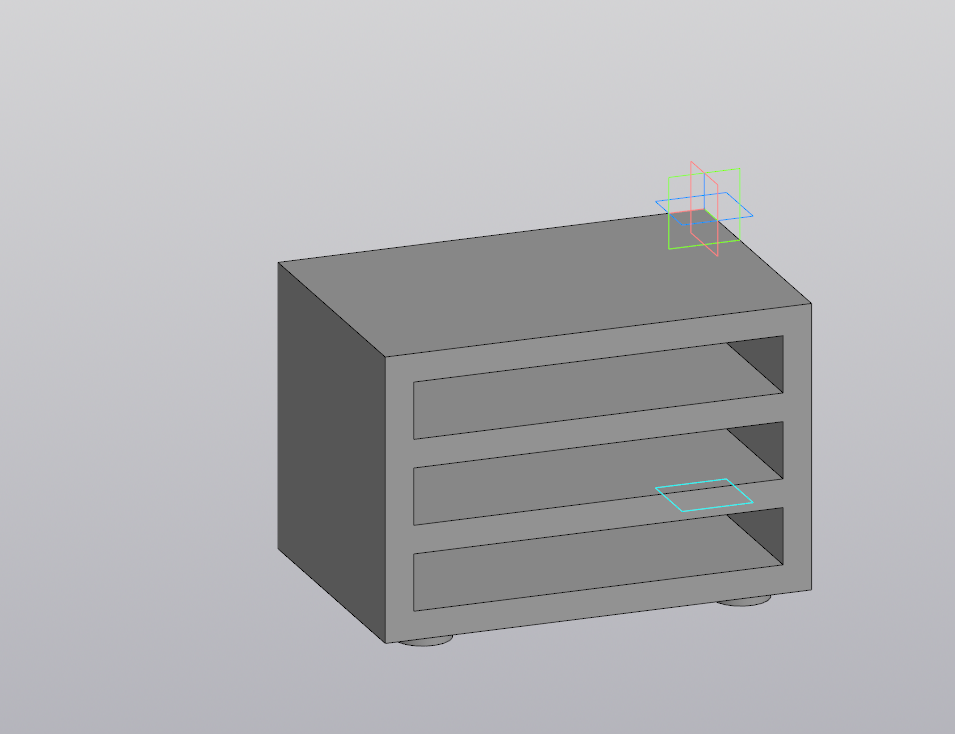
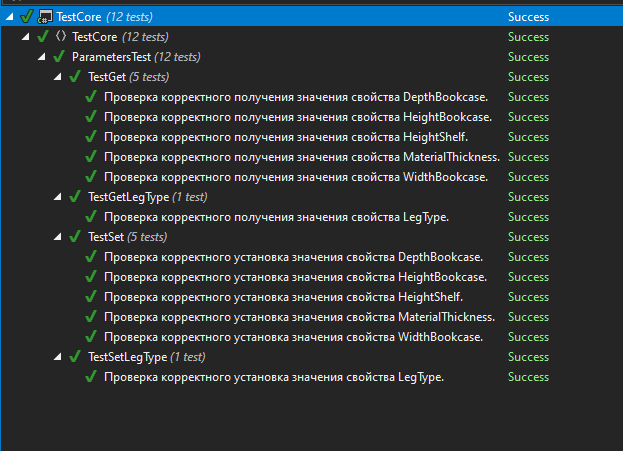


Рисунок 6.2 – Модель с минимальными веденными параметрами в Kompas 3D

## **6.2 Модульное тестирование**

В целях проверки корректности работы методов и свойств классов при помощи тестового фреймворка NUnit версии 3.13 проведено модульное тестирование [11], проверялись открытые поля и методы. На рисунке 6.3 представлено тестирование классов проекта Core. Степень покрытия проекта — сто процентов. Было написано 12 тестов.

  
Рисунок 6.3 – Тестирование классов

В таблице 6.1 перечислены все тестовые методы.

Таблица 6.1 – Тестовые методы ParametersTest

|  |  |
| --- | --- |
| Название теста | Описание |
| TestGet | Проверка корректного получения значения свойств. |
| TestSet | Проверка корректного установка значения свойств. |
| TestSetLegType | Проверка корректного установка значения свойства LegType. |
| TestGetLegType | Проверка корректного получения значения свойства LegType. |

## **6.3 Нагрузочное тестирование**

В целях проверки производительности работы плагина, было проведено нагрузочное тестирование [12]. Тестирование производилось на ПК со следующей конфигурацией:

* ЦП Intel Core I5-8300H 2.3ГГц;
* 16 ГБ ОЗУ;
* графический процессор объемом памяти 2 ГБ.

На рисунке 6.4 для проведения нагрузочного тестирования был добавлен секундомер («Stopwatch»), который засекал время от начала построения, с каждым успешным построением фигуры производилась запись результатов в текстовый файл «log.txt».

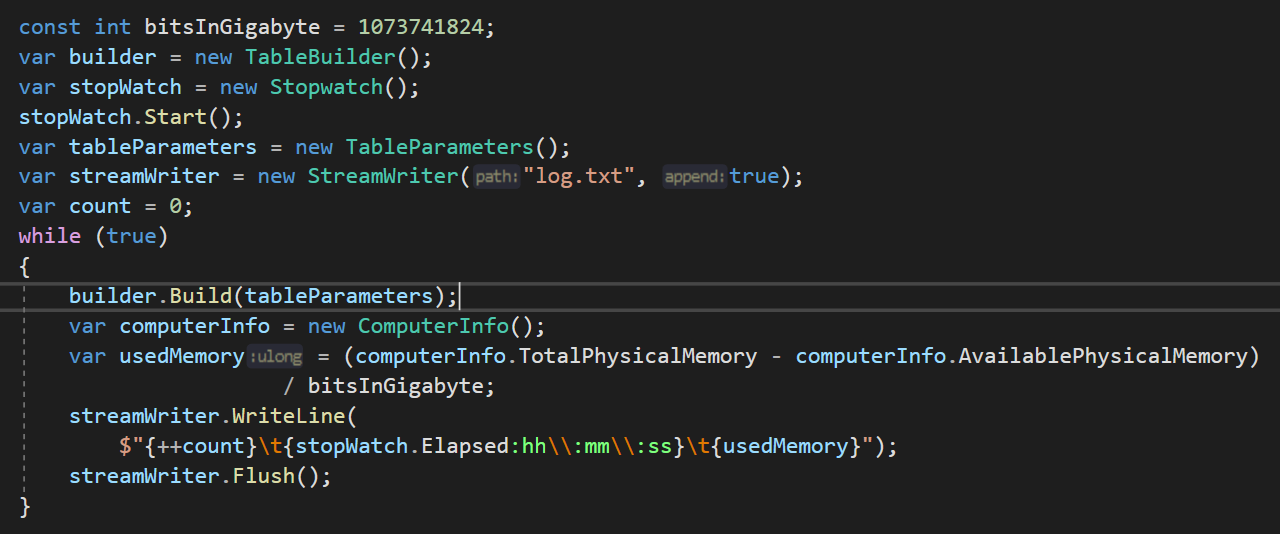


Рисунок 6.4 – Зацикливание перестроения фигуры

На графике, изображенном на рисунке 6.5 ось «Х» - количество построенных деталей, ось «Y» - количество потребляемой оперативной памяти. На графике, изображенном на рисунке 6.6 в текущей главе, ось «X» – время в секундах, ось «Y» – количество построенных деталей. На протяжении всех тестов (продолжительностью до сбоя Kompas 3D и Inventor) общая загруженность процессора была в пределах 30 процентов, потребление ОЗУ плагином прямолинейное от 13МБ до 17МБ для тестирования на САПР Kompas 3D.

На рисунке 6.5 представлено тестирование зацикленного перестроения фигуры со следующими параметрами:

* ширина 500 миллиметров;
* высота 2000 миллиметров;
* глубина шкафа 200 миллиметров;
* высота полок 100 миллиметров;
* толщина материала 20 мм.

Рисунок 6.5 – График зависимости загруженности памяти от количества деталей

Рисунок 6.6 – График зависимости времени от количества построенных деталей с параметрами по умолчанию

Исходя из вышеуказанных графиков на рисунках 6.5 и 6.6, время построения было линейно возрастало до 85 построения, потом время следующего построения увеличилось и затем вернулось в линейное возрастание. Возможно, это связано с тем, что фоном операционная система запустила задачу, что сказалось на времени построения. На рисунке 6.6 видно, что до запуска плагина, было занято около 7000 МБ оперативной памяти системой и сторонними процессами, которые к самому плагину отношения не имеют. До 55 построения занимаемая память увеличилась. Начиная с 56 построение занимаемая память изменялась в промежутке 14 до 15,5 ГБ. Возможно, это связано с тем, что Компас 3D запустил алгоритмы оптимизации.

# Заключение

В ходе выполнения лабораторных работ были изучены предметная область проектирования, предмет проектирования, аналоги предмета проектирования, API, на основании полученных данных были спроектированы UML диаграммы классов. Был разработан плагин для создания 3D моделей «Книжный шкаф» в САПР Компас 3D. Было проведено функциональное и нагрузочное тестирование плагина.

# Список использованных источников

1. Автоматизация вычислительных процедур в прикладных задачах инженерного проектирования [Электронный ресурс]. – URL: https://scienceforum.ru/2014/article/2014000201 (дата обращения: 10.04.2022).
2. Visual Studio [Электронный ресурс]. – URL: <https://visualstudio.microsoft.com/ru/> (дата обращения: 10.04.2022).
3. КОМПАС-3D. Официальный сайт САПР КОМПАС [Электронный ресурс]. — Режим доступа: https://kompas.ru/ (дата обращения: 10.04.2022).
4. 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: 10.04.2022).
5. КОМПАС-3D для разработчиков [Электронный ресурс]. – URL: https://kompas.ru/solutions/developers/ (дата обращения: 10.04.2022).
6. NUnit [Электронный ресурс]. – URL: <https://nunit.org/> (дата обращения: 10.04.2022).
7. Руководство по классическим приложениям (Windows Forms .NET) [Электронный ресурс]. – URL: https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/overview/?view=netdesktop-6.0 (дата обращения: 10.04.2022).
8. Сайт К3-Мебель [Электронный ресурс]. – https://k3-mebel.ru/ (дата обращения: 10.04.2022).
9. UML. [Электронный ресурс]. – Режим доступа: http://www.uml.org/ (дата обращения: 10.04.2022).
10. Функциональное тестирование [Электронный ресурс]. – URL: https://daglab.ru/funkcionalnoe-testirovanie-programmnogo-obespechenija/ (дата обращения: 10.04.2022).
11. Юнит-тестирование для чайников [Электронный ресурс]. – URL: <https://habr.com/ru/post/169381/> (дата обращения: 10.04.2022).
12. Нагрузочное тестирование: с чего начать и куда смотреть [Электронный ресурс]. – URL: <https://habr.com/ru/company/jugru/blog/329174/> (дата обращения: 10.04.2022).