Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение  
высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

Разработка плагина «Табурет» для САПР «КОМПАС-3D v21»

Пояснительная записка по дисциплине «Основы разработки САПР»

Выполнил:

студент гр.581

\_\_\_\_\_\_\_\_\_Качаева С.А.

« 23 » декабря 2024 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_Калентьев А. А.

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2024 г.

Томск 2024

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc186051070)

[1 ПОСТАНОВКА И АНАЛИЗ ЗАДАЧИ 4](#_Toc186051071)

[2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ 5](#_Toc186051072)

[3 ВЫБОР ИНСТРУМЕНТОВ И СРЕДСТВ РЕАЛИЗАЦИИ 6](#_Toc186051073)

[4 НАЗНАЧЕНИЕ ПЛАГИНА 7](#_Toc186051074)

[5 ОБЗОР АНАЛОГОВ 8](#_Toc186051075)

[6 ОПИСАНИЕ РЕАЛИЗАЦИИ 10](#_Toc186051076)

[7 ОПИСАНИЕ ПРОГРАММЫ ДЛЯ ПОЛЬЗОВАТЕЛЯ 20](#_Toc186051077)

[8 ТЕСТИРОВАНИЕ ПЛАГИНА 23](#_Toc186051078)

[8.1 Функциональное тестирование 23](#_Toc186051079)

[8.2 Модульное тестирование 28](#_Toc186051080)

[Список использованных источников 29](#_Toc186051081)

# Введение

Разработка и реализация современных технологических процессов невозможна без использования средств автоматизации. В настоящее время автоматизация – основной способ повышения производительности и эффективности труда инженерно-технических работников, занимающихся моделированием сложных устройств.

Применение автоматизации в процессе проектирования дает возможность разрабатывать все более сложные технические сооружения и оперативно адаптироваться к новым решениям и технологиям в различных областях инженерии. Это способствует значительному улучшению точности расчетов, выбору оптимальных вариантов реализации на основе детального математического анализа большинства или всех проектных вариантов с учетом технических, технологических и экономических характеристик создания и эксплуатации проектируемого объекта. Кроме того, автоматизация значительно улучшает качество конструкторской документации, существенно сокращает сроки проектирования и передачи документации в производство, а также позволяет более эффективно использовать технологическое оборудование с программным управлением [1]

Таким образом, целью данной работы является разработка плагина, автоматизирующего построение модели «Табурет» для системы автоматизированного проектирования КОМПАС 3D с помощью интегрированной среды разработки Visual Studio 2021.

# 1 ПОСТАНОВКА И АНАЛИЗ ЗАДАЧИ

Целью работы является разработка плагина «Табурет» для системы автоматизированного проектирования (САПР) КОМПАС-3D v21.

У плагина должны быть изменяемые параметры: длина сиденья, ширина сиденья, толщина сиденья, высота ножек, ширина и длина ножек. Все параметры указываются в миллиметрах. Если вводимые данные некорректные, то над кнопкой «Построить» в контейнере для предупреждений появятся сообщения о том, какие параметры введены неверно, а поля для ввода, в которые ввели некорректные данные, подсветятся красным цветом.

В рамках проекта были поставлены следующие задачи:

1. Создание технического задания (23.09.2024 — 08.10.2024)
2. Создание проекта системы (09.10.2024 — 29.10.2024)
3. Реализация плагина (30.10.2024 — 10.12.2024)
4. Доработка плагина (11.12.2024 — 29.12.2024)
5. Создание пояснительной записки (11.12.2024 — 29.12.2024)

# 2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ

Табурет — предмет мебели для сидения одного человека, без спинки (в отличие от стула) и подлокотников.[8]

Параметры табурета:

- L1 – Длина сиденья: от 300 до 400 мм;

- W1 – Ширина сиденья: от 300 до 600 мм;

- T1 – Толщина сиденья: от 20 до 35 мм;

- H2 – Высота ножек: от 300 до 400 мм;

- W2 – Ширина ножек: от 25 до 35 мм;

- L2 – Длина ножек: от 25 до 35 мм.

Зависимые параметры:

- L2 = W2 – Ширина ножки равняется ее длине;

- Полная высота табурета (T1+L2) должна быть больше 330 мм.

На рисунке 2.1 представлена модель табурета с параметрами.



Рисунок 2.1 – 3D модель табурета с параметрами

В качестве дополнительной функциональности были выбраны следующие пункты:

• Добавление типов сиденья: круглая и прямоугольная;

• Добавление типов ножек: круглая и квадратная.

# 3 ВЫБОР ИНСТРУМЕНТОВ И СРЕДСТВ РЕАЛИЗАЦИИ

На основе требований к техническому заданию программа выполнена на языке программирования C# в среде Microsoft Visual Studio 2021 с использованием .NET Framework, API7 для Kompas-3D.

Инструментом тестирования и создания модульных тестов был выбран тестовый фреймворк NUnit.

Для реализации пользовательского интерфейса использовалась система для построения настольных приложений Windows Forms.

# 4 НАЗНАЧЕНИЕ ПЛАГИНА

Назначение разрабатываемого плагина обусловлено быстрым моделированием табурета за счет автоматизированного проектирования по разным пользовательскими входными параметрами.

# 5 ОБЗОР АНАЛОГОВ

Косвенный аналог для плагина создания табуретов является программа проектирования мебели Объемник, с помощью которой можно осуществить также проектирование дизайна интерьера любого уровня сложности. Кроме того, эта программа для расчета и раскроя корпусной мебели, позволяющая не только создавать отдельные проекты, но и просчитывать их стоимость, что необходимо для профессиональной работы.

Программа Объемник представлена на рисунке 5.1.



Рисунок 5.1 – Программа Объемник

Вторым косвенным аналогом является конструктор мебели Астра. Каждый инструмент, каждая функция в программе подчинены одной цели — быстро, просто, качественно спроектировать мебель, создать чертежи деталей и вывести файлы на станки с ЧПУ, раскроить материалы и подготовить документацию для изготовления мебели. Применение программы эффективно как на небольших предприятиях, изготавливающих мебель на заказ, так и на крупных сервисных центрах, оказывающих услуги мебельщикам.

Программа Астра представлена на рисунке 5.2.



Рисунок 5.2 – Программа Астра

# 6 ОПИСАНИЕ РЕАЛИЗАЦИИ

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.

UML диаграмма классов для плагина «Табурет» представлена на рисунке 6.1.

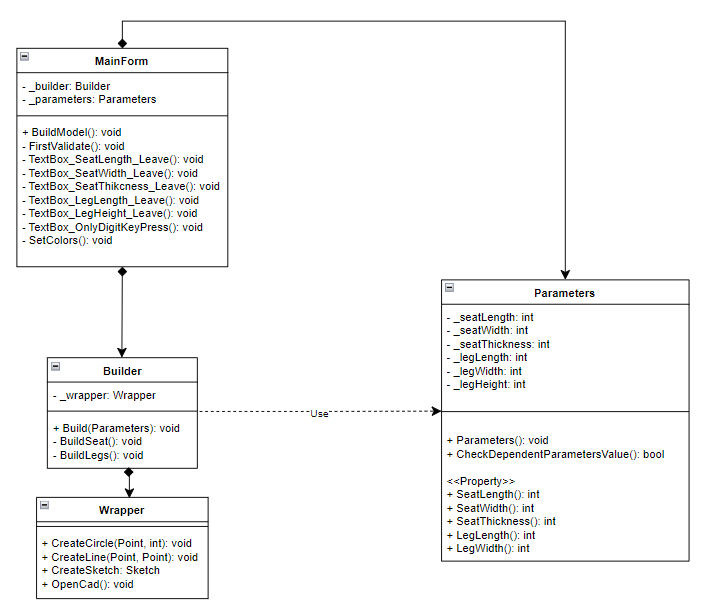


Рисунок 6.1 – UML диаграмма классов для плагина «Табурет»

В таблицах 6.1 – 6.7 представлена информация о свойствах и методов каждого класса.

Таблица 6.1 – Свойства класса MainForm

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_builder | Builder | Хранит в себе объект построения |
| \_parameters | Parameters | Хранит в себе параметры для объекта построения |

Таблица 6.2 – Методы класса MainForm

|  |  |  |
| --- | --- | --- |
| Название | Тип возвращаемых параметров | Описание |
| BuildModel | void | Запуск построения модели по заданным параметрам |
| FirstValidate | void | Первый уровень валидации |
| TextBox\_SeatLength\_Leave() | void | Валидация введенного значения длины сиденья |
| TextBox\_SeatWidth\_Leave() | void | Валидация введенного значения ширины сиденья |
| TextBox\_SeatThickness\_Leave() | void | Валидация введенного значения толщины сиденья |
| TextBox\_LegLength\_Leave() | void | Валидация введенного значения длины ножки |
| TextBox\_LegHeight\_Leave() | void | Валидация введенного значения высоты ножки |
| TextBox\_OnlyDigitKeyPress | void | Обработчик событий, позволяющий вводить в поля только цифры |
| SetColors | void | Присваивает цвет для фона поля для ввода |

Таблица 6.3 – Свойства класса Builder

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_wrapper | Wrapper | Хранит в себе объект обертки API |

Таблица 6.4 – Методы класса Builder

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых параметров | Описание |
| Build | Parameters | Void | Построение модели по заданным параметрам |
| BuildSeat | - | Void | Построение сиденья табурета |
| BuildLegs | - | void | Построение ножен табурета |

Таблица 6.5 – Методы класса Wrapper

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых параметров | Описание |
| CreateCircle | Point, int | Void | Построение окружности с заданной центральной точкой и радиусом |
| CreateLine | Point, Point | Void | Создание линии по двум точкам |
| CreateSketch | - | Void | Создание эскиза |
| OpenCAD | - | coid | Открытие Компас3D |

Таблица 6.6 – Свойства класса Parameters

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_seatLength | int | Значение длины сиденья |
| \_seatWidth | int | Значение ширины сиденья |
| \_seatThickness | int | Значение толщины сиденья |
| \_legLength | int | Значение длины ножки |
| \_legWidth | int | Значение ширины ножки |
| \_legHeight | int | Значение высоты ножки |

Таблица 6.7 – Методы класса Parameters

|  |  |  |
| --- | --- | --- |
| Название | Тип возвращаемых параметров | Описание |
| Parameters | Void | Конструктор класса |
| CheckDependentParametersValue | String | Валидация зависимых значений параметров |
| SeatLength | Int | Установка значения параметра длина сиденья |
| SeatWidth | Int | Установка значения параметра ширина сиденья |
| SeatThickness | Int | Установка значения параметра толщина сиденья |
| LegLength | Int | Установка значения параметра длина ножки |
| LegWidth | Int | Установка значения параметра ширина ножки |
| LegHeight | int | Установка значения параметра высота ножки |

Классы и методы, использованные после реализации проекта, приведены на UML-диаграмме на рисунке 6.2.

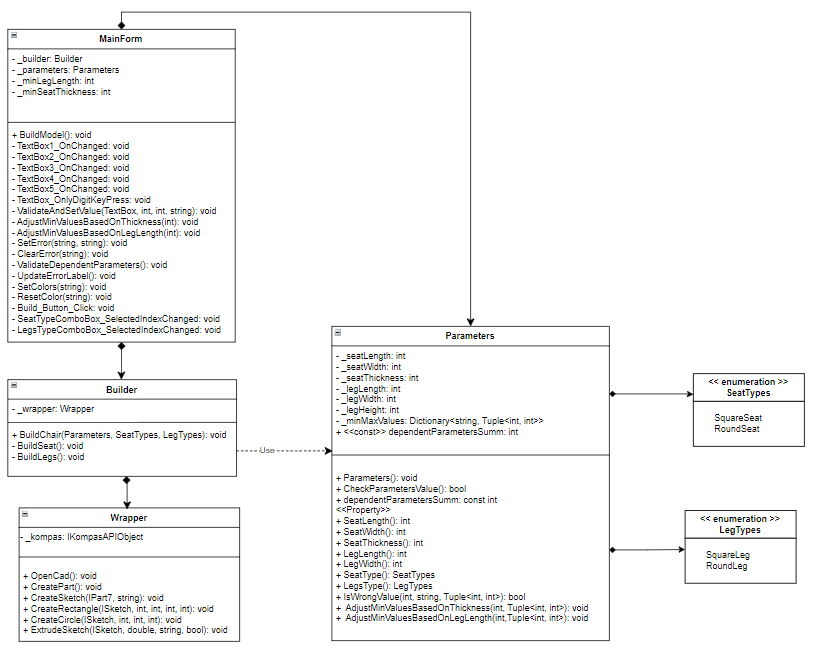


Рисунок 6.2 – UML-диаграмма после реализации проекта

В таблицах 6.8 – 6.17 представлена информация о свойствах и методах каждого класса, получившиеся после реализации проекта.

Перечисления LegTypes и SeatTypes предназначены для облегчения работы с выбранными типами ножек и сиденья соответственно.

Таблица 6.8 – Свойства перечислений LegTypes

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| SquareLeg | enum | Квадратный тип ножек |
| RoundLeg | enum | Круглый тип ножек |

Таблица 6.9 – Свойства перечислений SeatTypes

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| SquareSeat | enum | Квадратный\прямоугольный тип сиденья |
| RoundSeat | enum | Круглый тип сиденья |

Свойства \_minLegLength и \_minSeatThickness динамически изменяются после ввода в поля «Толщина сиденья» и «Высота ножки» соответственно для того, чтобы указать пользователю, какое минимальное значение можно ввести.

Таблица 6.10 – Свойства класса MainForm

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_builder | Builder | Хранит в себе объект построения |
| \_parameters | Parameters | Хранит в себе параметры для объекта построения |
| \_minLegLength | int | Хранит в себе минимальное возможное вводимое значение длины ножек |
| \_minSeatThickness | int | Хранит в себе минимальное возможное вводимое значение толщины сиденья |

Для обработки введенных значений было принято решение использовать свойство «OnChange» вместо свойства «Leave», чтобы валидация происходила в реальном времени, а не после переключения на другие поля для ввода.

Методы класса AdjustMinValuesBasedOnThickness и AdjustMinValuesBasedOnLegLength предназначены для расчета минимального значения длины ножек и толщины сиденья.

Методы SeatTypeComboBox\_SelectedIndexChange и LegsTypeComboBox\_SelectedIndexChange предназначены для обработки события выбора типов сиденья и ножек соответственно.

Таблица 6.11 – Методы класса MainForm

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых параметров | Описание |
| BuildModel | - | void | Запуск построения модели по заданным параметрам |
| TextBox1\_OnChanged | - | void | Валидация введенного значения длины сиденья |
| TextBox2\_OnChanged | - | void | Валидация введенного значения ширины сиденья |
| TextBox3\_OnChanged | - | void | Валидация введенного значения толщины сиденья |
| TextBox4\_OnChanged | - | void | Валидация введенного значения длины ножки |
| TextBox5\_OnChanged | - | void | Валидация введенного значения высоты ножки |
| TextBox\_OnlyDigitKeyPress | - | void | Обработчик событий, позволяющий вводить в поля только цифры |

Таблица 6.11 – Продолжение

|  |  |  |  |
| --- | --- | --- | --- |
| ValidateAndSetValue | TextBox, int, int, string | void | Валидация и присваивание значений из поле для ввода в параметры |
| AdjustMinValuesBasedOn  Thickness | int | void | Регулирование минимального значения длины ножек, основываясь на толщину сиденья |
| AdjustMinValuesBasedOn  LegLength | int | void | Регулирование минимального значения толщины сиденья, основываясь на длину ножек |
| SetError | String, string | void | Указание ошибки в полях для ввода в области для предупреждений |
| ClearError | String | void | Удаление ошибки из области с предупреждениями |
| UpdateErrorLabel | - | void | Обновление списка с ошибками в поле с предупреждениями |
| ValidateDependentParameters | - | void | Валидация зависимых параметров |
| SetColors | String | void | Присваивает цвет для фона поля для ввода с некорректным значением |

Таблица 6.11 – Продолжение

|  |  |  |  |
| --- | --- | --- | --- |
| ResetColor | String | void | Возвращает полю для ввода исходный цвет |
| Build\_Button\_Click | - | void | Валидация значений параметров и построение табурета |
| SeatTypeComboBox\_  SelectedIndexChange | - | void | Обработка события выбора типа сиденья |
| LegsTypeComboBox\_  SelectedIndexChange | - | void | Обработка события выбора типа ножек |

Таблица 6.12 – Свойства класса Builder

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_wrapper | Wrapper | Хранит в себе объект обертки API |

В метод Build были добавлены входные параметры SeatTypes и LegTypes, чтобы плагин построил табурет в соответствии с выбранными типами сиденья и ножек.

Таблица 6.13 – Методы класса Builder

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых параметров | Описание |
| Build | Parameters, SeatTypes, LegTypes | Void | Построение модели по заданным параметрам |
| BuildSeat | - | Void | Построение сиденья табурета |
| BuildLegs | - | void | Построение ножен табурета |

Таблица 6.14 – Свойства класса Wrapper

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_kompas | IKompasAPIObject | Обертка для взаимодействия с API Kompas |

Был добавлен метод ExtrudeSketch для выдавливания эскиза.

Таблица 6.15 – Методы класса Builder

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых параметров | Описание |
| CreateCircle | ISketch, int, int, int, int | Void | Построение окружности с заданной центральной точкой и радиусом |
| CreateRectangle | ISketch, int, int, int, int | void | Построение прямоугольника |
| CreatePart | - | Void | Создание части в 3D-документе |
| CreateSketch | IPart7, string | Void | Создание эскиза |
| ExtrudeSketch | ISketch, double, string, bool | void | Выдавливание эскиза |
| OpenCAD | - | void | Открытие Компас3D |

Свойства \_legsType и \_seatType хранят в себе выбранное значение типа ножки и сиденья соответственно.

Таблица 6.16 – Свойства класса Parameters

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_seatLength | int | Значение длины сиденья |
| \_seatWidth | int | Значение ширины сиденья |
| \_seatThickness | int | Значение толщины сиденья |
| \_legLength | int | Значение длины ножки |
| \_legWidth | int | Значение ширины ножки |
| \_legHeight | int | Значение высоты ножки |

Таблица 6.16 – Продолжение

|  |  |  |
| --- | --- | --- |
| \_legsType | LegTypes | Тип ножек |
| \_seatType | SeatTypes | Тип сиденья |

Таблица 6.17 – Методы класса Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых параметров | Описание |
| Parameters | - | Void | Конструктор класса |
| CheckDependentParametersValue | - | String | Валидация зависимых значений параметров |
| SeatLength | - | Int | Установка значения параметра длина сиденья |
| SeatWidth | - | Int | Установка значения параметра ширина сиденья |
| SeatThickness | - | Int | Установка значения параметра толщина сиденья |
| LegLength | - | Int | Установка значения параметра длина ножки |

Таблица 6.17 – Продолжение

|  |  |  |  |
| --- | --- | --- | --- |
| LegWidth | - | Int | Установка значения параметра ширина ножки |
| LegHeight | - | Int | Установка значения параметра высота ножки |
| LegsType | - | LegTypes | Установка типа ножек |
| SeatType | - | SeatTypes | Установка типа сиденья |
| IsWrongValue | int, string, Tuple<int, int> | Bool | Валидация значений свойств |
| AdjustMinValuesBasedOnThickness | int, Tuple<int, int> | void | Изменение границ значения толщины сиденья |
| AdjustMinValuesBasedOnLegLength | int, Tuple<int, int> | void | Изменение границ значения высоты ножек |

# 7 ОПИСАНИЕ ПРОГРАММЫ ДЛЯ ПОЛЬЗОВАТЕЛЯ

После запуска плагина открывается макет пользовательского интерфейса. Макет представляет собой форму для ввода параметров табурета и выбора типа ножек и типа сиденья, в зависимости от которых меняются поля для ввода. На рисунке 7.1 представлен внешний вид формы при выборе круглых ножек и сиденья, на рисунке 7.2 представлен внешний вид формы при выборе прямоугольного сиденья и квадратных ножек.

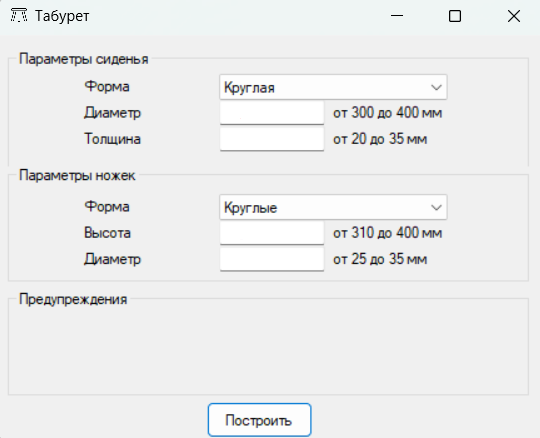


Рисунок 7.1 – Внешний вид формы при выборе круглых ножек и сиденья

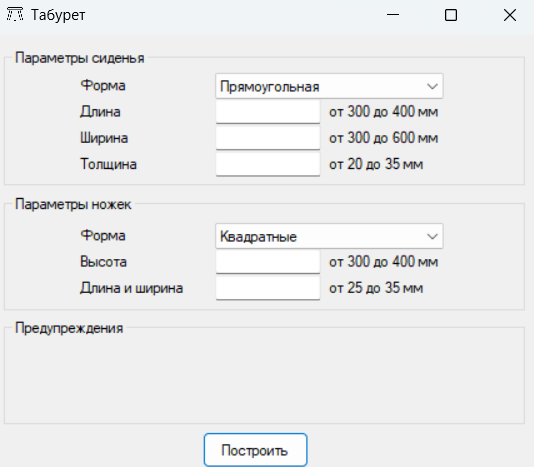


Рисунок 7.2 – Внешний вид формы при выборе прямоугольного сиденья и квадратных ножек

На рисунках 7.3-7.4 представлен внешний вид выпадающих списков.

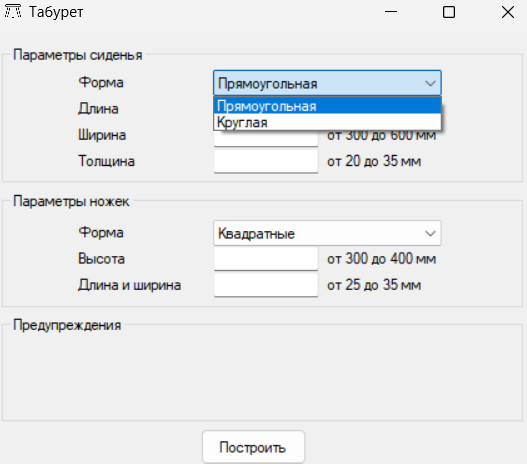


Рисунок 7.3 – Внешний вид формы с развернутым выпадающем списком с формами сиденья

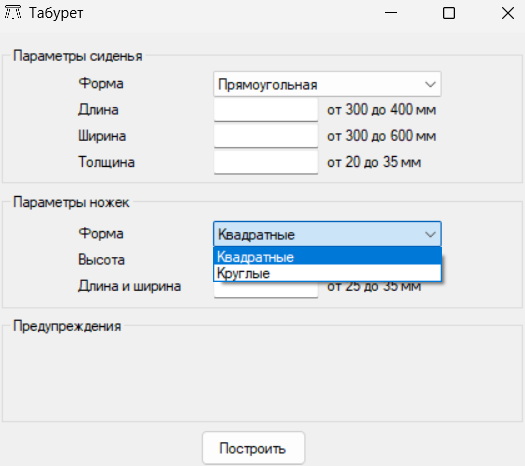


Рисунок 7.4 – Внешний вид формы с развернутым выпадающем списком с формами ножек

Интерфейс с корректно введенными данными и построенной моделью представлен на рисунке 7.5.

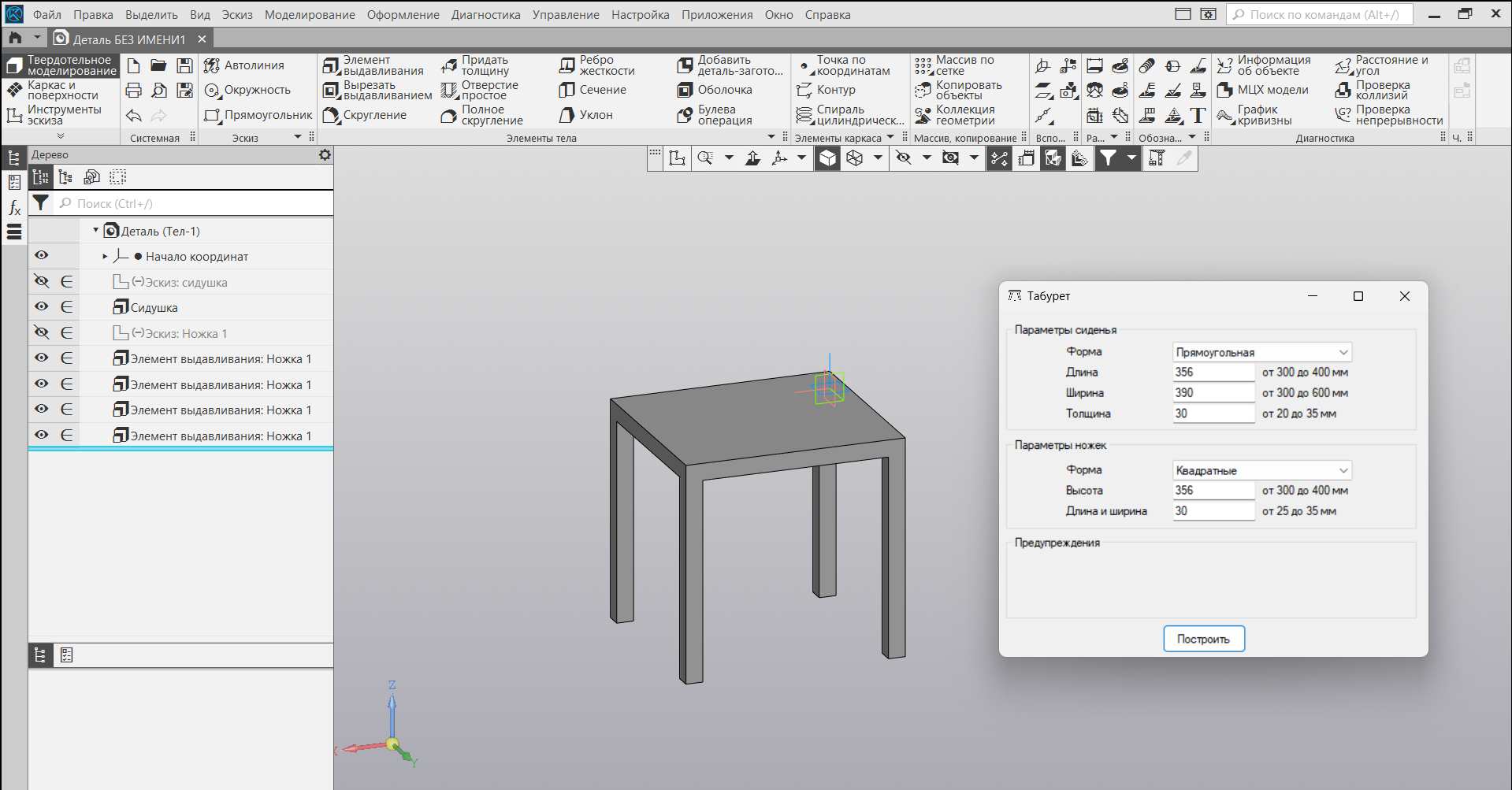


Рисунок 7.5 – Интерфейс с корректно введенными данными и построенной моделью

# 8 ТЕСТИРОВАНИЕ ПЛАГИНА

## 8.1 Функциональное тестирование

После ввода некорректных данных, например, значение параметра не входит в диапазон или поле для ввода пустое, на форме поле с некорректным параметром будет выделено красным цветом и поле «Предупреждения» заполнится строкой с ошибками, что представлено на рисунке 8.1. На рисунке 8.2 представлен вывод сообщения при введенных значениях, которые не подходят условиям зависимых параметров.

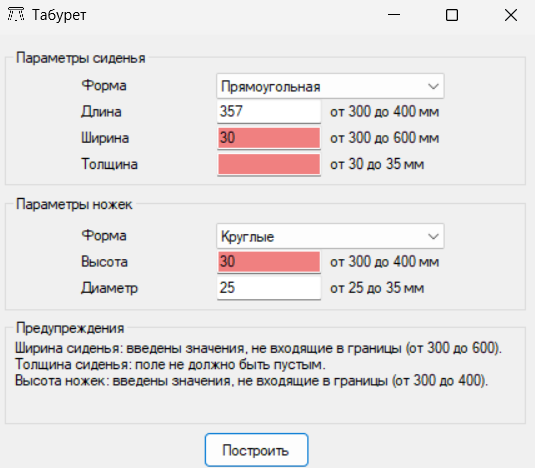


Рисунок 8.1 – Внешний вид формы при выборе прямоугольного сиденья и квадратных ножек

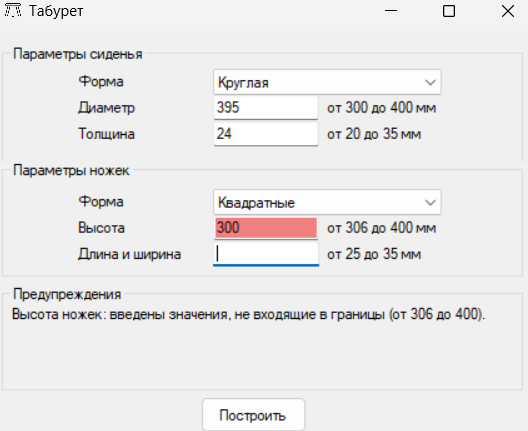


Рисунок 8.2 – Внешний вид формы с введенными значениями, не подходящими условиям зависимых параметров

В таблице 8.1 представлены значения с минимальными и максимальными параметрами для прямоугольной формы сиденья и круглой формы ножек. На рисунках 8.3 – 8.5 представлено построение модели по данным параметрам.

Таблица 8.1 – Значения с минимальными и максимальными параметрами для прямоугольной формы сиденья и круглой формы ножек

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Значения  параметров | Длина сиденья | Ширина сиденья | Толщина сиденья | Высота ножек | Диаметр ножек |
| Минимальные | 300 | 300 | 30 | 300 | 25 |
| 20 | 310 |
| Максимальные | 400 | 600 | 35 | 400 | 35 |

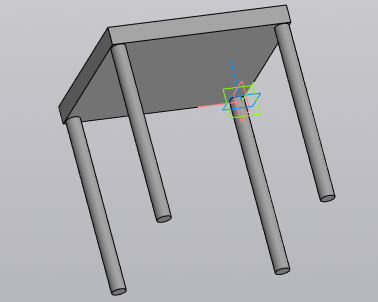


Рисунок 8.3 – Построение модели с минимальными значениями (толщина сиденья = 30, высота ножек = 300)

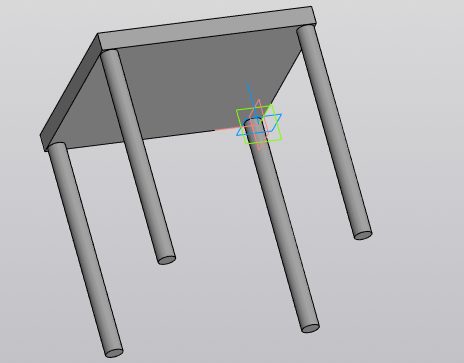


Рисунок 8.4 – Построение модели с минимальными значениями (толщина сиденья = 20, высота ножек = 310)

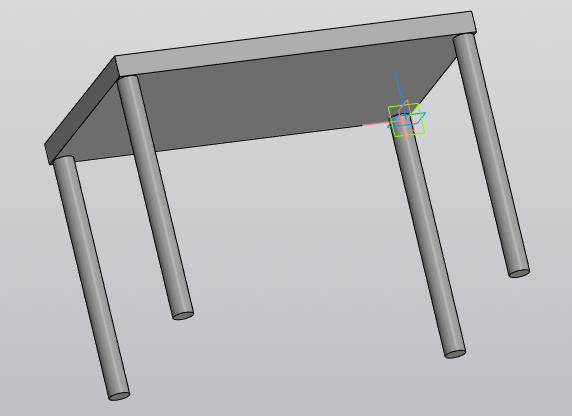


Рисунок 8.5 – Построение модели с максимальными значениями

В таблице 8.2 представлены значения с минимальными и максимальными параметрами для круглой формы сиденья и квадратной формы ножек. На рисунках 8.5 – 8.7 представлено построение модели по данным параметрам

Таблица 8.2 – Значения с минимальными и максимальными параметрами для круглой формы сиденья и квадратной формы ножек

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Значения  параметров | Диаметр сиденья | Толщина сиденья | Высота ножек | Длина и ширина ножек |
| Минимальные | 300 | 20 | 310 | 25 |
| 30 | 300 |
| Максимальные | 400 | 35 | 400 | 35 |

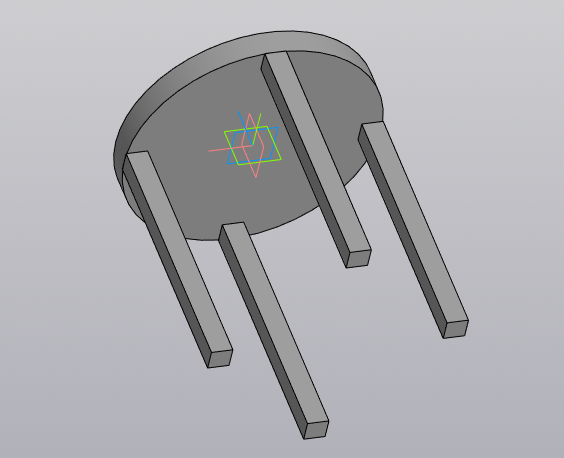


Рисунок 8.5 – Построение модели с минимальными значениями (толщина сиденья = 30, высота ножек = 300)

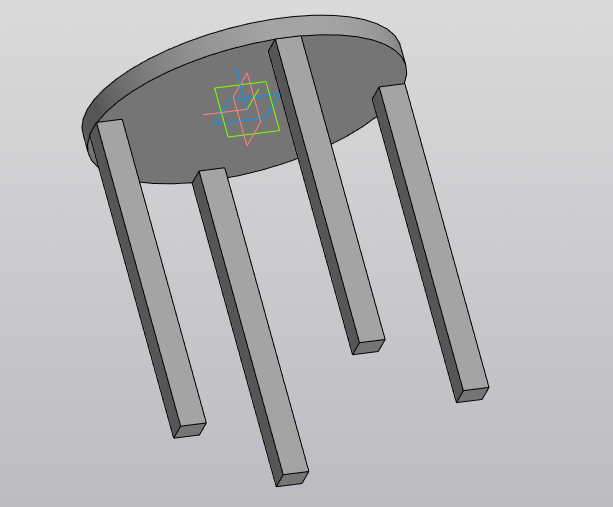


Рисунок 8.6 – Построение модели с минимальными значениями (толщина сиденья = 20, высота ножек = 310)

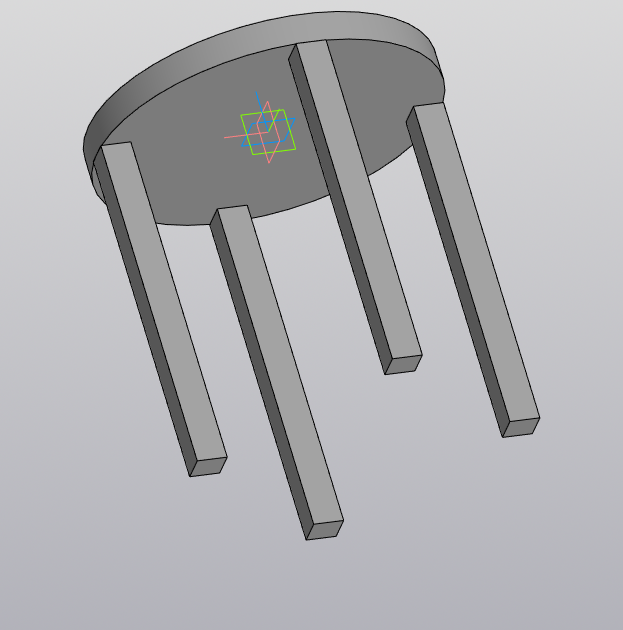


Рисунок 8.7 – Построение модели с максимальными значениями

## 8.2 Модульное тестирование

В целях проверки корректности работы методов и свойств класса Parameters при помощи фреймворка для модульного тестирования NUnit версии 3.14 было проведено модульное тестирование открытых свойств и методов. Покрытие модели тестами составило 100%, что представлено 8.8.

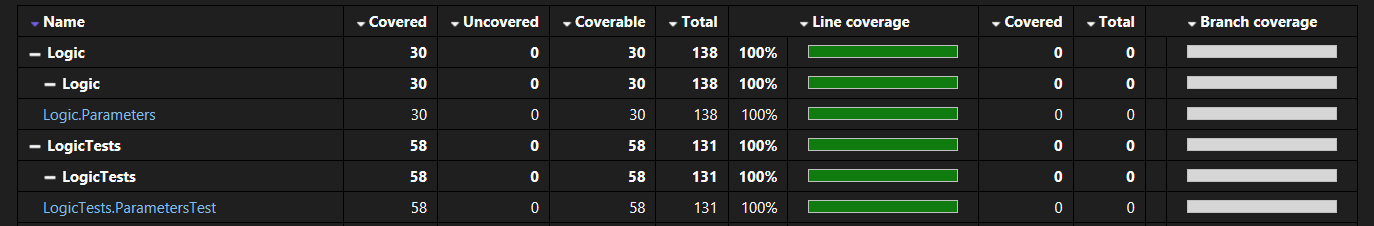


Рисунок 8.8 – Покрытие тестами класса Parameters

В таблице 8.3 представлены тесты класса Parameters.

Таблица 8.3 – Тесты класса Parameters

|  |  |
| --- | --- |
| Название | Описание |
| SeatLengthTest() | Тесты модели параметров |
| SeatWidthTest() | Тест присваивания значения длины сиденья |
| SeatThicknessTest() | Тест присваивания значения ширины сиденья |
| LegLengthTest() | Тест присваивания значения толщины сиденья |
| LegWidthTest() | Тест присваивания значения длины ножек |
| LegHeightTest() | Тест присваивания значения ширины ножек |
| ValidateTest() | Тест присваивания значения высоты ножек |
| LegsTypeTest() | Тест присваивания значений зависимых параметров |

## 8.3 Нагрузочное тестирование

Нагрузочное тестирование плагина «Табурет» проведено на ПК со следующей конфигурацией:

− процессор Intel Core i3-1115G4;

− оперативная память объемом 8 ГБ (доступно 7,8 ГБ);

− видеокарта Intel(R) UHD Graphics;

− операционная система Майкрософт Windows 11 Pro. Тестирование заключалось в построении 137 табуретов с минимальными параметрами. После 137 итерации произошел сбой в работе программы КОМПАС. Элементы панели задач начинают сильно дергаться, при наведении на нее курсора мыши. На рисунках 8.9 и 8.10 представлены получившиеся графики.

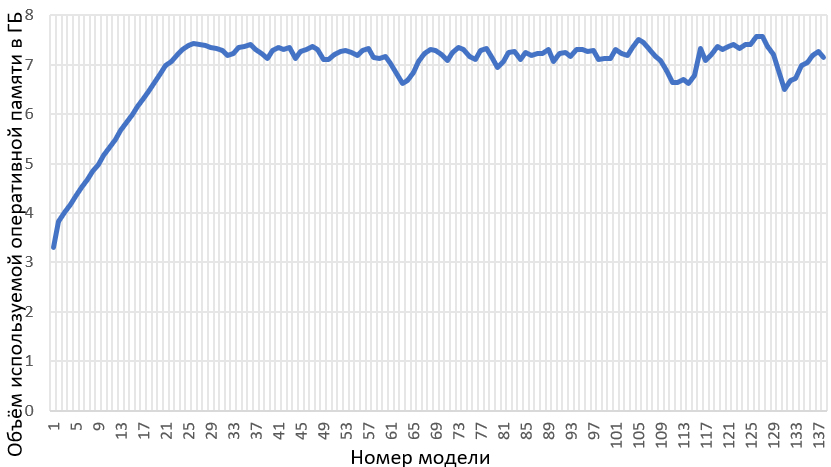


Рисунок 8.9 − График зависимости загруженности памяти от количества построенных моделей

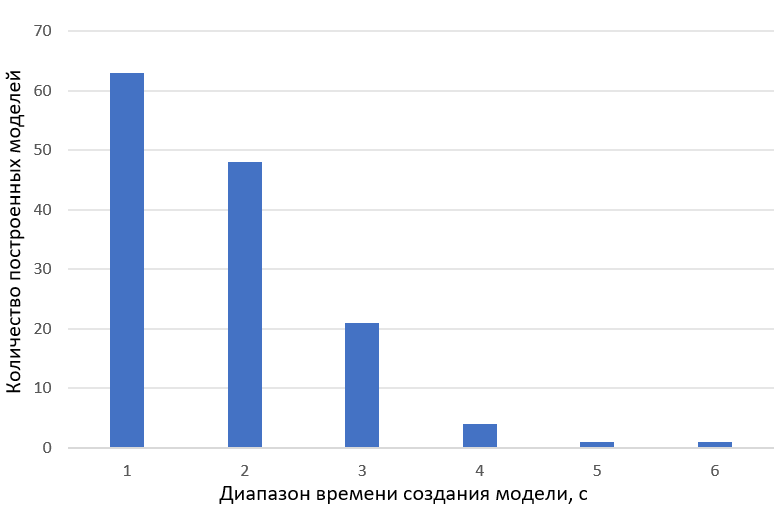


Рисунок 8.10 − График гистограммы построения модели

Проанализировав график, изображенный на рисунке 8.9, можно сделать

вывод, что объем оперативной памяти, затрачиваемый плагином на построение трехмерных моделей, линейно увеличивается до достижения предела объема оперативной памяти. По окончании свободного места оперативная память частично очищается, после чего данный процесс повторяется. Предположительно, это объясняется использованием файла подкачки для компенсации недостатка оперативной памяти.

Проанализировав график, изображенный на рисунке 8.10, можно сделать вывод, что основное время построение модели от 1 до 2 секунд. Это может быть связано с тем, что модель является достаточно простой, из-за чего даже при значительных количествах моделей их нагрузки на оперативную память и процессор недостаточно для замедления построения моделей. Время построения более 2 секунд можно связать с загруженностью ОС другими задачами, которые находятся в фоновом режиме.

# Заключение

В процессе разработки приложения был создан плагин, позволяющий создавать 3D-модели табуретов в САПР КОМПАС-3D v21.

Для разработки были изучены:

* Компас API7 для программного взаимодействия с САПР КОМПАС-3D v21;
* Библиотека NUnit для написания Unit-тестов.

# Список использованных источников

1. Использование системы трёхмерного моделирования в программе КОМПАС-3D – Мультиурок. [Электронный ресурс]. – Режим доступа: https://multiurok.ru/files/ispolzovanie-sistemy-triokhmernogo-modelirovaniia.html (дата обращения: 14.10.2024).

2. Что такое API и как он работает – Skillbox [Электронный ресурс]. – Режим доступа: <https://skillbox.ru/media/code/chto_takoe_api/> 23741 (дата обращения: 04.10.2024).

3. Компонент сборки (деталь или подсборка). Интерфейсы ksPart и IPart. [Электронный ресурс]. – Режим доступа: https://help.ascon.ru/KOMPAS\_SDK/22/ru-RU/kspart.html (дата обращения: 20.10.2024).

4. Справочная система SDK КОМПАС-3D – Российское инженерное ПО АСКОН. [Электронный ресурс]. – Режим доступа: https://help.ascon.ru/KOMPAS\_SDK/22/ru-RU/index.html (дата обращения: 06.10.2024).

5. Kompas 8: создание прикладных библиотек – Форум программистов Vingrad. [Электронный ресурс]. – Режим доступа: https://forum.vingrad.ru/topic-194348.html (дата обращения: 06.10.2024).

6. Программа "Объемник - мебельное предприятие" – Программа "Объемник - мебельное предприятие" [Электронный ресурс]. – Режим доступа: https://mebelsoft.org/ (дата обращения: 06.10.2024).

7. Астра Конструктор Мебели – Астра Конструктор Мебели. [Электронный ресурс]. – Режим доступа: https://astra-km.ru/software/astra-konstruktor-mebeli/ (дата обращения: 06.10.2024).

8. Электронный фонд правовых и нормативно-технических документов [Электронный ресурс]. – Режим доступа: https://docs.cntd.ru/document/1200017702?ysclid=m2hacx3pul675779226 (дата обращения: 20.10.2024).

9. Фаулер M. UML. Основы, 3-е издание. – Пер. с англ. – СПб: СимволПлюс, 2004. – 192 с.