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)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР»

Выполнил:

студент гр.581

\_\_\_\_\_\_\_\_\_Глотов Д.Д.

«13» января 2025 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_Калентьев А. А.

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2025 г.

Томск 2024

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc187435603)

[1 ПОСТАНОВКА И АНАЛИЗ ЗАДАЧИ 4](#_Toc187435604)

[2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ 5](#_Toc187435605)

[3 ВЫБОР ИНСТРУМЕНТОВ И СРЕДСТВ РЕАЛИЗАЦИИ 6](#_Toc187435606)

[4 НАЗНАЧЕНИЕ ПЛАГИНА 7](#_Toc187435607)

[5 ОБЗОР АНАЛОГОВ 8](#_Toc187435608)

[6 ОПИСАНИЕ РЕАЛИЗАЦИИ 10](#_Toc187435609)

[7 ОПИСАНИЕ ПРОГРАММЫ ДЛЯ ПОЛЬЗОВАТЕЛЯ 19](#_Toc187435610)

[8 ТЕСТИРОВАНИЕ ПЛАГИНА 22](#_Toc187435611)

[8.1 Функциональное тестирование 22](#_Toc187435612)

[8.2 Модульное тестирование 27](#_Toc187435613)

[8.3 Нагрузочное тестирование 28](#_Toc187435614)

[Заключение 31](#_Toc187435615)

[Список используемых источников 32](#_Toc187435616)

# Введение

Современные технологические процессы невозможны без автоматизации, которая является ключом к повышению производительности и эффективности труда инженерных работников, занимающихся моделированием сложных устройств. Автоматизация в проектировании позволяет разрабатывать сложные конструкции, быстро адаптироваться к новым технологиям и существенно повышать точность расчетов, выбирая оптимальные варианты на основе детального математического анализа. Она также улучшает качество конструкторской документации, сокращает сроки проектирования и ускоряет передачу документации в производство, а также способствует более эффективному использованию программируемого оборудования.

Цель данной работы — разработка плагина для автоматизации создания модели «Стол» в системе КОМПАС 3D с использованием Visual Studio 2022.

# 1 ПОСТАНОВКА И АНАЛИЗ ЗАДАЧИ

Целью работы является разработка плагина «Стол» для системы автоматизированного проектирования (САПР) КОМПАС-3D v22.

У плагина есть изменяемые параметры:

− Ширина столешницы;

− Глубина столешницы;

− Толщина столешницы;

− Ширина ножки;

− Высота стола.

Если данные введены неверно, то над кнопкой «Построить» появятся сообщения о том, какие параметры введены неверно, а поля для ввода будут окрашены 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

Стол — предмет мебели, имеющий приподнятую горизонтальную или наклонную поверхность и предназначенный для размещения предметов, выполнения работ, принятия пищи, игр, рисования, обучения и другой деятельности.

***Изменяемые параметры для плагина*** (также все обозначения показаны на рисунке 2.1):

− Ширина столешницы a (от 500мм до 5000мм);

− Глубина столешницы b (от 500мм до 5000мм);

− Зазор между ножками a – 2d и b – 2c (величина считается автоматически, не должна быть меньше 200мм)

− Ширина ножки c == d (от 20 до 200) (Площадь ()) сечения всех ножек должна быть не больше площади столешницы, с учетом минимального зазора между ножками);

− Высота столешницы h (от 16мм до 100мм);

− Высота стола L(от 500мм до 1400мм).



Рисунок 2.1 − Модель стола с размерами

В качестве дополнительной функциональности было выбрано расширение плагина для его использования с Autodesk Inventor.

# 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.

Для реализации пользовательского интерфейса использовалась система для построения настольные приложения Windows Forms.

Инструментом для и создания модульных тестов была выбрана библиотека для тестирования NUnit [1] версии 3.13.

# 4 НАЗНАЧЕНИЕ ПЛАГИНА

Цель разрабатываемого плагина заключается в быстром моделировании столов, что достигается посредством автоматизированного проектирования на основе различных входных пользовательских параметров.

# 5 ОБЗОР АНАЛОГОВ

Прямым аналогом для плагина создания стола является плагин для программы SketchUp под названием CraftReports. В этом плагине есть уже готовые предметы мебели, в том числе и столы, которые можно редактировать.

К плюсам можно отнести то, что сама программа SketchUp достаточно популярна и распространена среди новичков в 3D моделировании. Из минусов- для работы плагина требуется интернет-соединение, чтобы подгружать библиотеку моделей. Интерфейс программы и плагина представлен на рисунке 5.1.

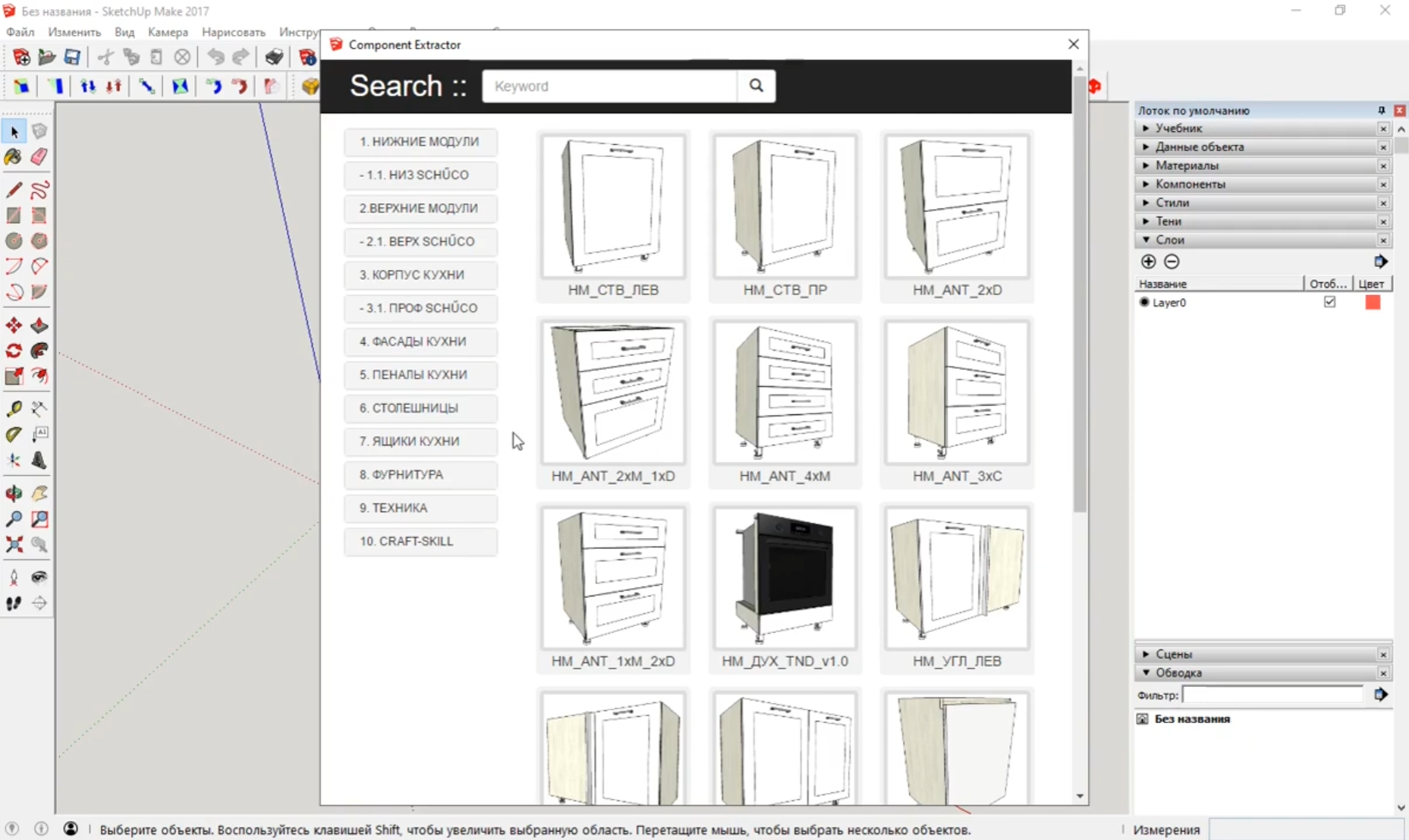


Рисунок 5.1 – Плагин CraftReport для SketchUp

Косвенным аналогом является программа PRO100[3], представляет из себя программу для 3D проектирования мебели и интерьера. Она за короткое время позволяет проектировать мебель и интерьеры помещений, дает красивую качественную картинку, автоматически считает стоимость проекта. Для данной программы также существуют библиотеки мебели и столов, модели из которых можно использовать и редактировать.

Плюсы программы: так же, как и у предыдущего аналога большое число пользователей, и так же большое число библиотек мебели. Минусы: так же нет простого и легкого редактирования параметров мебели, весь этот процесс происходит через среду моделирования. Интерфейс программы представлен на рисунке 5.2.

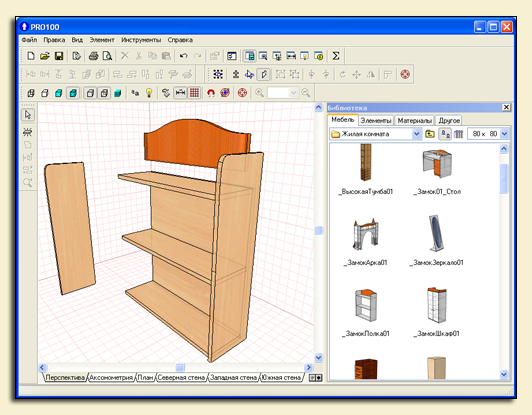


Рисунок 5.2 – Интерфейс программы Pro100

Другим примером косвенного аналога является Объемник. Это полнофункциональная программа для мебельного проектирования. Обширная библиотека содержит набор типовых элементов с изменяемыми габаритными и конструкционными параметрами. Кроме того, конструктор может самостоятельно создавать и добавлять в базу новые элементы, формировать собственную базу данных из разработанных проектов.

Плюсы программы: простой интерфейс, подходящий для новичков и опытных специалистов, библиотека мебели с возможностью редактирования. Минусы: небольшое количество пользователей по сравнению с предыдущими аналогами.

# 6 ОПИСАНИЕ РЕАЛИЗАЦИИ

Перед тем, как приступить к разработке системы была разработана ее архитектура, чтобы понять, какие сущности, и за что будут отвечать в системе.

Архитектура была разработана с учетом плагина как отдельного приложения, которое бы запускало САПР Компас-3D. Изначальная диаграмма классов представлена на рисунке 6.1.

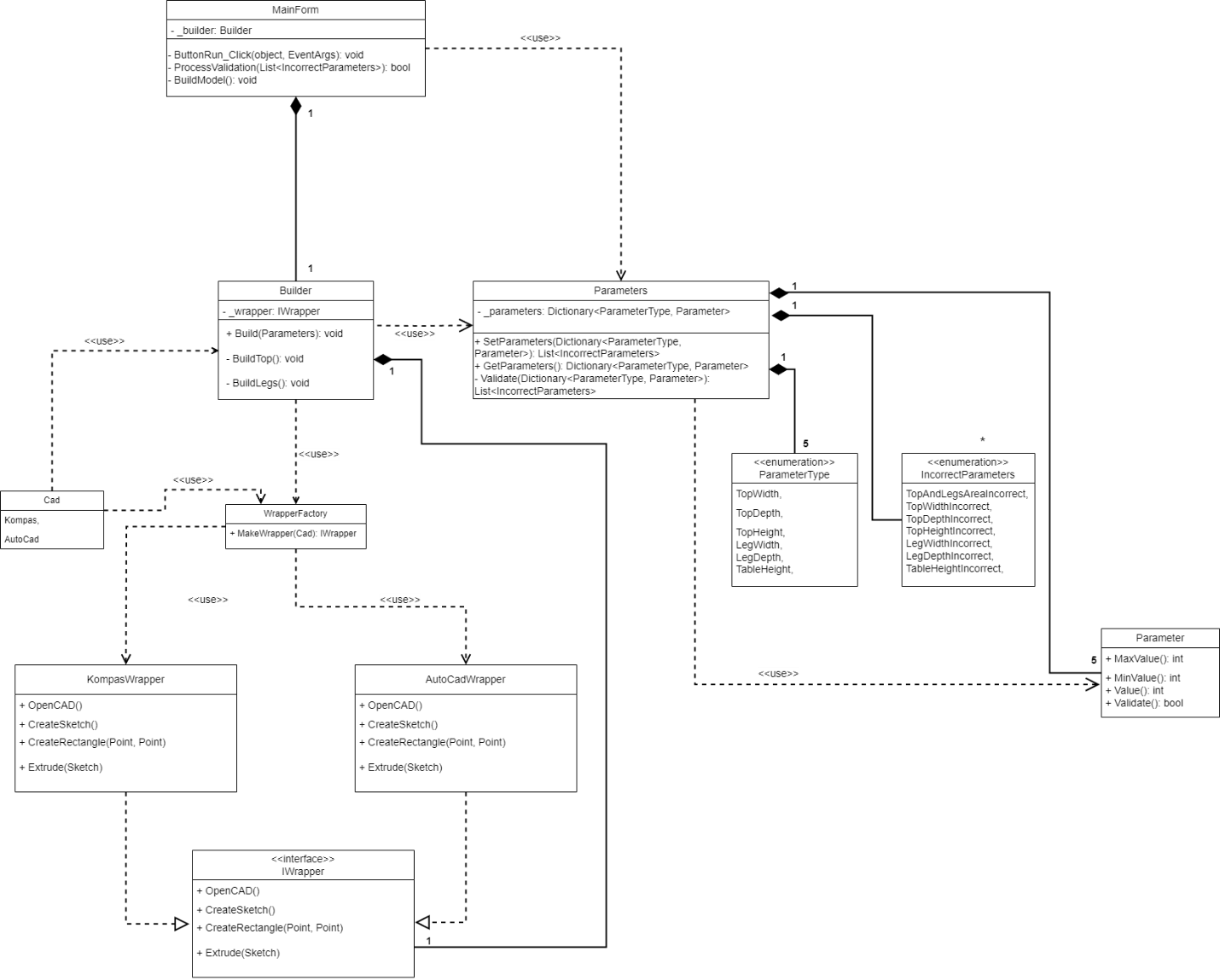


Рисунок 6.1 – Изначальная диаграмма классов плагина

Класс MainForm представляет собой объект, через который будет осуществляться обработка пользовательского ввода и передача его в модель (таблица 6.1).

Таблица 6.1 – Свойства и методы класса MainForm

|  |  |
| --- | --- |
| Название | Описание |
| \_builder | Хранит экземпляр объекта для построения модели |
| \_parameters | Хранит экземпляр объекта «Параметры» для хранения параметров и их валидации |
| BuildModel | Строит модель искомой детали |
| ButtonRun\_click | Метод нажатия на кнопку “Построить” |
| ProcessValidation | Обработать неправильные параметры, вернувшиеся из SetParameters() |

Класс Parameters представляет собой объект, хранящий параметры изделия и предоставляет функционал их валидации (таблица 6.2).

Таблица 6.2 – Свойства и методы класса Parameters

|  |  |
| --- | --- |
| Название | Описание |
| \_parameters | Хранит словарь параметров |
| SetParameters | Установить параметры. Внутри происходит вызов Validate |
| GetParameters | Получить словарь параметров |
| Validate | Проверяет значения из словаря |

Перечисление ParameterType типов параметров (таблица 6.3).

Таблица 6.3 – Описание перечисления ParameterType

|  |  |
| --- | --- |
| Название | Описание |
| TopWidth | Ширина столешницы |
| TopDepth | Глубина столешницы |
| TopHeight | Высота столешницы |
| LegWidth | Ширина ножки |
| LegDepth | Глубина ножки |
| TableHeight | Высота стола |

Класс Parameter является представлением параметра, который отвечает за первичную валидацию параметра и дальнейшее использование параметра на уровне логики (таблица 6.4).

Таблица 6.4 – Свойства и методы класса Parameter

|  |  |
| --- | --- |
| Название | Описание |
| MaxValue | Максимальное разрешенное значение параметра |
| MinValue | Минимальное разрешенное значение параметра |
| Value | Значение параметра |
| Validate | Валидирует значение параметра относительно минимального и максимального |

Класс Builder использует методы для построения фигур, реализованные в классах ниже, для построения полноценной модели (таблица 6.5).

Таблица 6.5 – Свойства и методы класса Builder

|  |  |
| --- | --- |
| Название | Описание |
| \_wrapper | Хранит ссылку на базовый интерфейса построителя примитивов, в котором реализовано использование API САПР |
| Build | Построить модель |
| BuildTop | Построить столешницу |
| BuildLegs | Построить ножки |

Перечисление Cad определяет все возможные САПР, которые будут использоваться в плагине (таблица 6.6).

Таблица 6.6 – Описание перечисления Cad

|  |  |
| --- | --- |
| Название | Описание |
| Kompas | САПР Компас-3D |
| AutoCad | САПР Autodesk AutoCad |

Класс WrapperFactory представляет из себя фабрику объектов, реализующих интерфейс IWrapper (таблица 6.7).

Таблица 6.7 – Методы класса WrapperFactory

|  |  |
| --- | --- |
| Название | Описание |
| MakeWrapper | Принимает САПР, возвращает класс, реализующий IWrapper для этой САПР. |

Классы KompasWrapper и InventorWrapper реализуют интерфейс IWrapper для определенной САПР (таблица 6.8).

Таблица 6.8 – Методы классов KompasWrapper и InventorWrapper

|  |  |
| --- | --- |
| Название | Описание |
| OpenCad | Открыть САПР |
| CreateSketch | Создать Эскиз |
| CreateRectangle | Начертить прямоугольник заданных размеров |
| Extrude | Выдавить эскиз |

Перечисление IncorrectParameters представляет из себя все возможные ошибки при валидации (таблица 6.9).

Таблица 6.9 – Перечисление IncorrectParameters неверных значений параметров

|  |  |
| --- | --- |
| Название | Описание |
| TopWidthIncorrect | Неправильная ширина столешницы |
| TopDepthIncorrect | Неправильная глубина столешницы |
| TopHeightIncorrect | Неправильная высота столешницы |
| LegWidthIncorrect | Неправильная ширина ножки |
| TableHeightIncorrect | Неправильная высота стола |
| TopAndLegsAreaIncorrect | Неправильные значения зависимых параметров- размеры ножек и столешницы |

Изначальный макет пользовательского интерфейса представлен на рисунке 6.2.

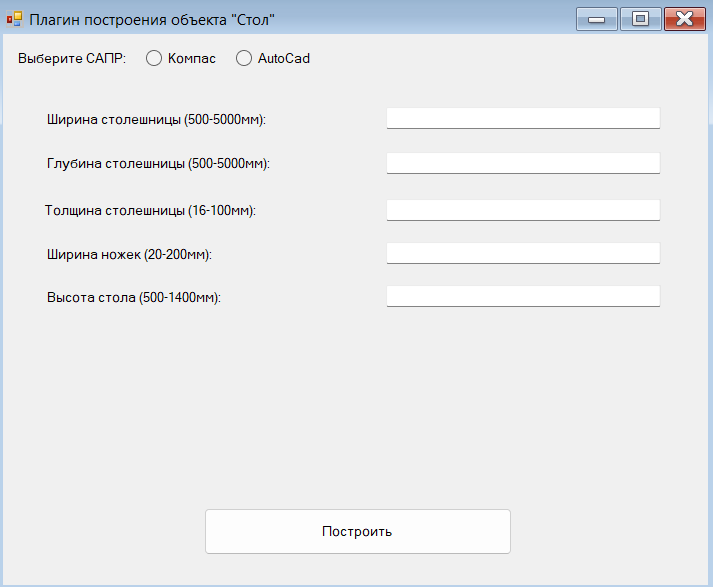


Рисунок 6.2 – Изначальный пакет пользовательского интерфейса

В итоговом проекте созданы следующие и методы, которые отображены на итоговой диаграмме классов (рисунок 6.3).

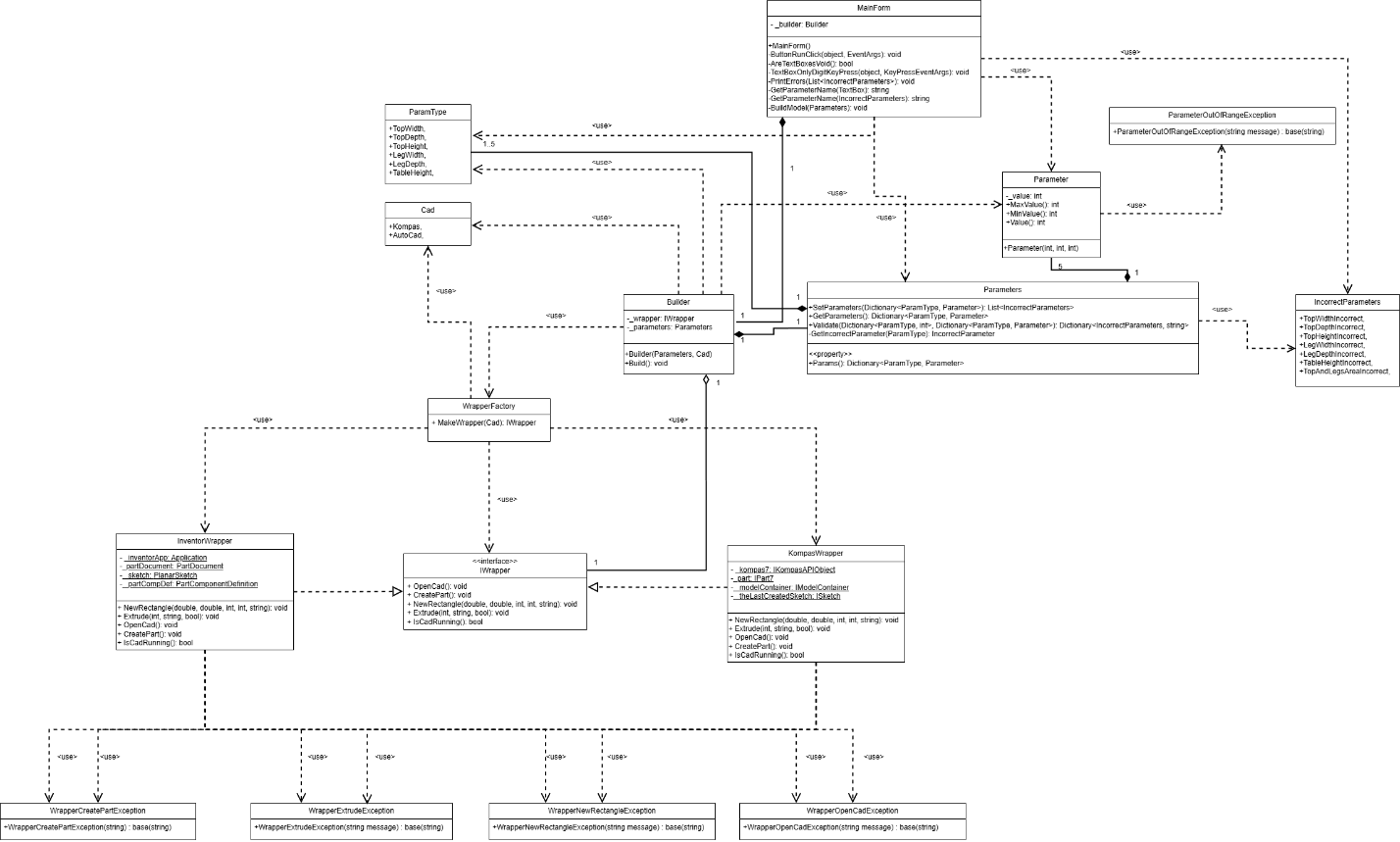


Рисунок 6.2 – Диаграмма классов в итоговом проекте

При реализации системы были сделаны следующие архитектурные изменения:

1. Задание крайних значений параметров для валидации перенесена из класса MainForm в класс Parameters, так как класс пользовательского интерфейса не может быть ответственным за элементы логики. Соответственно, удален класс Parameter.
2. Добавлены собственные исключения для классов Wrapper-ов.
3. Добавлено перечисление некорректных параметров для валидации в классе Parameters.
4. В интерфейс IWrapper и его реализации добавлен метод IsCadRunning(), для того, чтобы можно было не запускать для каждой задачи новый процесс САПР, а создавать новый проект в том процессе, который уже работает. Это также полезно, например, для модульных и нагрузочных тестов.

Класс MainForm представляет собой объект, через который будет осуществляться обработка пользовательского ввода и передача его в модель (таблица 6.10).

Таблица 6.10 – Свойства и методы класса MainForm

|  |  |
| --- | --- |
| Название | Описание |
| \_builder | Хранит экземпляр объекта для построения модели |
| BuildModel | Строит модель искомой детали |
| ButtonRunclick | Метод нажатия на кнопку “Построить” |
| TextBoxOnlyDigitKeyPress | Метод проверки вводимых в TextBox значений на принадлежность символов цифрам |
| PrintErrors | Метод печати ошибок в форму и обозначения неверных параметров красным цветом |
| GetParameterName | Метод конвертации TextBox-a в строку с названием соответствующего параметра |
| GetParameterName | Метод конвертации перечисления неверного параметра в строку с названием соответствующего параметра |

Класс Parameters представляет собой объект, хранящий параметры изделия и предоставляет функционал их валидации (таблица 6.11).

Таблица 6.11 – Свойства и методы класса Parameters

|  |  |
| --- | --- |
| Название | Описание |
| \_parameters | Хранит словарь параметров |
| SetParameters | Установить параметры. Внутри происходит вызов Validate |
| GetParameters | Получить словарь параметров |
| Validate | Проверяет значения из словаря |
| GetIncorrectParameter | Получить тип некорректного параметра, который относится к определенному типу параметра |

Перечисление ParameterType типов параметров (таблица 6.12).

Таблица 6.12 – Описание перечисления ParameterType

|  |  |
| --- | --- |
| Название | Описание |
| TopWidth | Ширина столешницы |
| TopDepth | Глубина столешницы |
| TopHeight | Высота столешницы |
| LegWidth | Ширина ножки |
| LegDepth | Глубина ножки |
| TableHeight | Высота стола |

Перечисление IncorrectParameters представляет из себя все возможные ошибки при валидации (таблица 6.13).

Таблица 6.13 – Перечисление IncorrectParameters неверных значений параметров

|  |  |
| --- | --- |
| Название | Описание |
| TopWidthIncorrect | Неправильная ширина столешницы |
| TopDepthIncorrect | Неправильная глубина столешницы |
| TopHeightIncorrect | Неправильная высота столешницы |
| LegWidthIncorrect | Неправильная ширина ножки |
| TableHeightIncorrect | Неправильная высота стола |
| TopAndLegsAreaIncorrect | Неправильные значения зависимых параметров- размеры ножек и столешницы |

Класс Builder использует методы для построения фигур, реализованные в классах ниже, для построения полноценной модели (таблица 6.14).

Таблица 6.14 – Свойства и методы класса Builder

|  |  |
| --- | --- |
| Название | Описание |
| \_wrapper | Хранит ссылку на базовый интерфейса построителя примитивов, в котором реализовано использование API САПР |
| Build | Построить модель |

Перечисление Cad определяет все возможные САПР, которые будут использоваться в плагине (таблица 6.15).

Таблица 6.15 – Описание перечисления Cad

|  |  |
| --- | --- |
| Название | Описание |
| Kompas | САПР Компас-3D |
| Inventor | САПР Autodesk Inventor |

Класс WrapperFactory представляет из себя фабрику объектов, реализующих интерфейс IWrapper (таблица 6.16).

Таблица 6.16 – Методы класса WrapperFactory

|  |  |
| --- | --- |
| Название | Описание |
| MakeWrapper | Принимает САПР, возвращает класс, реализующий Iwrapper для данной САПР. |

Классы KompasWrapper и InventorWrapper реализуют интерфейс IWrapper для определенной САПР (таблица 6.17).

Таблица 6.17 – Методы классов KompasWrapper и InventorWrapper

|  |  |
| --- | --- |
| Название | Описание |
| OpenCad | Открыть САПР |
| CreateSketch | Создать Эскиз |
| CreateRectangle | Начертить прямоугольник заданных размеров |
| Extrude | Выдавить эскиз |
| IsCadRunning | Узнать, запущена ли сейчас САПР |

Классы WrapperCreatePartException, WrapperExtrudeException, WrapperNewRectangleException и WrapperOpenCadException унаследованы от класса Exception для возможности передавать из реализаций IWrapper исключения для дальнейшей их обработки (таблица 6.18-6.21).

Таблица 6.18 – Методы класса WrapperCreatePartException

|  |  |
| --- | --- |
| Название | Описание |
| WrapperCreatePartException | Использует конструктор родительского класса, принимает сообщение типа string. Доступ к сообщению можно получить, например, после того как исключение попало в блок catch |

Таблица 6.19 – Методы класса WrapperExtrudeException

|  |  |
| --- | --- |
| Название | Описание |
| WrapperExtrudeException | Использует конструктор родительского класса, принимает сообщение типа string. Доступ к сообщению можно получить, например, после того как исключение попало в блок catch |

Таблица 6.20 – Методы класса WrapperNewRectangleException

|  |  |
| --- | --- |
| Название | Описание |
| + WrapperNewRectangleException | Использует конструктор родительского класса, принимает сообщение типа string. Доступ к сообщению можно получить, например, после того как исключение попало в блок catch |

Таблица 6.21 – Методы класса WrapperOpenCadException

|  |  |
| --- | --- |
| Название | Описание |
| WrapperOpenCadException | Использует конструктор родительского класса, принимает сообщение типа string. |

Класс Parameter хранит в себе целочисленное значение и минимальное и максимальное значение для данного параметра, которые задаются извне (таблица 6.22).

Таблица 6.22 – Свойства и методы класса Parameter

|  |  |
| --- | --- |
| Название | Описание |
| Parameter | Конструктор. Принимает целочисленное значение и сохраняет в свойство Value |
| Value | Хранит целочисленное значение параметра |
| MaxValue | Хранит максимальное значение параметра |
| MinValue | Хранит минимальное значение параметра |

# 7 ОПИСАНИЕ ПРОГРАММЫ ДЛЯ ПОЛЬЗОВАТЕЛЯ

Как и было указано в ТЗ, плагин запускается как отдельное приложение с помощью .exe файла. После загрузки открывается окно плагина с полями для ввода данных. Пользовательский интерфейс представлен на рисунке 7.1.

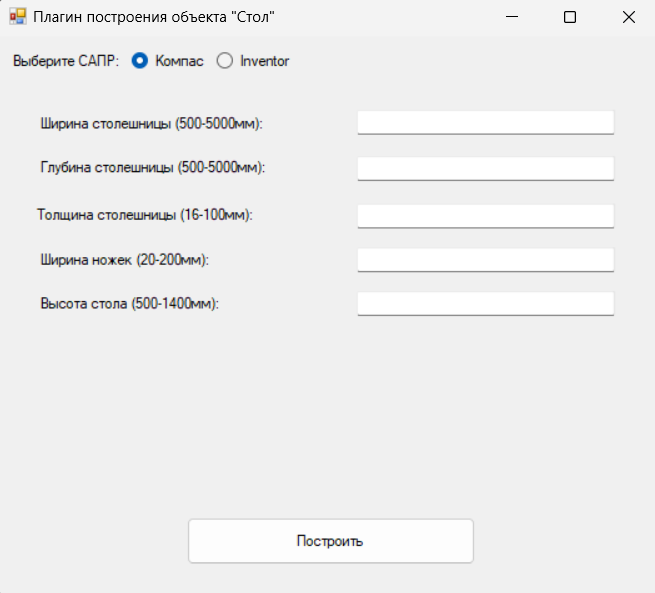


Рисунок 7.1 – Пользовательский интерфейс плагина

Пользовательский интерфейс плагина состоит из пяти параметров, доступных для ввода пользовательских данных, кнопки построения и строки состояния, которая сообщает о наличии ошибок пользовательского ввода, а так же кнопками для выбора САПР.

В плагине реализована проверка пользовательского ввода на минимально и максимально допустимые значения параметров, а также проверка зависимых параметров друг относительно друга.

При вводе некорректных пользовательских данных, соответствующие поля подсвечиваются красным цветом, а в строке состояния под кнопкой построения появляются сообщения об ошибках.

Если все данные были введены корректно, то при нажатии кнопки построения происходит построение модели в файле, из которого была загружена библиотека с плагином. Пользовательский интерфейс с неверно введёнными параметрами представлен на рисунке 7.2.

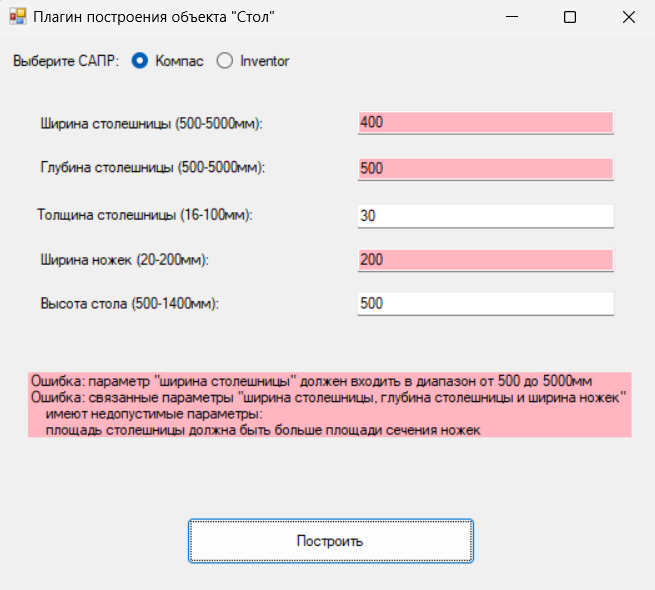


Рисунок 7.2 – Интерфейс с неверно введёнными параметрами

Корректное построение модели в САПР Компас 3D и Inventor представлено на рисунках 7.3-7.4 соответственно.



Рисунок 7.3 - Корректное построение модели в САПР Компас 3D

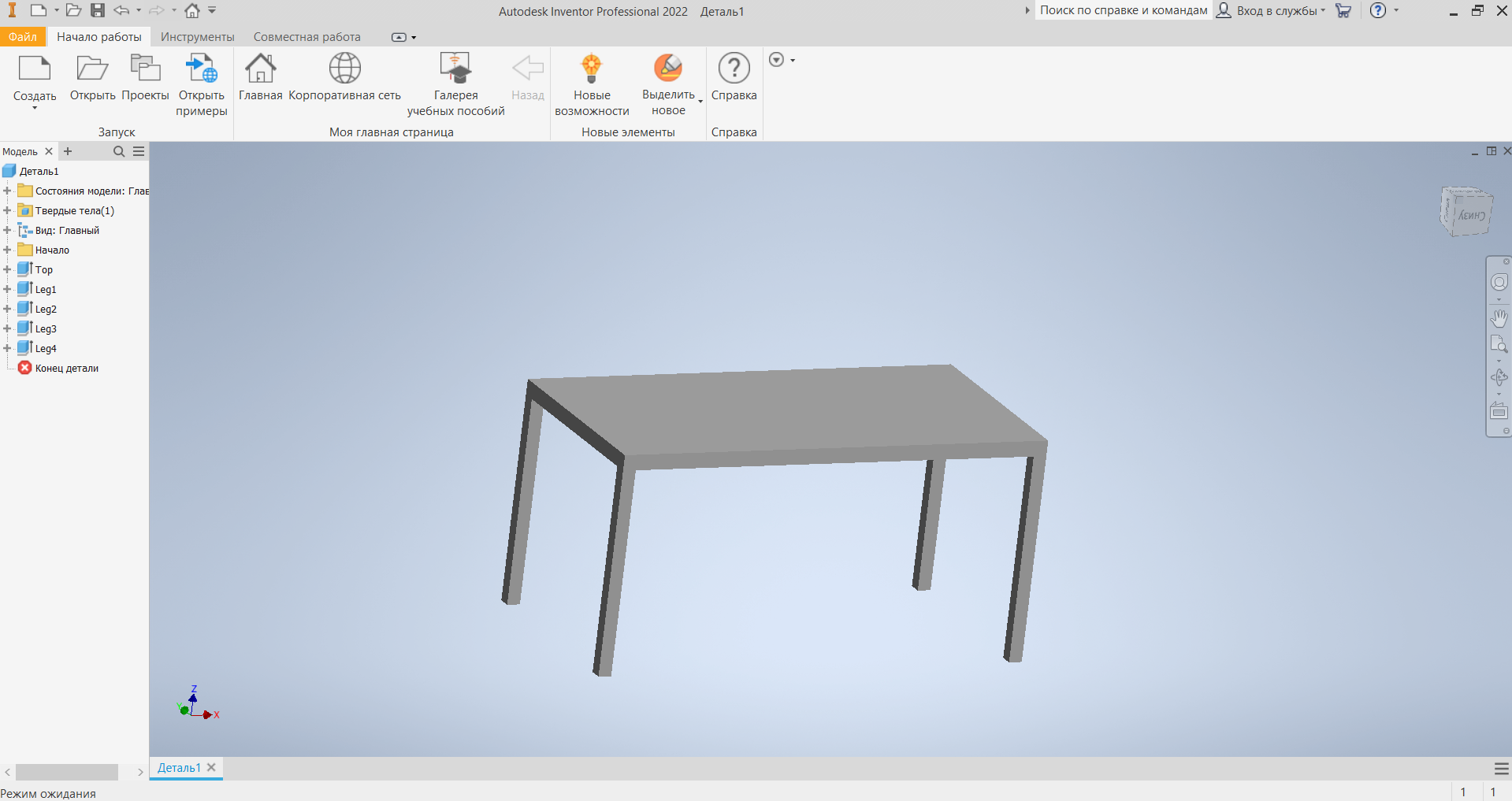


Рисунок 7.4 - Корректное построение модели в САПР Autodesk Inventor

# 8 ТЕСТИРОВАНИЕ ПЛАГИНА

Тестирование позволяет убедиться в работоспособности программы, выявить ошибки при изменении функциональности.

## 8.1 Функциональное тестирование

Вывод различных сообщений об ошибке показан на рисунках 8.1 – 8.3.

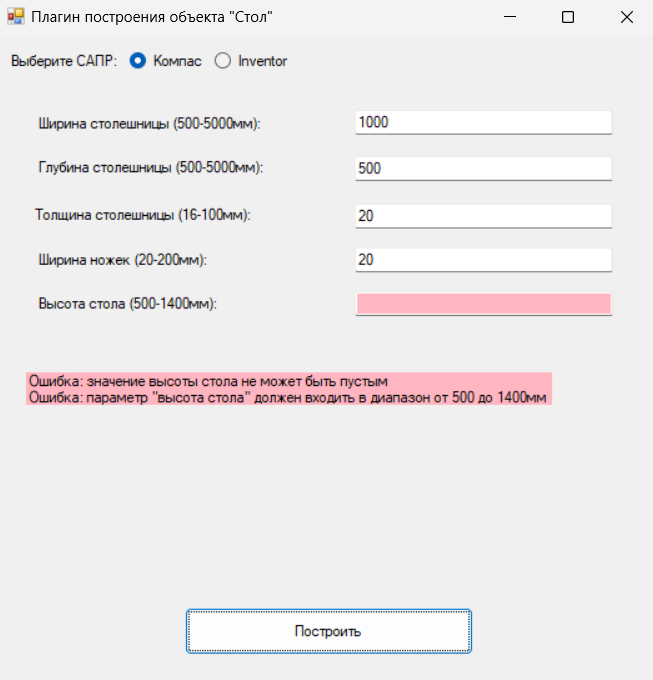


Рисунок 8.1 – Вывод сообщения при пустом текстовом поле

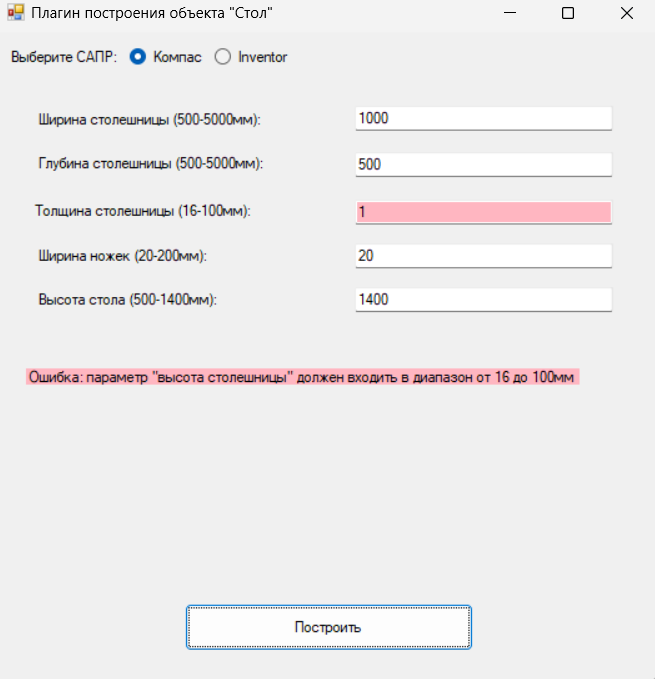


Рисунок 8.2 – Вывод сообщения при значениях, не входящих в допустимый диапазон

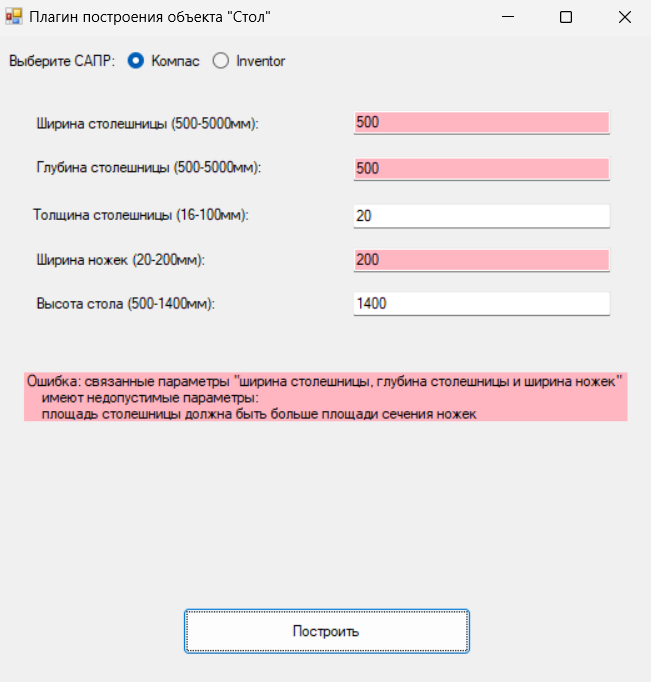


Рисунок 8.3 – Вывод сообщения при значениях, не подходящих условиям в зависимых параметрах

Минимальные значения:

1. Ширина столешницы = 500 мм;
2. Глубина столешницы = 500 мм;
3. Толщина столешницы = 16 мм;
4. Ширина ножек = 20 мм;
5. Высота стола = 500 мм.

Максимальные значения:

1. Ширина столешницы = 5000 мм;
2. Глубина столешницы = 5000 мм;
3. Толщина столешницы = 100 мм;
4. Ширина ножек = 200 мм;
5. Высота стола = 1400 мм.

На рисунках 8.4-8.5 представлены модели с минимальными и максимальными введёнными параметрами, построенные в Компас 3D.

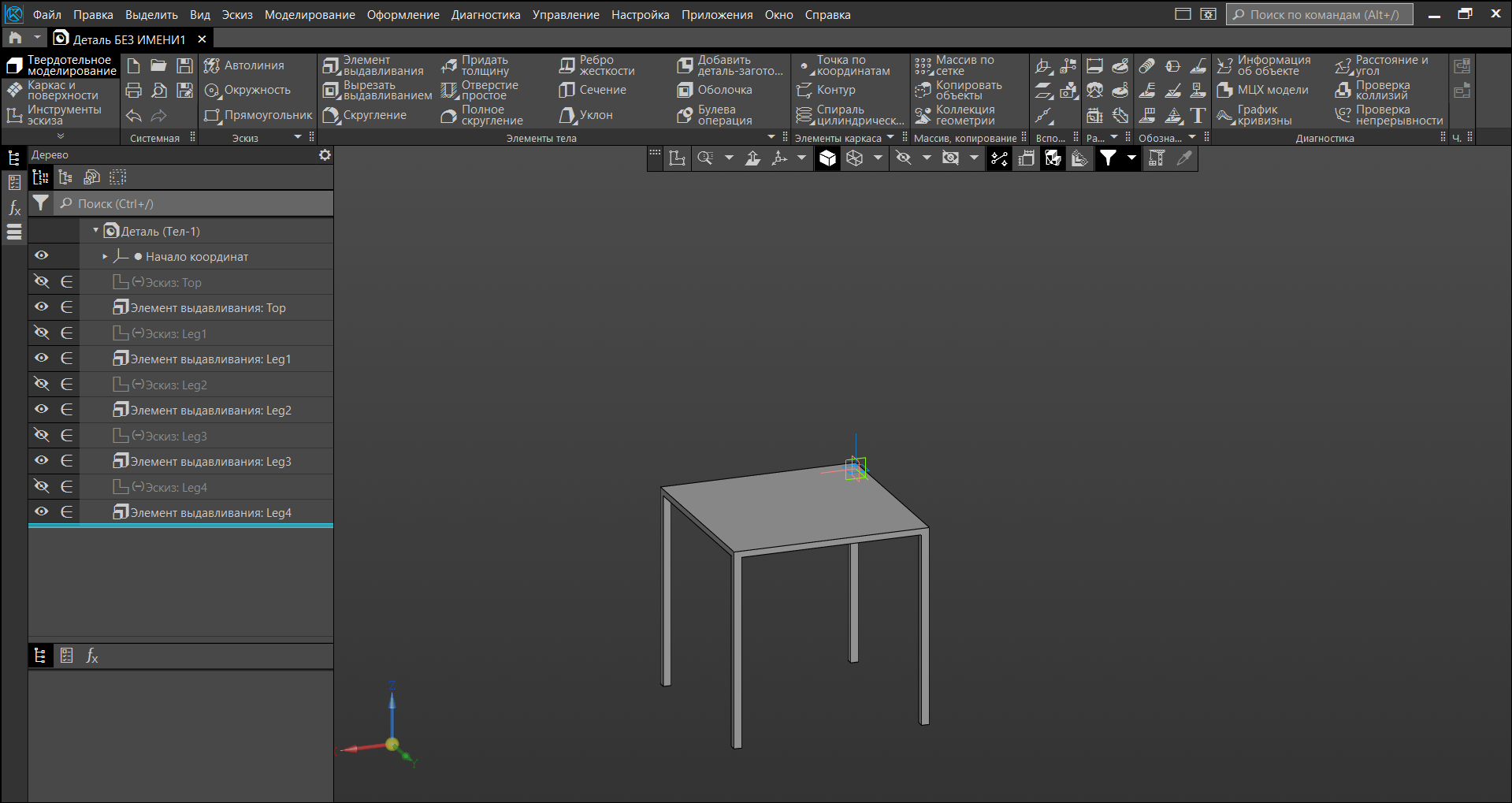


Рисунок 8.4 – Модель с минимальными параметрами в Компас 3D

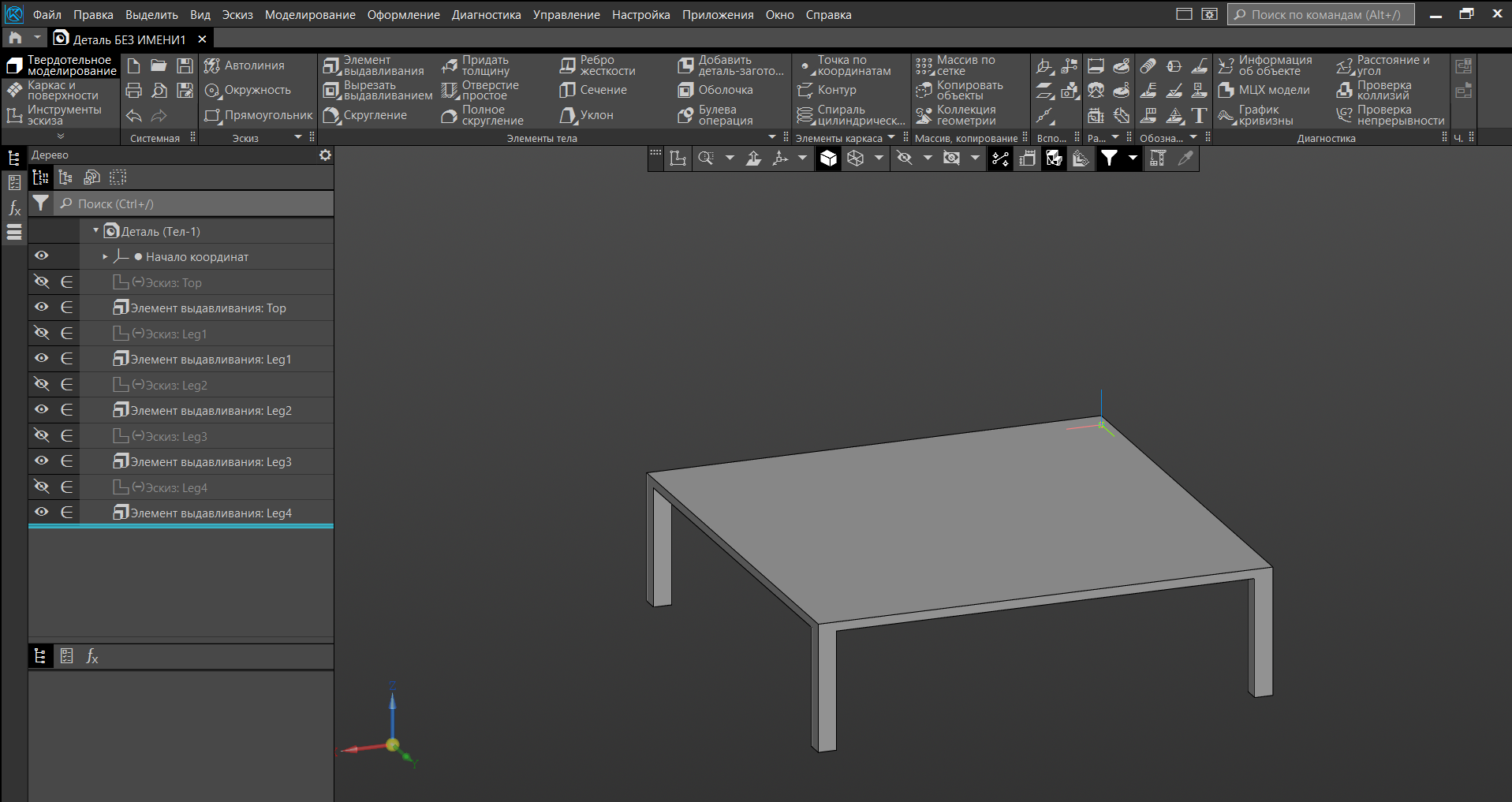


Рисунок 8.5 – Модель с максимальными параметрами в Компас 3D

На рисунках 8.6-8.7 представлены модели с минимальными и максимальными введёнными параметрами, построенные в Autodesk Inventor.

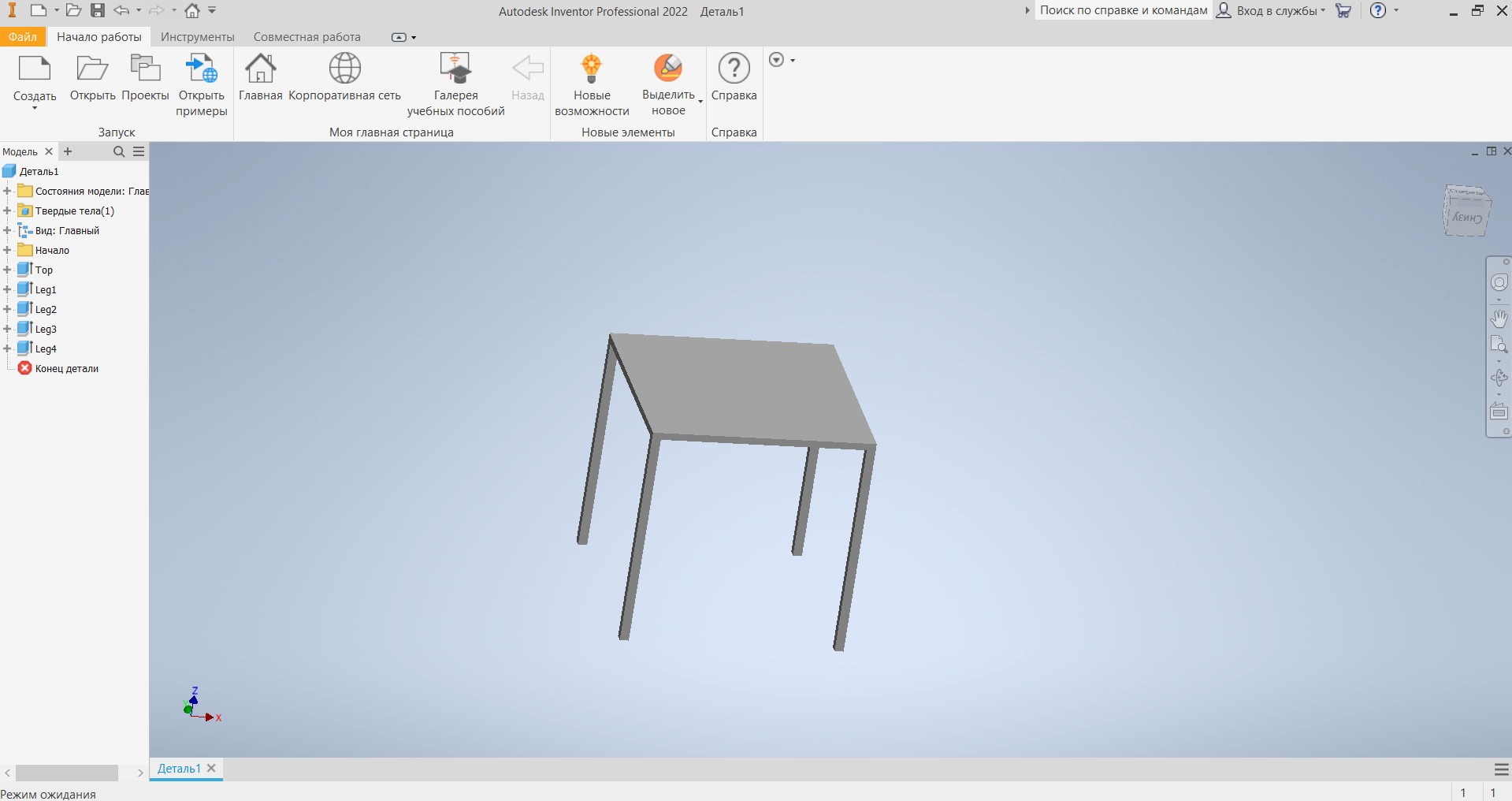


Рисунок 8.6 - Модель с минимальными параметрами в Autodesk Inventor

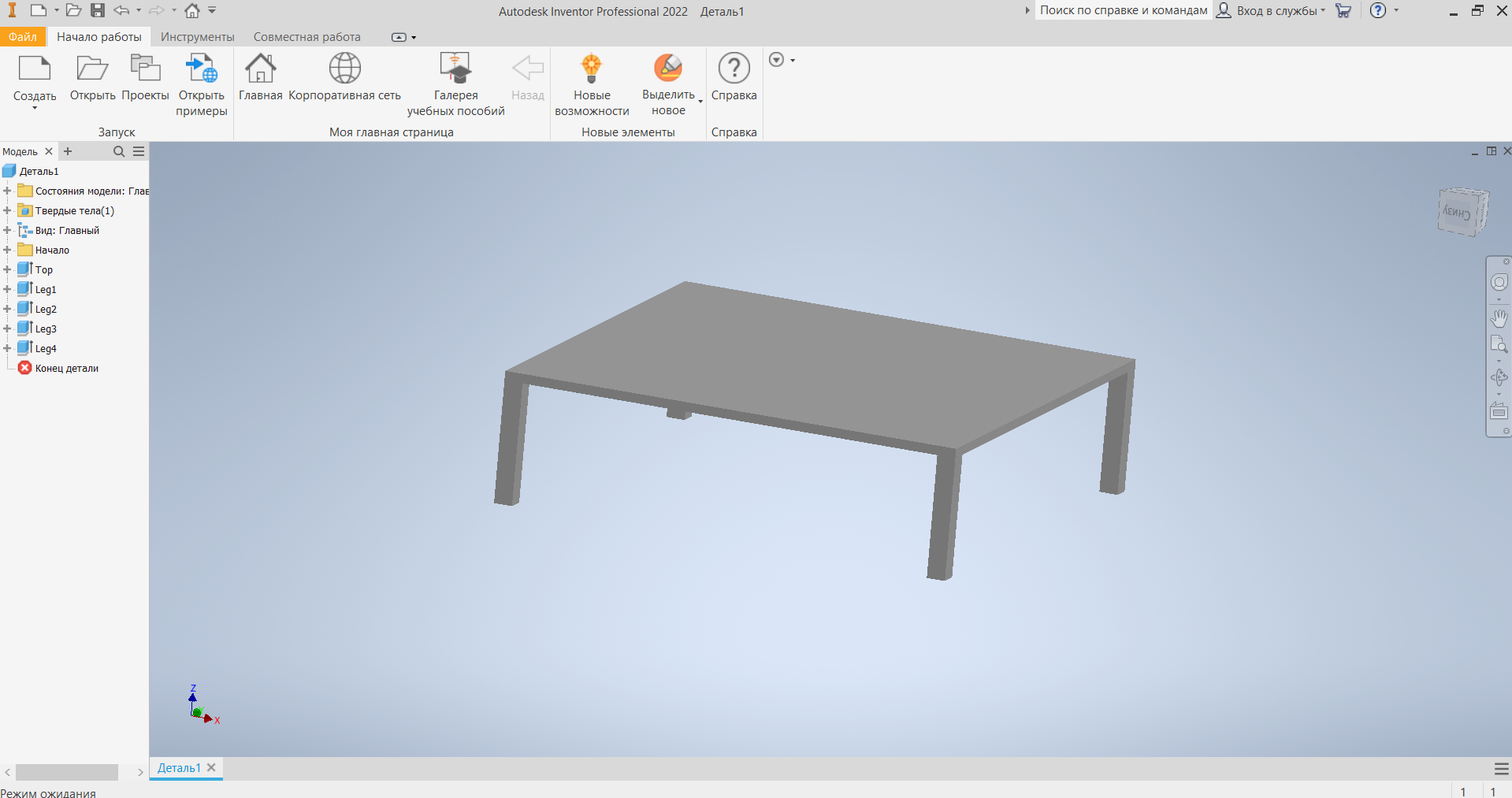


Рисунок 8.7 – Модель с максимальными параметрами в Autodesk Inventor

## 8.2 Модульное тестирование

В целях проверки корректности работы методов и свойств классов при помощи фреймворка для модульного тестирования NUnit версии 3.13 было проведено модульное тестирование открытых свойств и методов. Были протестированы следующие классы: Parameters, KompasWrapper и InventorWrapper. Покрытие модели тестами составило 100%, что показано на рисунке 8.8.

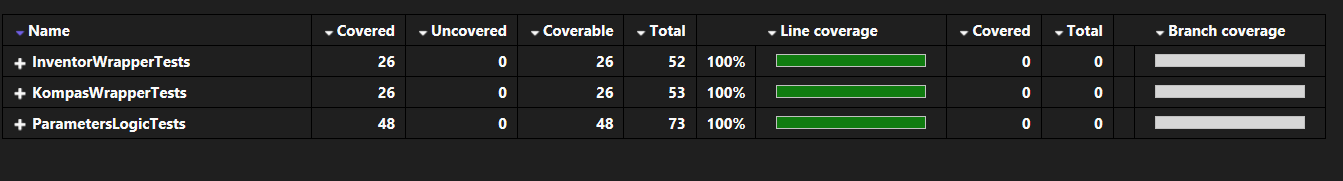


Рисунок 8.8 - Покрытие кода тестами

Тесты для класса Parameters представлены в таблице 8.1.

Таблица 8.1 – Описание тестов для класса Parameters

|  |  |
| --- | --- |
| Название | Описание |
| ParametersRightTest | Тест валидации заведомо правильных параметров |
| ParametersIndependentWrongTest | Тест валидации заведомо неправильных параметров |
| ParametersDependentWrongTest | Тест валидации зависимых параметров |

Тесты для классов KompasWrapper и InventorWrapper представлены в таблице 8.2.

Таблица 8.2 – Описание тестов для классов KompasWrapper и InventorWrapper

|  |  |
| --- | --- |
| Название | Описание |
| CreatePartTest | Тест создания детали в САПР |
| NewRectangleTest | Тест создания прямоугольника в САПР |
| ExtrudeTest | Тест операции выдавливания в САПР |

## 8.3 Нагрузочное тестирование

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

* Процессор AMD Ryzen 5 3500U 2.10 GHz;
* 7 ГБ доступной ОЗУ;
* Операционная система Windows 11 домашняя x64;
* Видеокарта Radeon Vega Mobile Gfx с объёмом памяти 1 ГБ.

Для нагрузочного тестирования был задан цикл построения детали. Для измерения времени был использован класс Stopwatch. Тестирование заключалось в построении хотя бы 100 столов со средними параметрами. На рисунках 8.9 – 8.12 показан результат данного тестирования.

Рисунок 8.9 – График занятой ОЗУ (ось ординат) от количества построенных моделей (ось абсцисс) в САПР Компас 3D

Рисунок 8.10 – График времени построения (ось ординат) от количества построенных моделей (ось абсцисс) в САПР Компас 3D

Рисунок 8.11 – График занятой ОЗУ (ось ординат) от количества построенных моделей (ось абсцисс) в САПР Inventor

Рисунок 8.12 – График времени построения (ось ординат) от количества построенных моделей (ось абсцисс) в САПР Inventor

Компас 3D не справился с задачей – из-за него компьютер полностью завис на построении 75 детали. Autodesk Inventor справился лучше, с помощью него удалось построить 131 деталь до полного зависания машины.

Исходя из графиков, представленных на рисунках 8.9 и 8.11, можно заметить линейное увеличение объёма занимаемой оперативной памяти в зависимости от количества моделей до тех пор, пока оперативная память не закончится. Однако, после заполнения её объёма, работа плагина не останавливается, потому что начинает использоваться файл подкачки, восполняющий необходимость в оперативной памяти за счёт использования постоянной памяти с SSD накопителя.

# Заключение

В процессе разработки приложения был создан плагин, позволяющий создавать 3D-модели шестерней в САПР Компас 3D и Autodesk Inventor.

Для разработки были изучены:

* Компас API v7 для взаимодействия с САПР из программного кода;
* Autodesk Inventor API для взаимодействия с САПР из программного кода;
* Библиотека NUnit для написания Unit-тестов.

# Список используемых источников

1. Официальный сайт NUnit. [Электронный ресурс]. Режим доступа: свободный (дата обращения 19.12.24), https://nunit.org/
2. Официальный сайт SketchUp. [Электронный ресурс]. Режим доступа: свободный (дата обращения 20.10.24), <https://www.sketchup.com/en>
3. Официальный сайт плагина CraftReprts. [Электронный ресурс]. Режим доступа: свободный (дата обращения 20.10.24), https://craftreports.org/
4. Официальный сайт Pro100. [Электронный ресурс]. Режим доступа: свободный (дата обращения 20.10.24), https://www.ecru.pl/ru
5. Официальный сайт Объемник. [Электронный ресурс]. Режим доступа: свободный (дата обращения 20.10.24), https://mebelsoft.org/
6. Толковый словарь Ожегова. Стол. [Электронный ресурс]. Режим доступа: свободный (дата обращения 20.10.24), https://gufo.me/dict/ozhegov/%D1%81%D1%82%D0%BE%D0%BB
7. Официальный сайт САПР Kompas-3D. [Электронный ресурс]. Режим доступа: свободный (дата обращения 20.10.24), https://kompas.ru/kompas-3d/about/
8. Введение в UML от создателей языка [Текст] : руководство пользователя / Г. Буч, Д. Рамбо, И. Якобсон. - 2-е изд. - М. : ДМК Пресс, 2012. - 494 с. : ил. - (Классика программирования). - Предм. указ.: с. 483- 493. - ISBN 978-5-94074-644-7;
9. ОС ТУСУР 01-2021 «Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления от 25.11.2021»;
10. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку. Изд: Вильямс, 2013, с.739 (3-е издание)
11. Официальный сайт САПР Kompas-3D. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), <https://kompas.ru/kompas-3d/about/>
12. Официальный сайт САПР Autodesk Inventor. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), https://www.autodesk.com/products/inventor/overview?term=1-YEAR&tab=subscription