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)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Студент гр. 586-2

\_\_\_\_\_\_\_\_\_\_\_\_\_Аханов А.А.

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020г.

Руководитель

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_А.А. Калентьев

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020г.

Томск 2020

**Оглавление**

[1 Введение 3](#_Toc40590102)

[2 Постановка и анализ задачи 4](#_Toc40590103)

[2.1 Описание предмета проектирования 4](#_Toc40590104)

[2.2 Выбор инструментов и средств реализации 5](#_Toc40590105)

[2.3 Назначение плагина 6](#_Toc40590106)

[3 Обзор аналогов 7](#_Toc40590107)

[3.1 AUTOCAD 7](#_Toc40590108)

[3.2 Комплекс программ (ЭЛЕКРАН СОФТ, г. Одесса) 8](#_Toc40590109)

[3.3 bCAD-Мебельщик (ПроПро Группа, г. Новосибирск) 10](#_Toc40590110)

[3.4 WOODY (НФ ИНТЕАР Лтд., г. Киев) 11](#_Toc40590111)

[3.5 К3-МЕБЕЛЬ (ГЕОС, г. Нижний Новгород) 12](#_Toc40590112)

[4 Описание реализации 15](#_Toc40590113)

[4.1 Диаграмма вариантов использования (Use Cases) 15](#_Toc40590114)

[4.2 Диаграмма классов 16](#_Toc40590115)

[5 Описание программы для пользователя 20](#_Toc40590116)

[6 Тестирование программы 23](#_Toc40590117)

[6.1 Функциональное тестирование 23](#_Toc40590118)

[6.2 Модульное тестирование 24](#_Toc40590119)

[6.3 Нагрузочное тестирование 25](#_Toc40590120)

[Заключение 28](#_Toc40590121)

[Список использованных источников 29](#_Toc40590122)

[Приложение А 31](#_Toc40590123)

# 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 «Табурет» для системы автоматизированного проектирования SOLIDWORKS 2020 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 SOLIDWORKS 2020, строит модель «Табурет». [3] Необходимо чтобы плагин позволял задавать параметры по умолчанию, а также изменять входные параметры табурета, такие как:

* длина ножки;
* высота ножки;
* ширина ножки;
* длина сиденья;
* ширина сиденья;
* толщина сиденья;
* ширина связи;
* высота связи;
* длина связи.

# 2.1 Описание предмета проектирования

Предметом проектирования является табурет. Табурет – мебельное изделие для сиденья одного человека без спинки и подлокотников. Табуреты изготавливаются из дерева, металла, пластмасс и других материалов. Сиденье табурета бывает, как жёстким, так и с мягким элементом с обивкой из ткани и кожи. Параметры и ограничения следующие:

1. Leg height – высота ножки, не может быть меньше 500мм и больше 1000мм.
2. Leg width – ширина ножки, должна быть равна ее длине.
3. Leg length – длина ножки, не может быть меньше 40мм и больше 100мм.
4. Seat width – ширина сиденья, не может быть больше длины сиденья.
5. Seat length – длина сиденья, не может быть меньше 320мм и больше 500мм.
6. Seat thickness – толщина сиденья, не может быть меньше 40мм и больше 100мм.
7. Bond height – высота связи, не может быть меньше 30мм и больше 50мм.
8. Bond width – ширина связи, не может быть меньше 20мм и больше ширины ножки.
9. Bond length – длина связи, полностью зависит от расстояния между ножками табурета и не может быть длиннее сиденья. [4]

Пример проектируемого изделия приведен ниже, на рисунке 2.1.

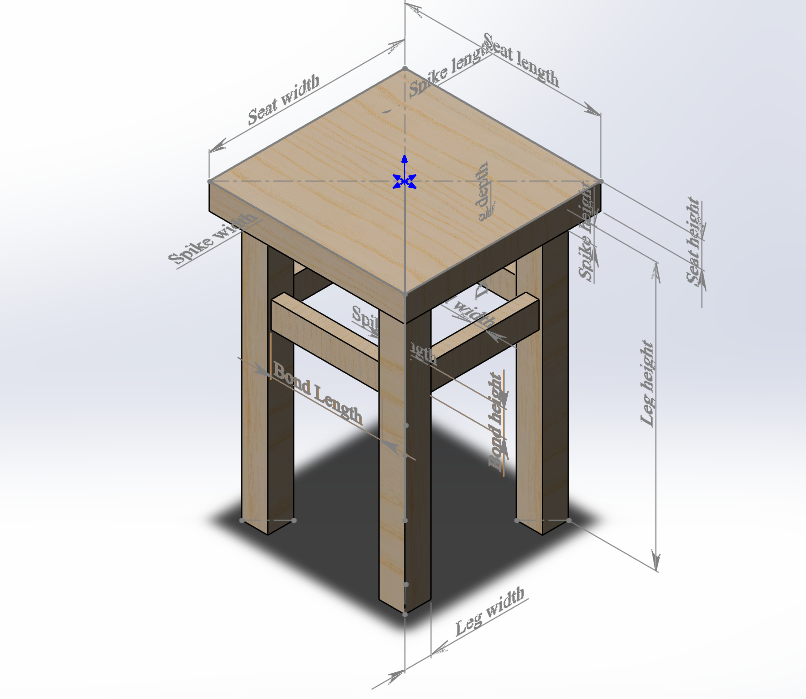


Рисунок 2.1 – Предмет изделия

# 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 4.8 [2], библиотеки «SolidWorks.Interop. sldworks» [5] версии 27.3 для основных операций в системе SOLIDWORKS 2020.

Инструментом тестирования и создания модульных тестов был выбран тестовый фреймворк NUnit [6] версии 3.12.

Для реализации пользовательского интерфейса использовалась система для построения настольные и мобильные приложения Windows – Windows Forms (WinForms) [7].

# 2.3 Назначение плагина

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

# 3 Обзор аналогов

Исторически первые системы автоматизированного проектирования разрабатывались для машиностроения. Первой САПР, получившей широкое распространение на отечественных предприятиях, стала система AutoCAD. В дальнейшем на рынке появилось несколько специализированных систем для мебельной промышленности, большинство из которых представляют собой адаптацию машиностроительных систем для специфики отрасли.

В настоящее время отечественными мебельными предприятиями для автоматизации проектирования и технологической подготовки производства мебели чаще всего используются следующие системы:

* AutoCAD ([Autodesk](http://www.cadcatalog.ru/company/autodesk.html), США);
* Комплекс программ 3D-Constructor, 32D-Flat, 2D-Place ([ЭЛЕКРАН СОФТ](http://www.cadcatalog.ru/company/r_elecran.html), г. Одесса);
* bCAD-Мебельщик ([ПроПро Группа](http://www.cadcatalog.ru/company/r_propro.html" \t "_self), г. Новосибирск);
* WOODY ([НФ ИНТЕАР Лтд.](http://www.cadcatalog.ru/company/r_intear.html), г. Киев);
* БАЗИС ([БАЗИС-ЦЕНТР](http://www.cadcatalog.ru/company/r_bazis.html), г. Коломна);
* К3-Мебель ([ГеоС](http://www.cadcatalog.ru/company/r_geos.html" \t "_self), г. Нижний Новгород);
* AutoCAD (Autodesk, США).

Несмотря на то, что для автоматизации мебельных предприятий разработано несколько специализированных САПР, в той или иной степени учитывающих особенности проектирования и производства мебели, для решения этих задач нередко используется система AutoCAD в «чистом» виде. Это объяснятся ее широкими функциональными возможностями и распространенностью на предприятиях различного профиля. [8]

# 3.1 AUTOCAD

Система AutoCAD – одна из самых мощных систем автоматизированного проектирования в классе систем, работающих на персональных компьютерах. Она может выполнять практически все виды чертежно-конструкторских работ, необходимых в самых разнообразных областях технического проектирования, в том числе и для проектирования мебели. [9]

Основные возможности системы AutoCAD следующие:

* построение и редактирования двухмерных (плоских) чертежей и получение твердых копий графических документов;
* развитые технологии работы со структурами данных (блоки, внешние ссылки и т.п.), что позволяет организовать совместное использование информации с другими приложениями, а также применение информации из внешних баз данных;
* трехмерное (пространственное) моделирование и создание реалистического изображения пространственных объектов с удалением невидимых линий, наложением теней и текстур, различными вариантами освещения;
* эффективная работа с архивами чертежей: стандартизация, сопровождение архивов, работа в сети, защита от сбоев;
* индивидуальная настройка рабочей среды в соответствии с индивидуальными потребностями и наклонностями конкретного пользователя и спецификой предметной области;
* наличие встроенного языка программирования AutoLISP (и его развития – Visual LISP), который позволяет создавать прикладные проблемно-ориентированные системы проектирования для различных сфер применения. [9]

# 3.2 Комплекс программ (ЭЛЕКРАН СОФТ, г. Одесса)

Системой автоматизации мебельных предприятий, использующей в качестве базового графического ядра AutoCAD, является комплекс программ 3D-Constructor, 3D-Flat, 2D-Place. Он состоит из трех независимых компонентов, которые выполняют следующие функции:

* 3D-Constructor – программа для конструирования мебели и выпуска конструкторско-технологической документации;
* 3D-Flat – программа для приема заказов в мебельном салоне;
* 2D-Place – программа оптимального раскроя листового материала.

Основным отличием данного комплекса является полная параметризация модели мебели, что является очень удобным при наличии на предприятии ряда типоразмеров мебели. При конструировании нестандартных изделий используются графические возможности системы AutoCAD.

База крепежной фурнитуры настраивается пользователем на основе типовых шаблонов, поставляемых с программой. Расстановка ее обеспечивается при помощи создания так называемых стилей крепления (например, эксцентриковая стяжка плюс шкант). При облицовке кромкой допускается указание любого торца детали, которая может иметь произвольную геометрическую форму. Важной особенностью системы является возможность установки на изделие типовых сборок, например, выдвижных ящиков. После создания модели изделия процесс получения чертежей, спецификаций и ведомостей материалов происходит автоматически, причем формы выходных документов могут настраиваться пользователем. Виды (стили) размеров, проставляемых на чертежах, настраиваются средствами AutoCAD. Дополнительно система позволяет компоновать чертежи, задавать масштаб, проставлять позиции деталей, выполнять разноску сборки. [10]

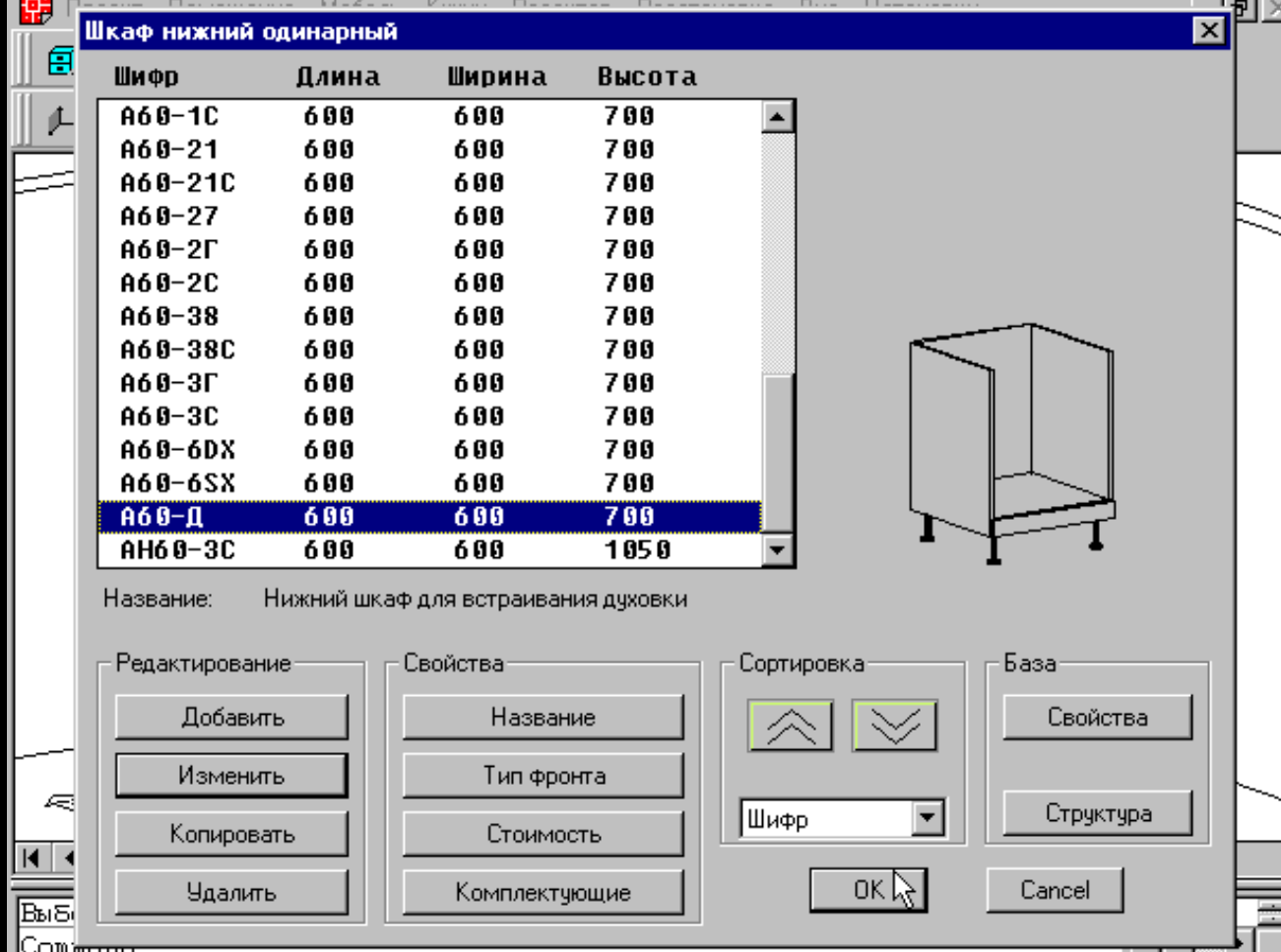


Рисунок 3.2 – Интерфейс программы от Элекран Софт

# 3.3 bCAD-Мебельщик (ПроПро Группа, г. Новосибирск)

Система bCAD-Мебельщик предназначена для автоматизации проектирования корпусной мебели. Основная ее идея заключается в том, что пользователь работает в основном с объемной моделью изделия, собирая ее из отдельных панелей и крупных готовых блоков, или модифицирует уже готовые изделия в соответствии с новыми требованиями. С технической точки зрения bCAD-Мебельщик представляет собой универсальный графический редактор bCAD, дополненный специальными командами для учета особенностей проектирования мебели. [11]

При работе с системой bCAD-Мебельщик предполагается, что мебельные изделия состоят из следующих элементов:

* плоские листовые панели – основные элементы мебели, для которых требуется выполнить конструкторскую и технологическую документацию (перегородки, столешницы, полки и т.д.);
* крепеж – элементы мебели, которые обеспечивают соединение панелей и закрепления фурнитуры на них;
* фурнитура – остальные элементы мебели, не попадающие под выше перечисленные категории.

Модель изделия создается при помощи размещения листовых панелей так, как они должны находиться в проектируемом изделии. Из панелей получается трехмерная модель, на которой и отрабатывается компоновка будущего изделия. На втором этапе конструирования расставляется крепеж, затем фурнитура.

По готовой модели формируются отчет, смета, чертежи и карты раскроя, информация для которых берется непосредственно из модели.

Отличительной особенностью системы bCAD является наличие инструментов для создания моделей типовых комнат и расстановки в них мебели. Это позволяет использовать систему при приеме заказов в мебельных салонах. [11]

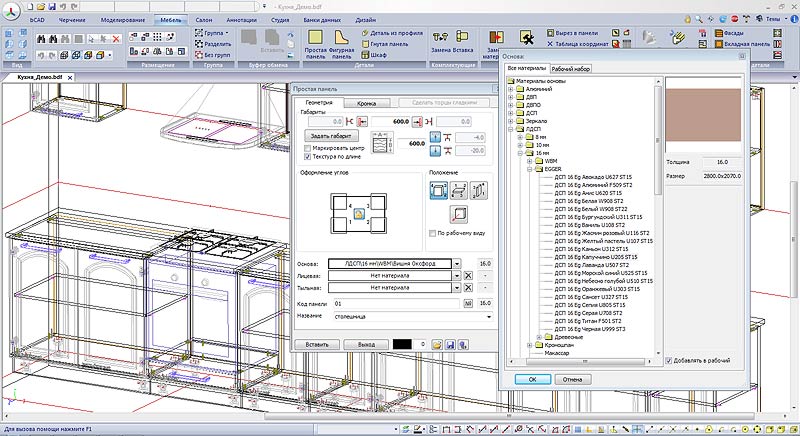


Рисунок 3.3 – Интерфейс программы bCAD-Мебельщик

# 3.4 WOODY (НФ ИНТЕАР Лтд., г. Киев)

Система WOODY состоит из двух модулей: WOODY – для проектирования корпусной мебели и шкафов-купе, и SAWYE – для подготовки карт раскроя и ведения учета материалов. Конструкторская база данных системы WOODY содержит необходимые материалы для проектирования мебели: плитные материалы, облицовочную ленту и фигурные профили, фурнитуру различного функционального назначения, мебельные фасады и т.п. Она снабжена инструментарием, позволяющим дополнять и редактировать содержимое базы. [12]

Для создания объектов проектирования в системе WOODY используются специальные мастера, предназначенные для выполнения всех необходимых операций. Например, мастер деталей создает детали корпуса мебельного изделия, а мастер чертежей – генерирует чертежную документацию. Проектирование изделия ведется как на ортогональных проекциях, так и на аксонометрических и перспективных изображениях. В процессе этого многие операции выполняются автоматически, например, размещение фурнитуры, формирование отверстий, проверка корректности сборки и другие.

Система WOODY позволяет придавать деталям сложную геометрическую форму при помощи редактора контуров в мастере чертежей. Помимо этого, она может работать одновременно с несколькими изделиями в пространстве одного документа, размещать и поворачивать их друг относительно друга, формировать иерархию проекта.

После создания модели изделия или проекта в системе WOODY автоматически формируются чертежи, спецификации расхода материалов, смета проекта, цветные иллюстрации, инструкция по сборке изделия, и подготавливается информация для системы раскроя.

Для стыковки системы WOODY с другим программным обеспечением в ней реализованы возможности импорта и экспорта информации во многие популярных форматы хранения данных. [12]

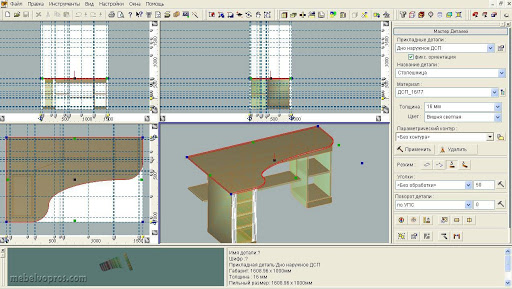


Рисунок 3.4 – Интерфейс программы Woody

# 3.5 К3-МЕБЕЛЬ (ГЕОС, г. Нижний Новгород)

Комплекс программ К3-Мебель представляет собой набор программ, созданных на базе геометрического ядра К-3, которые обеспечивают автоматизацию всех работ по проектированию, подготовке производства и продаже мебели. Каждая программа выполняет свою функцию и передает необходимые данные другим программам комплекса. Это позволяет унифицировать и автоматизировать все работы в рамках конкретного предприятия и его торговой сети.

Составными частями комплекса являются следующие программы:

* К3-Торговля мебелью – для оформления заказов на стандартную мебель по заранее подготовленным электронным каталогам;
* К3-Салон мебели – программа, объединяющая средства подготовки электронных каталогов на стандартную мебель и средства приема заказов на нестандартную мебель;
* К3-Мебель ПКМ (Проектирование Корпусной Мебели) – объединение возможностей программы К3-Салон мебели со средствами проектирования.

Программа К3-Мебель ПКМ позволяет проектировать изделия мебели, используя стандартизованные заготовки в виде прототипов изделий мебели. Ее возможности:

* создание пространственных моделей мебели;
* расстановка крепежа и расчет присадочных отверстий;
* работа со справочником материалов и комплектующих;
* выпуск конструкторской документации;
* подготовка полноцветных изображений мебели;
* расчет комплектации и себестоимости мебели;
* подготовка компьютерных каталогов изделий;
* создание библиотек параметрических моделей мебели.

Для автоматизации выполнения ряда операций в составе комплекса К3-Мебель созданы специальный Конструкторский модуль и программа "Раскрой листовых материалов". При подготовке производства стандартной и нестандартной мебели с целью обеспечения единства работы с информацией в системе имеются средства для обработки заказов и подготовке производственных заданий. Выполнение анализа информации по заказам обеспечивает программа График заказов, которая позволяет осуществлять контроль прохождения заказа от момента приема до момента выполнения и передачи в архив. [13]

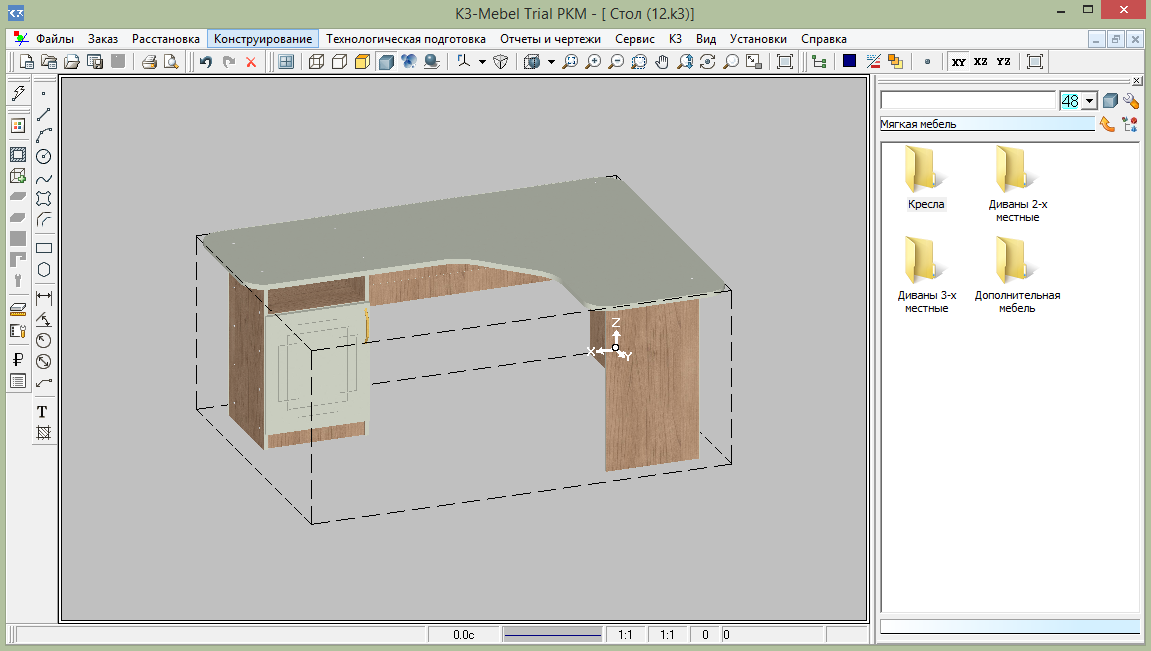


Рисунок 3.5 – Интерфейс программы К3-Мебель

# 4 Описание реализации

Для графического описания абстрактной модели проекта, а также пользовательского взаимодействия (сценария действий) использован стандарт UML. Построение диаграмм выполнялось в программном продукте Enterprise Architect 12.1.

# 4.1 Диаграмма вариантов использования (Use Cases)

Диаграмма вариантов использования в UML — диаграмма, отражающая отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне. Основное назначение диаграммы — описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему. [14]

На рисунке 4.1 представлена изначальная диаграмма вариантов использования для плагина.

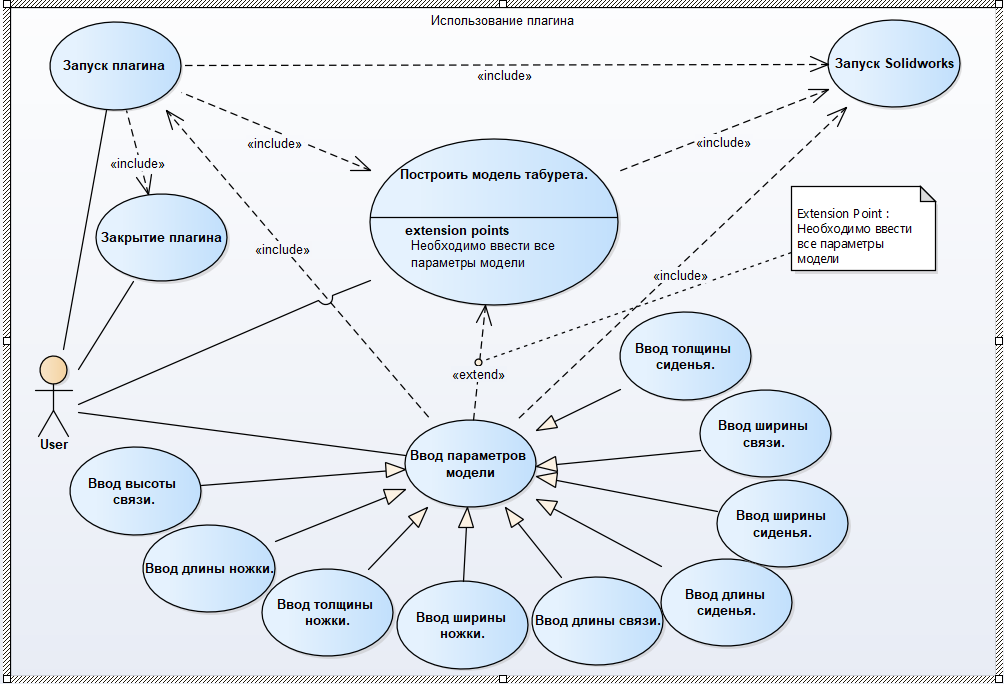


Рисунок 4.1– Изначальная диаграмма вариантов использования

На рисунке 4.2 представлена финальная версия диаграммы вариантов использования, в которой появилась возможность выбора модели табурета, в отличии от изначальной диаграммы вариантов использования.

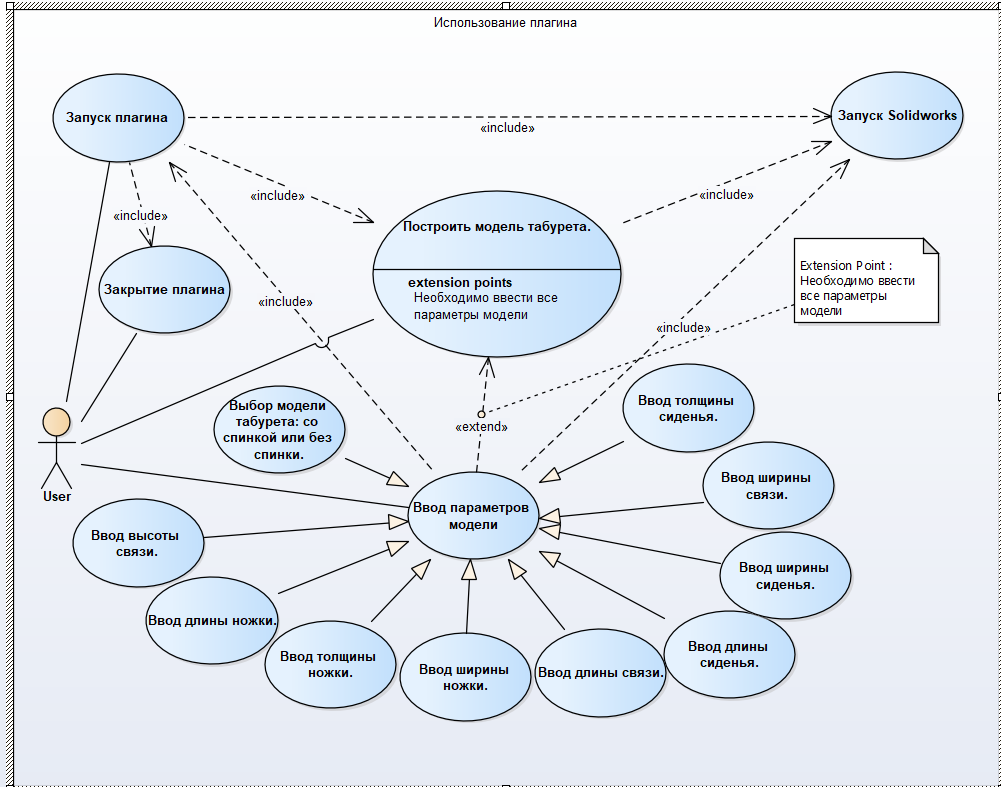


Рисунок 4.2 – Финальная диаграмма вариантов использования

# 4.2 Диаграмма классов

На рисунке 4.2 представлена диаграмма классов, описывающая архитектуру плагина.

UML – это 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. [15]

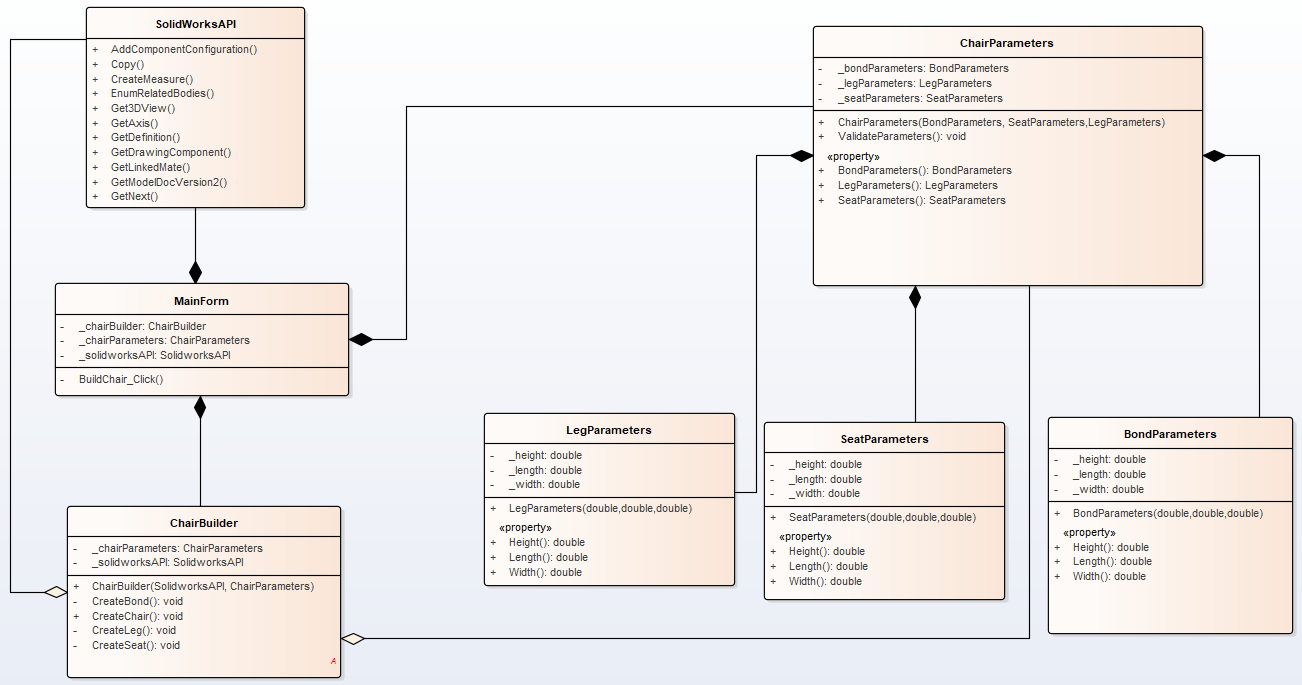


Рисунок 4.2 – Изначальная диаграмма классов

Для реализации подсистемы были спроектированы следующие классы:

* MainForm – класс диалогового окна обеспечивающий взаимодействие между пользователем и программой;
* ChairParameters – класс, хранящий в себе все параметры модели, а также обеспечивающий проверку введенных параметров;
* LegParameters – класс, хранящий в себе параметры ножки табурета;
* SeatParameters – класс, хранящий в себе параметры сиденья табурета;
* BondParameters – класс, хранящий в себе параметры связей табурета;
* ChairBuilder – класс, отвечающий за вызов методов Solidworks API необходимых для постройки объекта проектирования;
* SolidworksAPI – класс, отвечающий за работу с API Solidworks.

В итоговом проекте созданы следующие классы и методы, которые отображены на итоговой диаграмме классов (рисунок 4.3).

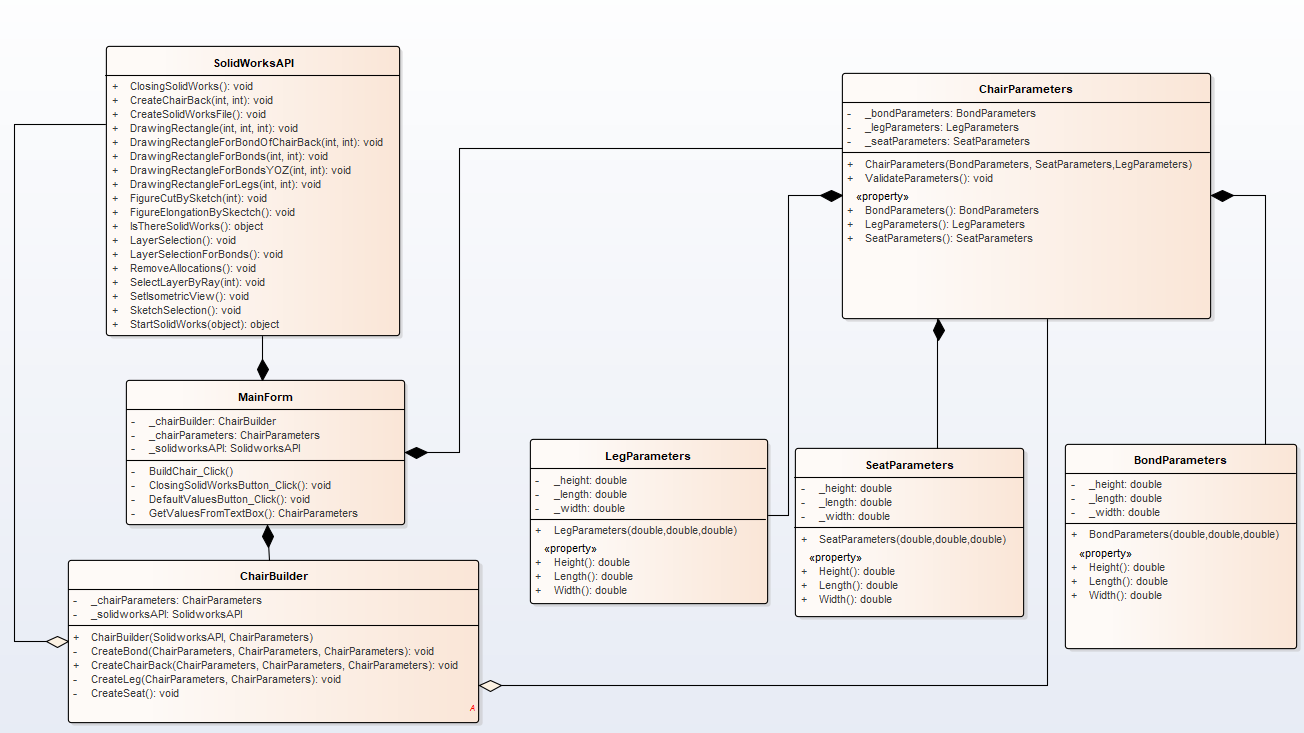


Рисунок 4.3 – Итоговая диаграмма классов.

Были произведены следующие изменения:

* Полностью обновлены методы класса SolidWorksAPI, так как на этапе проектирования они были взяты приблизительно;
* В классе MainForm были добавлены методы, без которых использование программы было бы менее комфортным для конечного пользователя;
* В классе ChairBuilder добавлен метод CreateChairBack, который создает спинку табурета. Метод CreateChairBack был реализован в связи с введением дополнительной функциональности.

В классе SolidWorksAPI были реализованы следующие методы:

* ClosingSolidWorks – метод, который необходим для закрытия SolidWorks;
* CreateChairBack – метод, создающий спинку табурета;
* CreateSolidWorksFile – метод, создающий файл в котором моделируется объект;
* DrawingRectangle – метод рисующий квадрат для сиденья табурета;
* DrawingRectangleForBonds – метод рисующий квадрат для связей табурета;
* DrawingRectangleForBondsOfChairBack – метод, рисующий квадрат для связей в спинке табурета;
* DrawingRectangleForBondsYOZ – метод, рисующий квадрат для связей табурета по плосоксти Z;
* DrawingRectangleForLegs – метод, рисующий квадрат для ножек табурета;
* FigureCutBySketch – метод, для вырезания по эскизу;
* FigureElongationBySketch – метод, вытягивающий объект по эскизу;
* IsThereSolidWorks – метод, проверяющий наличие SolidWorks 2020 на персональном компьютере пользователя;
* LayerSelection – метод, необходимый для выбора вида и плосокости;
* LayerSelectionForBonds – метод, необходимый для выбора вида и плоскости при постройке связей;
* RemoveAllocations – метод, убирающий выделения с объекта моделирования;
* SelectLayerByRay – метод, необходимый для смены осей;
* SetIsometricView – метод, необходимый для возврата вида в изометрический;
* SketchSelection – метод, для выбора эскиза;
* StartSolidWorks – метод, запускающий SolidWorks 2020.

В класс MainForm были добавлены следующие методы:

* ClosingSolidWorksButton\_Click – метод, необходимый для закрытия SolidWorks 2020, по нажатию кнопки;
* DefaultValuesButton\_Click – метод, выставляющий параметры по умолчанию в поля ввода параметров;
* GetValuesFromTextBox – метод, забирающий введенные пользователем параметры в поля ввода.

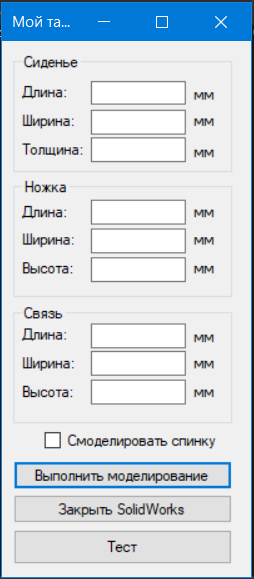
В класс ChairBuilder были добавлены следующие методы:

* CreateChairBack – метод, запускающий моделирование спинки табурета.

# 5 Описание программы для пользователя

Макет пользовательского интерфейса – это визуальное статическое представление концепции интерфейса пользователя. Интерфейс не только решает проблему взаимодействия с приложением, но и делает это взаимодействие максимально комфортным.[16]

Плагин состоит из диалогового окна, которое имеет 9 полей ввода параметров, 3 кнопки и 1 checkBox. При запуске имеет следующий вид (рисунок 5.1).

  
Рисунок 5.1 – Меню плагина

Если ввести неверные параметры, после нажатия кнопки «Выполнить моделирование», высветится окно с просьбой ввести правильные параметры в поля ввода (рисунок 5.2).

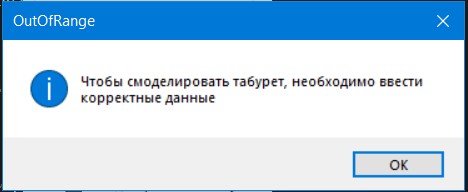
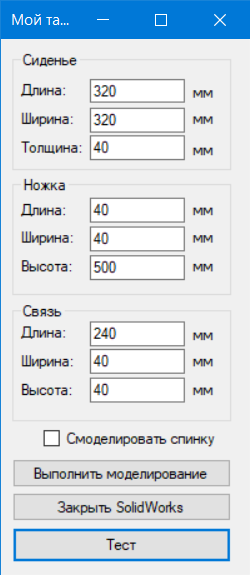
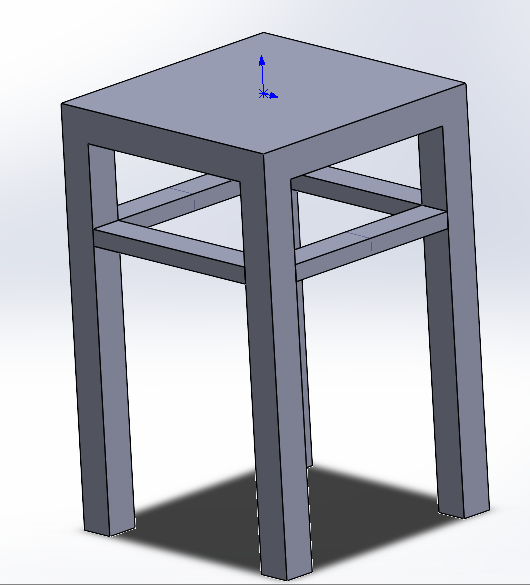


Рисунок 5.2 – Ошибка ввода параметра

Нажав, на «Тест», форма с полями для ввода данных примет следующий вид (рисунок 5.3).

  
Рисунок 5.3 – Окно плагина с параметрами по умолчанию

После ввода необходимых параметров, построить деталь в САПР SOLIDWORKS 2020 можно с помощью кнопки «Выполнить моделирование». Табурет, построенный по заданным параметрам по умолчанию, выглядит следующим образом (рисунок 5.4).

  
Рисунок 5.4 – Табурет, построенный по заданным параметрам по умолчанию

Также реализована дополнительная функциональность, после нажатия пользователем на пункт «Смоделировать спинку», табурет моделируется со спинкой.

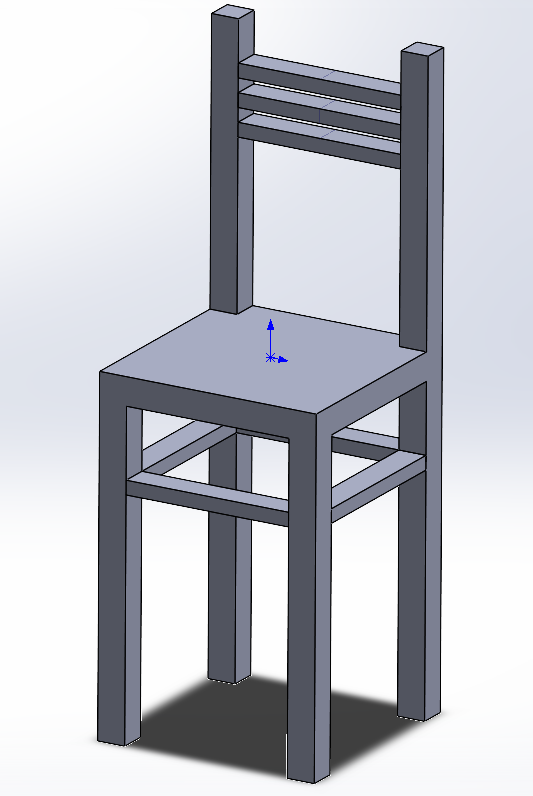


Рисунок 5.5 – Табурет со спинкой (стул)

# 6 Тестирование программы

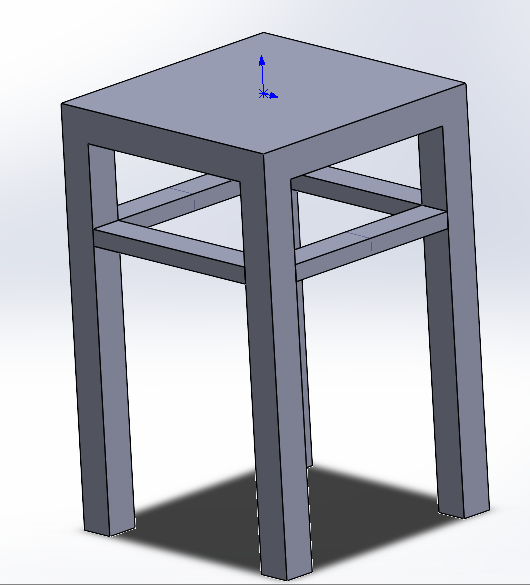
Тестирование позволяет убедиться в работоспособности программы, выявлять ошибки при изменении какого-либо функционала.

# 6.1 Функциональное тестирование

При функциональном тестировании проверялось корректность работы плагина «Табурет», а именно, соответствие полученного результата в виде трехмерной модели, с входными параметрами. [17]

Проведено тестирование максимальных и минимальных параметров модели.

Ниже на рисунке 6.1 представлена проверка размеров модели с минимальным введенными параметрами (ширина сиденья 320 мм, длина сиденья 320 мм, толщина сиденья 40 мм, длина ножки 40 мм, ширина ножки 40 мм, высота ножки 500 мм, длина связи 240 мм, ширина связи 40 мм, высота связи 40мм).

  
Рисунок 6.1 – Модель с минимальными введенными параметрами

Ниже на рисунке 6.2 представлена проверка размеров модели с максимальными введенными параметрами (ширина сиденья 500 мм, длина сиденья 500 мм, толщина сиденья 100 мм, длина ножки 100 мм, ширина ножки 100 мм, высота ножки 1000 мм, длина связи 300 мм, ширина связи 100 мм, высота связи 100 мм)

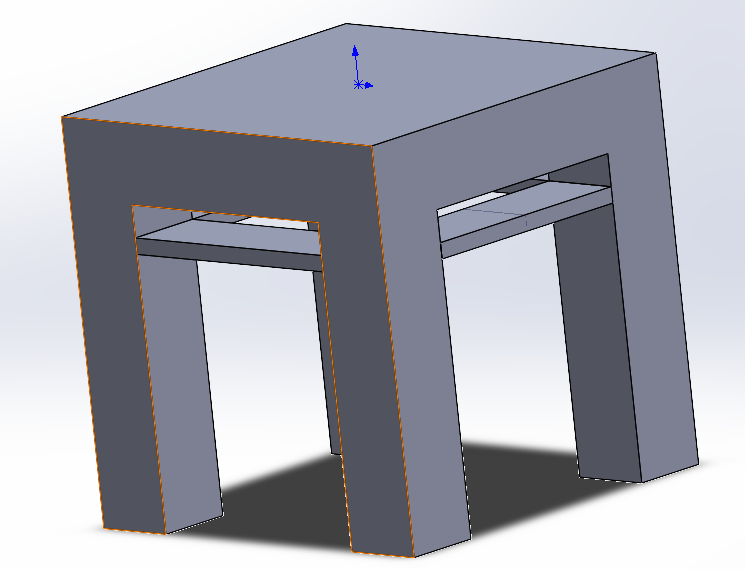
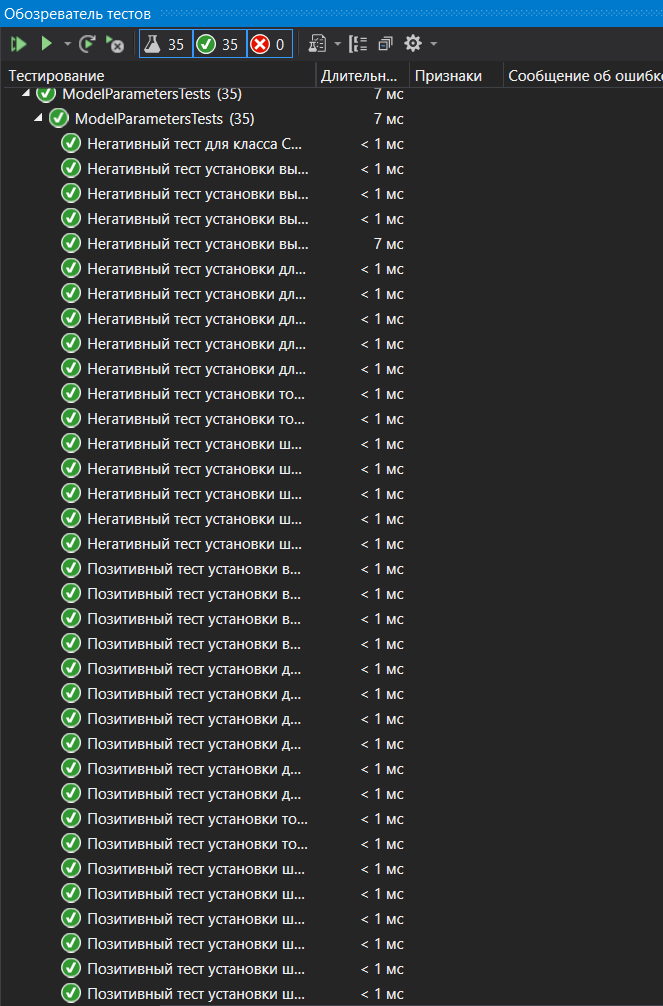


Рисунок 6.2 - Модель с максимально веденными параметрами

# 6.2 Модульное тестирование

В целях проверки корректности работы методов и свойств классов при помощи тестового фреймворка NUnit версии 3.12 проведено модульное тестирование [18], проверялись открытые поля и методы. На рисунке 6.3 представлено тестирование классов «BondParameters», «SeatParameters» и «LegParameters», «ChairParameters». Класс «SolidWorksAPI» не тестировался, поскольку хранит в себе API SolidWorks (описание тестоовых случаев находится в приложении А.1).

  
Рисунок 6.3 – Тестирование классов «BondParameters», «SeatParameters» и «LegParameters», «ChairParameters»

# 6.3 Нагрузочное тестирование

В целях проверки производительности работы плагина, было проведено нагрузочное тестирование [19]. Тестирование производилось на ПК со следующей конфигурацией:

Процессор: Intel Core i5 – 9300H

Видеоадаптер: Nvidia GTX 1650

ОЗУ: 8Gb DDR4 2666 Mhz

На рисунке 6.5 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 «logs.txt».

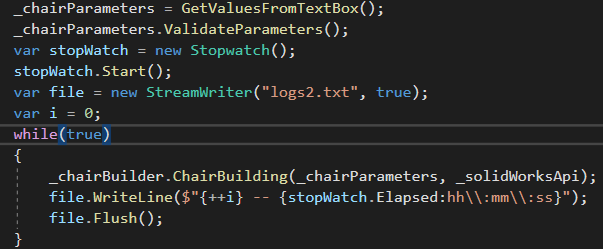


Рисунок 6.5 – Зацикливание перестроения фигуры

На графике изображенном на рисунке 6.6 в текущей главе, ось «X» – время в секундах, ось «Y» – количество построенных деталей. На графике изображенном на рисунке 6.7 ось «Х» - количество построенных деталей, ось «Y» - количество потребляемой оперативной памяти. На протяжении всех тестов (продолжительностью до сбоя SolidWorks) общая загруженность процессора была в пределах 22 процентов, потребление ОЗУ плагином прямолинейное от 14мб до 36мб.

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

* ширина сиденья 320 мм;
* длина сиденья 320 мм;
* толщина сиденья 40 мм;
* длина ножки 40 мм;
* ширина ножки 40 мм;
* высота ножки 500 мм;
* длина связи 240 мм;
* ширина связи 40 мм;
* высота связи 40мм.

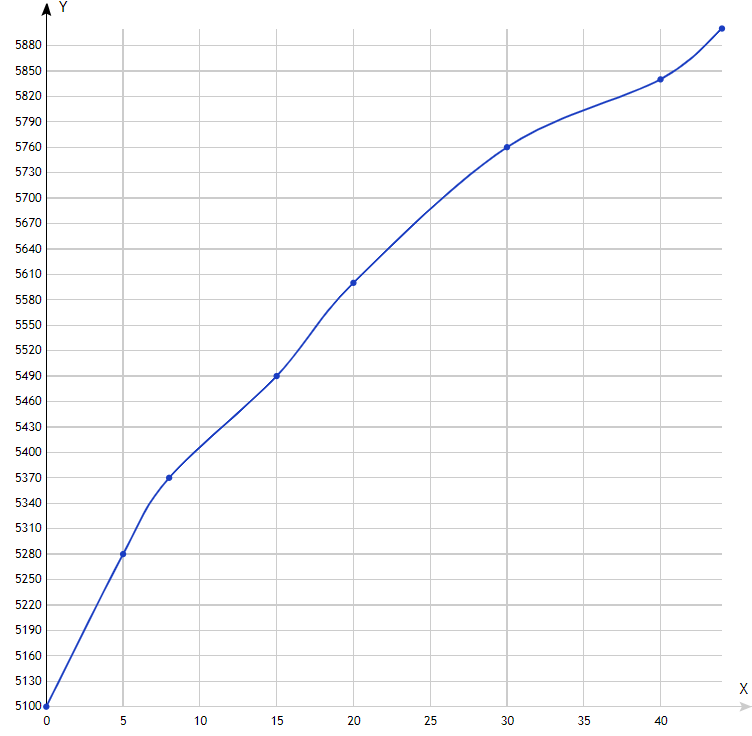
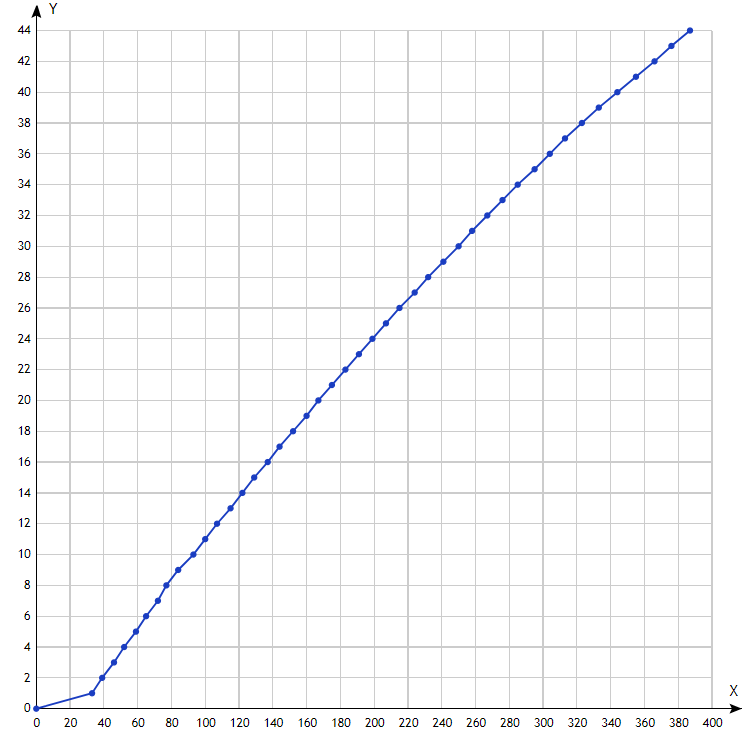


Рисунок 6.6 – График зависимости загруженности памяти от количества деталей

  
Рисунок 6.7 – График зависимости времени от количества построенных деталей с параметрами по умолчанию

Исходя из вышеуказанных графиков на рисунках 6.6 и 6.7, построение первой детали самое долгое – 33 секунды. После первого построения детали, построение последующих деталей занимало от 6-и до 10-и секунд, в зависимости от количества построенных деталей. Также можно заметить, то что в среднем после каждой 8-й детали время постройки следующих деталей занимало на 1 секунду больше времени. На рисунке 6.7 видно, что до запуска плагина, было занято 5100 МБ оперативной памяти системой и сторонними процессами, которые к самому плагину отношения не имеют. Также при запуске плагином процесса SolidWorks, оперативная память при постройке первых пяти деталей потребляется активнее, после чего потребление оперативной памяти при создании новых моделей идет на спад.

# Заключение

В ходе выполнения лабораторных работ были изучены предметная область проектирования, предмет проектирования, аналоги предмета проектирования, API, функциональное и нагрузочное тестирование и на основании полученных данных были спроектированы UML диаграммы классов, разработан плагин для создания 3D моделей «Табурет» в САПР SOLIDWORKS 2020, и 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: 15.05.2020);
2. Visual Studio [Электронный ресурс]. – URL: <https://visualstudio.microsoft.com/ru/> (дата обращения: 15.05.2020);
3. SOLIDWORKS 2020 [Электронный ресурс]. – URL: <https://www.solidworks.com/> (дата обращения: 15.05.2020);\
4. ГОСТ 13025.2-85 «Мебель бытовая. Функциональные размеры мебели для сиденья и лежания». [Электронный ресурс] – Режим доступа <http://docs.cntd.ru/document/1200017613> (дата образения 15.05.2020)
5. SolidWorks.Interop.sldworks Namespace. SOLIDWORKS API Help. [Электронный ресурс]. – URL: https://help.solidworks .com/2019/English/api/sldworksapi/SolidWorks.Interop.sldworks~SolidWorks.Interop.sldworks\_namespace.html (дата обращения: 15.05.2020);
6. NUnit [Электронный ресурс]. – URL: <https://nunit.org/> (дата обращения: 15.05.2020);
7. Начало работы с WinForms [Электронный ресурс]. – URL: <https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/> (дата обращения: 15.05.2020);
8. Мебельные САПР [Электронный ресурс] – Режим доступа: <http://www.cadcatalog.ru/a_statiy/meb_sapr.html> (дата обращения 15.05.2020)
9. AUTOCAD (САПР) [Электронный ресурс] – Режим доступа <https://www.autodesk.ru/products/autocad/overview> (дата обращения 15.05.2020)
10. Элекран софт (САПР) [Электронный ресурс] – Режим доступа <http://elecran.com.ua/mebel/program> (дата обращения 15.05.2020)
11. BCAD Мебельщик (САПР) [Электронный ресурс] – Режим доступа <https://www.bcad.ru/news/> (дата обращения 15.05.2020)
12. Woody (САПР) [Электронный ресурс] – Режим доступа <http://www.woodypro.ru/> (дата обращения 15.05.2020)
13. К3-Мебель (САПР) [Электронный ресурс] – Режим доступа <https://k3-mebel.ru/> (дата обращения 15.05.2020)
14. Новые технологии в программировании: учебное пособие / А.А. Калентьев, Д. В. Гарайс, А. Е. Горяинов. – Томск, 2014. − 176 стр.
15. UML. Основы, 3-е издание – Мартин Фаулер, Кендалл Скотт. Издательство Символ Плюс, Москва, 2004 год – 192 с.
16. Макет пользовательского интерфейса. [Электронный ресурс]. – Режим доступа: http://www.visualpharm.ru/design\_faq/kak-vyiglyadit-maket-interfeysa.html (дата обращения: 19.02.2020);
17. Функциональное тестирование [Электронный ресурс]. – URL: https://daglab.ru/funkcionalnoe-testirovanie-programmnogo-obespechenija/ (дата обращения: 13.04.2020);
18. Юнит-тестирование для чайников [Электронный ресурс]. – URL: <https://habr.com/ru/post/169381/> (дата обращения: 15.05.2020);
19. Нагрузочное тестирование: с чего начать и куда смотреть [Электронный ресурс]. – URL: <https://habr.com/ru/company/jugru/blog/329174/> (дата обращения: 15.05.2020);

# Приложение А

(Справочное)

Описание полей и методов используемых для проверки тестовых случаев класса ModelTests представлено в таблице А.1

Таблица А.1 – Тестовые случаи

|  |  |
| --- | --- |
| Наименование | Описание |
| SetSeatThickness\_NegativeTest(int Thickness)  (20, 120) | Тест для создания объекта SeatParameters c некорректными значениями. |
| SetSeatThickness\_PositiveTest(int Thickness)  (40, 100) | Тест на создание объекта SeatParameters со значениями входящими в диапазон правильных значений. |
| SetSeatLength\_NegativeTest(int Length)  (310, 510) | Тест для создания объекта SeatParameters c некорректными значениями. |
| SetSeatLength\_PositiveTest(int Length)  (320, 500) | Тест на создание объекта SeatParameters со значениями входящими в диапазон правильных значений. |
| SetSeatWidth\_NegativeTest(int Width, int Length)  (310, 320, 510, 500) | Тест для создания объекта SeatParameters c некорректными значениями. |
| SetSeatWidth\_PositiveTest(int Width, int Length)  (320, 320, 500, 500) | Тест на создание объекта SeatParameters со значениями входящими в диапазон правильных значений. |

Продолжение таблицы А.1

|  |  |
| --- | --- |
| SetLegHeight\_NegativeTest(int Height)  (300, 1020) | Тест для создания объекта LegParameters c некорректными значениями. |
| SetLegHeight\_PositiveTest(int Height)  (500, 1000) | Тест на создание объекта LegParameters со значениями входящими в диапазон правильных значений. |
| SetLegLength\_NegativeTest(int Length)  (30, 110) | Тест для создания объекта LegParameters c некорректными значениями. |
| SetLegLength\_PositiveTest(int Length)  (40, 100) | Тест на создание объекта LegParameters со значениями входящими в диапазон правильных значений. |
| SetLegWidth\_NegativeTest(int Length, int Width)  (45, 40, 95, 100) | Метод для создания объекта LegParameters с некорректными значениями |
| SetLegWidth\_PositiveTest(int Length, int Width)  (40, 40, 100, 100) | Тест на создание объекта LegParameters со значениями входящими в диапазон правильных значений. |
| SetBondWidth\_PositiveTest(int Width)  (40, 100) | Тест на создание объекта BondParameters со значениями входящими в диапазон правильных значений. |
| SetBondWidth\_NegativeTest(int Width)  (0, 0) | Тест для создания объекта BondParameters с некорректными значениями |

Продолжение таблицы А.1

|  |  |
| --- | --- |
| SetBondHeight\_PositiveTest(int Width, int Height)  (40, 40, 100, 100) | Тест на создание объекта BondParameters со значениями входящими в диапазон правильных значений. |
| SetBondHeight\_NegativeTest(int Width, int Height)  (40, 45, 100, 110) | Тест для создания объекта BondParameters с некорректными значениями |
| SetBondLength\_NegativeTest(int Length)  (240, 320) | Тест на создание объекта BondParameters со значениями входящими в диапазон правильных значений. |
| SetBondLength\_PositiveTest(int Length)  (0, 0) | Тест для создания объекта BondParameters с некорректными значениями |
| SetChairParameters\_NegativeTest(BondParameters bondParameters, SeatParameters seatParameters, LegParameters legParameters)  (null, null, null) | Тест для создания объекта ChairParameters с некорректными значениями |