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

\_\_\_\_\_\_\_\_\_\_\_\_\_А.А. Исанов

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

Руководитель

К.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_А.А. Калентьев

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

Томск 2020

**Оглавление**

[1 Введение 3](#_Toc40556141)

[2 Постановка и анализ задачи 4](#_Toc40556142)

[2.1 Описание предмета проектирования 4](#_Toc40556143)

[2.2 Выбор инструментов и средств реализации 5](#_Toc40556144)

[2.3 Назначение плагина 6](#_Toc40556145)

[2.4 Описание аналогов разрабатываемого продукта 6](#_Toc40556146)

[2.4.1 БАЗИС – Шкаф 6](#_Toc40556147)

[2.4.2 Оборудование: Металлоконструкции 7](#_Toc40556148)

[2.4.3 BricsCAD 9](#_Toc40556149)

[3 Описание реализации 12](#_Toc40556150)

[3.1 Диаграмма прецедентов плагина 12](#_Toc40556151)

[3.2 Диаграмма классов 14](#_Toc40556152)

[4 Описание программы для пользователя 16](#_Toc40556153)

[5 Тестирование программы 20](#_Toc40556154)

[5.1 Функциональное тестирование 20](#_Toc40556155)

[5.2 Модульное тестирование 22](#_Toc40556156)

[5.3 Нагрузочное тестирование 23](#_Toc40556157)

[Заключение 26](#_Toc40556158)

[Список использованных источников 27](#_Toc40556159)

[Приложение А 28](#_Toc40556160)

# 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.

Практическая реализация методов и идей автоматизированного моделирования происходит в рамках систем автоматизированного проектирования (САПР). Однако мало создать высокопроизводительные современные САПР. Надо уметь их эффективно использовать. Для этого нужны квалифицированные инженеры-пользователи САПР. В рамках современного «компьютеризированного» общества инженер любой специальности, занимающийся разработкой технических устройств, должен уметь использовать средства автоматизированного проектирования. Это позволяет повысить эффективность моделирования, улучшить его качество, снизить материальные затраты и уменьшить число разработчиков.

Таким образом, целью данной работы является разработка плагина, автоматизирующего построение модели металлического уголока, для системы автоматизированного проектирования «Компас-3D» v18.1 [1], с помощью интегрированной среды разработки Visual Studio Сommunity 2019 [2].

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

# 2 Постановка и анализ задачи

В рамках лабораторных работ в соответствии с технически заданием требовалось разработать плагин, который на основе входных параметров, интегрируя с системой КОМПАС 3D [1], строит деталь «Металлический уголок». Более того, требовалось, чтоб плагин позволял изменять входные параметры в соответствии с требованиями пользователей, а именно: высота уголка, ширина уголка, диаметр отверстий, толщину уголка, расстояния от граней до центров ближайших окружностей, количество отверстий на каждой плоскости.

## 2.1 Описание предмета проектирования

Предметом проектирования является металлический уголок. Металлический уголок – это один из базовых элементов металлических конструкций.

Параметры металлического уголка:

* Высота уголка H: от 50 мм до 200 мм;
* Ширина уголка W: от 10 мм до 100 мм;
* Диаметр отверстий D: от 5 мм до 20 мм;
* Толщина уголка T: от 3 мм до 20 мм;
* Расстояние от крайней грани до центра первого отверстия L: от 10 мм до 170 мм;
* Количество отверстий на каждой плоскости: N=(H-L-M)/(D+5);
* Расстояние от грани, прилежащей к другой плоскости, до центра ближайшего отверстия M: от 10 мм до 170 мм.

Изображение предмета проектирования с обозначенными параметрами приведено на рисунке 2.1

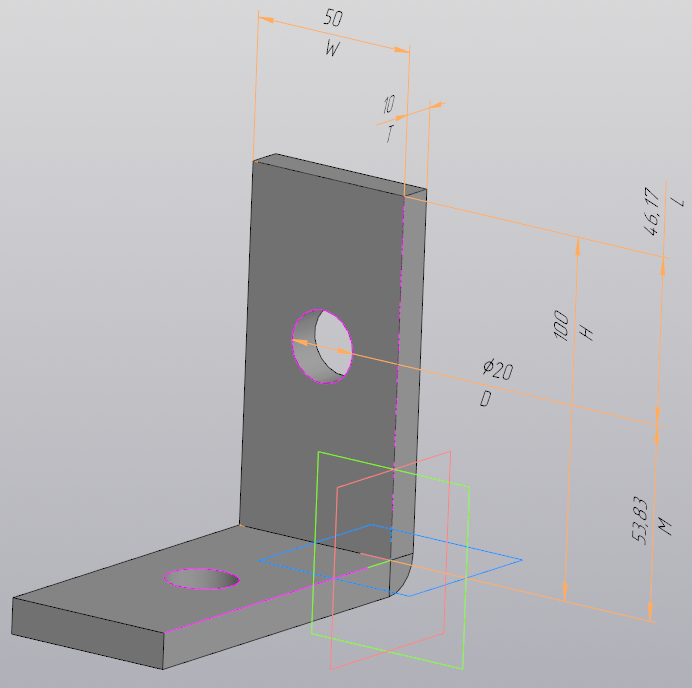


Рисунок 2.1 —­­ Модель металлического уголка в САПР «Компас-3D»

## 2.2 Выбор инструментов и средств реализации

В связи с требованием технического задания программа выполнена на языке C# в среде Microsoft Visual Studio 2019 [2], с использованием .NET Framework 4.7.0, для системы КОМПАС 3D v18.1. Инструментом тестирования и создания модульных тестов был выбран стандартный обозреватель тестов среды Microsoft Visual Studio 2019 [2] с тестовым фреймворком NUnit версии 3.12.0.

Для реализации пользовательского интерфейса использовался WinForm.

Взаимодействие плагина с системой КОМПАС 3D [1] осуществляется посредством интерфейсов, называемых API. В КОМПАС 3D на данный момент существует API двух версий API 5 [3] и API 7 [3]. Явных преимуществ между версиями нет, поскольку обе версии реализуют различные функции системы и взаимно дополняют друг друга. Для выполнения лабораторных была выбрана версия API 5, так как для полноценной реализации плагина «Металлический уголок» достаточно методов и свойств интерфейсов API 5.

## 2.3 Назначение плагина

Назначение, разрабатываемого плагина обусловлено выбором сферы его применения. В данном проекте выбором сферы применения стало проектирование металлического уголка, так как данная сфера, одна из наиболее популярных, ввиду того что данный объект необходим для построения и сборки различных конструкций. Для упрощения соединения различных конструкций порой требуются металлические уголки с необычными параметрами.

## 2.4 Описание аналогов разрабатываемого продукта

КОМПAС-3D [1] является открытой системой, что позволяет создавать дополнительные программные модули (пользовательские библиотеки) и применять их во время работы над документами. Таким образом, стандартные возможности чертежно-графического редактора и трехмерного моделирования могут быть дополнены исходя из тех специальных задач, которые приходится решать пользователю.

В настоящее время система КОМПAС-3D непрерывно расширяется плагинами и модулями, которые можно разделить на два условных класса: плагины или модули, реализующие (моделирующие) конкретные трехмерный объекты в зависимости от сферы применения.

### 2.4.1 БАЗИС – Шкаф

Базис–Шкаф [4] – это модуль проектирования с использованием параметров элементов модели и соотношений между этими параметрами. Параметрическое проектирование позволяет за короткое время «проиграть» (с помощью изменения параметров или геометрических соотношений) различные конструктивные схемы. Возможности модуля Базис-Шкаф сокращают время работы над проектом и уменьшают вероятность ошибок. На рисунке 2.1 представлен пример построения крепежного уголка в программе Базис–Шкаф.

Преимущества Базис-Шкаф:

* Модель изделия создается одним щелчком мыши благодаря заданию основных параметров;
* Модуль имеет огромное количество автоматически выполняемых функций;
* Возможность редактирования параметров одной командой. Изделие перестраивается автоматически.
* На создание модели шкафа в программе Базис-Шкаф требуется от двух до десяти минут.

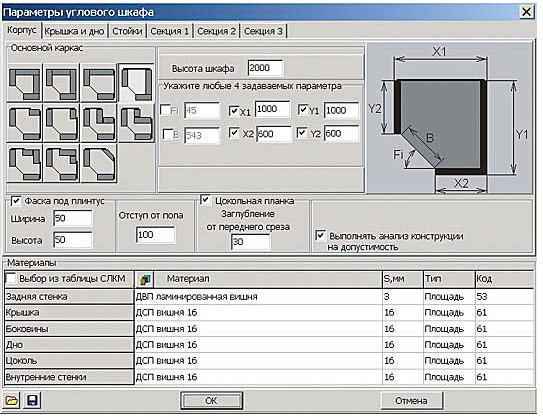


Рисунок 2.1 – Пример работы программы Базис–Шкаф

### 2.4.2 Оборудование: Металлоконструкции

Оборудование: Металлоконструкции [5] — приложение для КОМПАС-3D, предназначенное для автоматизации работ по проектированию конструкций из профильного металлопроката. Приложение позволяет быстро проектировать всевозможные рамы и каркасы, автоматически создавать комплект документации.

Создание металлоконструкции в приложении начинается с построения Трехмерного каркаса — геометрических осей, которые являются эскизом конструкции. После чего для каждой из осей назначается профиль. Сортамент профиля может выбираться из нового Каталога профилей, входящего в комплект поставки приложения, либо из Справочника Материалы и Сортаменты для КОМПАС. Для удобства построения и редактирования металлоконструкции в приложении реализован механизм Характерных точек, который позволяет задавать длину и угол поворота профиля непосредственно в окне построения. При изменении Трехмерного каркаса металлоконструкция перестроится автоматически.

После назначения профилей необходимо проработать отдельные узлы металлоконструкции. Для этого в приложении есть специальные инструменты. Можно корректировать длины деталей, задавать угловую или стыковую разделки, строить дополнительные элементы в виде ребер жесткости или фасонок.

Для созданной с помощью приложения конструкции можно автоматически получить спецификацию либо любые другие виды настраиваемых отчетов. Металлоконструкция, спроектированная в приложении, может быть проверена на наличие пересечений. Трехмерная модель позволяет сразу выявить возможные нестыковки. Оборудование: Металлоконструкции позволяет избежать дополнительных затрат на материал и инструмент.

На рисунке 2.2 представлен интерфейс, каталога выбора структуры швеллера, приложения Оборудование: Металлоконструкции.

Рисунок 2.2 – Пример работы приложения Оборудование: Металлоконструкции

### 2.4.3 BricsCAD

BricsCAD [6] — система автоматизированного проектирования (САПР), которая объединяет 2D черчение и 3D моделирование в едином формате .dwg. BricsCAD разрабатывается бельгийской компанией Bricsys с 2002 года. Программа выпускается на 18 языках и доступна для операционных систем Windows, Linux и MacOS. Для BricsCAD существует более 400 приложений, позволяющих использовать его в архитектуре, строительстве, машиностроении, проектировании инженерных сетей, электрике, автоматике, ГИС и других сферах проектирования.

BricsCAD Classic обеспечивает полный набор функциональности для работы в 2D. Включает совместную работу в облаке, поддержку динамических блоков, параметризацию чертежей, инструментальные палитры, подшивки, экспорт данных из чертежа, а также расширение возможностей с помощью LISP приложений. Кроме того, версия Classic позволяет работать в 3D, создавать и редактировать пространственные сети и поверхности, что вполне достаточно для моделирования поверхности рельефа и выполнения несложных трехмерных проектов.

BricsCAD Pro содержит все функциональные возможности BricsCAD Classic и дополнительно предлагает средства твердотельного 3D моделирования с поддержкой технологии прямого вариационного моделирования, двумерные и 3D-аппаратные библиотеки, рендеринг высокой четкости, библиотеку материалов рендеринга, просмотр механических сборок и систему разработки, совместимую с AutoCAD ObjectARX, которая поддерживает сотни сторонних прикладных программ. Кроме того, версия Pro обеспечивает автоматическое создание 2D видов и разрезов по трехмерной модели и фотореалистичную визуализацию.

BricsCAD Platinum включает все возможности версии Pro и дополнительно предлагает возможности трехмерной параметризации моделей и сборок, интеллектуальное распознавание модели, моделирование сборок, деформационное моделирование, автоматическое составление спецификаций, сравнение 3D моделей. Также возможности версии Platinum могут быть расширены применением модуля Sheet Metal (проектирования изделий из листового металла) и модуля BIM (информационное моделирование зданий). Интерфейс этой версии программы представлен на рисунке 2.3.

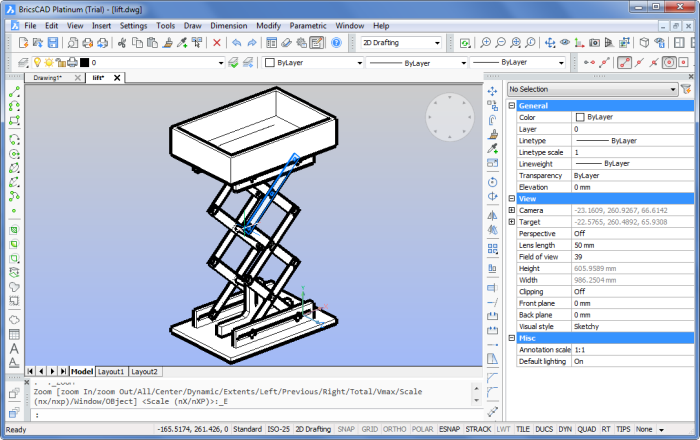


Рисунок 2.3 – Интерфейс программы BricsCAD Platinum

# 3 Описание реализации

На этапе разработки проекта программы для формального описания архитектурной особенности, пользовательского сценария системы был выбран унифицированный язык моделирования (UML) [7]. На основе UML построены: диаграммы вариантов использования диаграммы классов, диаграммы пакетов.

В процессе реализации диаграммы классов, вариантов использования и пакетов были дополнены.

Дополнения были обусловлены нарастанием функционала в реализации мелких детальных особенностей и изменений требований заказчика.

## 3.1 Диаграмма прецедентов плагина

Прецеденты [7] – это технология определения функциональных требований к системе. Работа прецедентов заключается в описании типичных взаимодействий между пользователем системы и самой системой.

Прецеденты представляют собой ценный инструмент для понимания функциональных требований к системе.

Рассмотрим изначальную версию диаграммы прецедентов на рисунке 3.1.

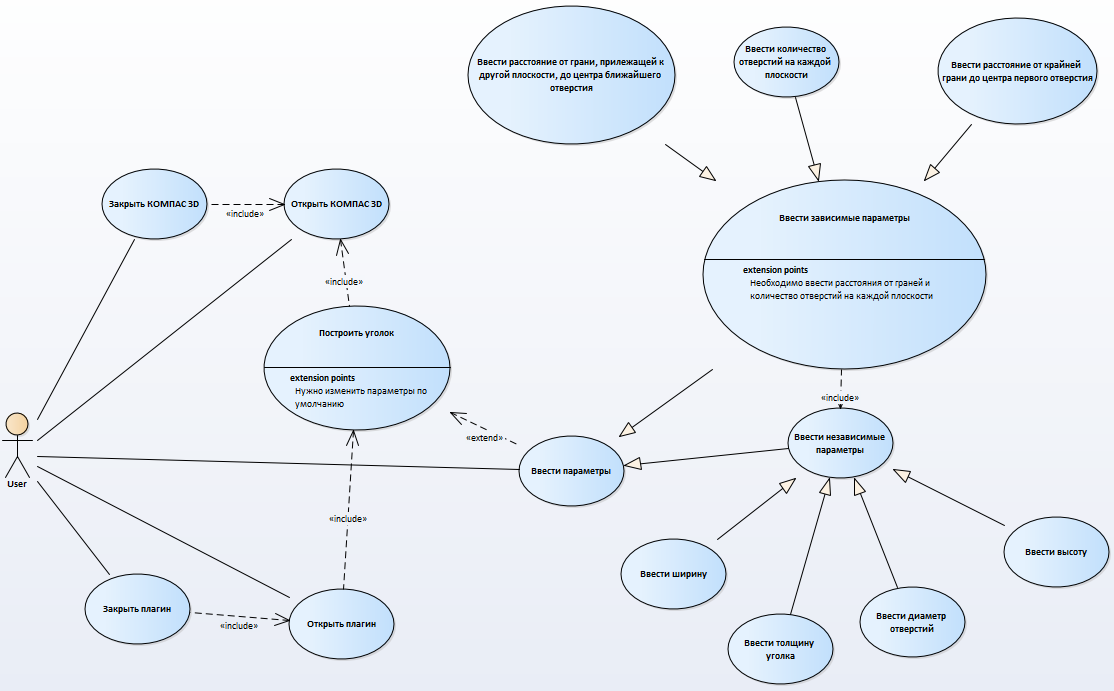


Рисунок 3.1 – Изначальная диаграмма прецедентов плагина

Действующим лицом выступает пользователь.

Для пользователя доступны следующие возможности:

* Ввести параметры;
* Создать модель металлического уголка;
* Запустить плагин;
* Закрыть плагин.

В результате изменений, внесенных заказчиком, диаграмма прецедентов подверглась изменениям.

Добавлены возможности для выбора и сохранения параметров каждой плоскости металлического уголка.

Рассмотрим измененную диаграмму прецедентов плагина на рисунке 3.2.

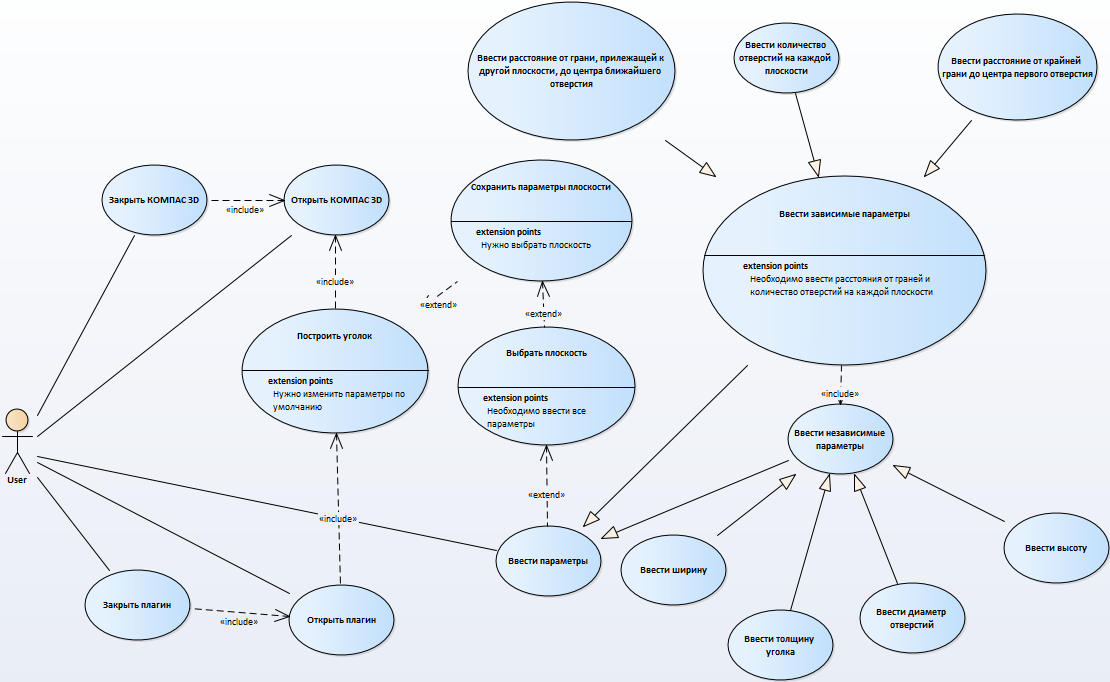


Рисунок 3.2 – Измененная диаграмма прецедентов плагина

## 3.2 Диаграмма классов

Диаграмма классов [7] описывает типы объектов системы и различного рада статические отношения, которые существуют между ними. На диаграммах классов отображаются также свойства классов, операции классов и ограничения, которые накладываются на связи между объектами. В UML термин функциональность применяется в качестве основного термина, описывающего и свойства, и операции класса. Целью создания диаграммы классов является графическое представление статической структуры декларативных элементов системы.

Рассмотрим изначальную версию диаграммы классов на рисунке 3.3.

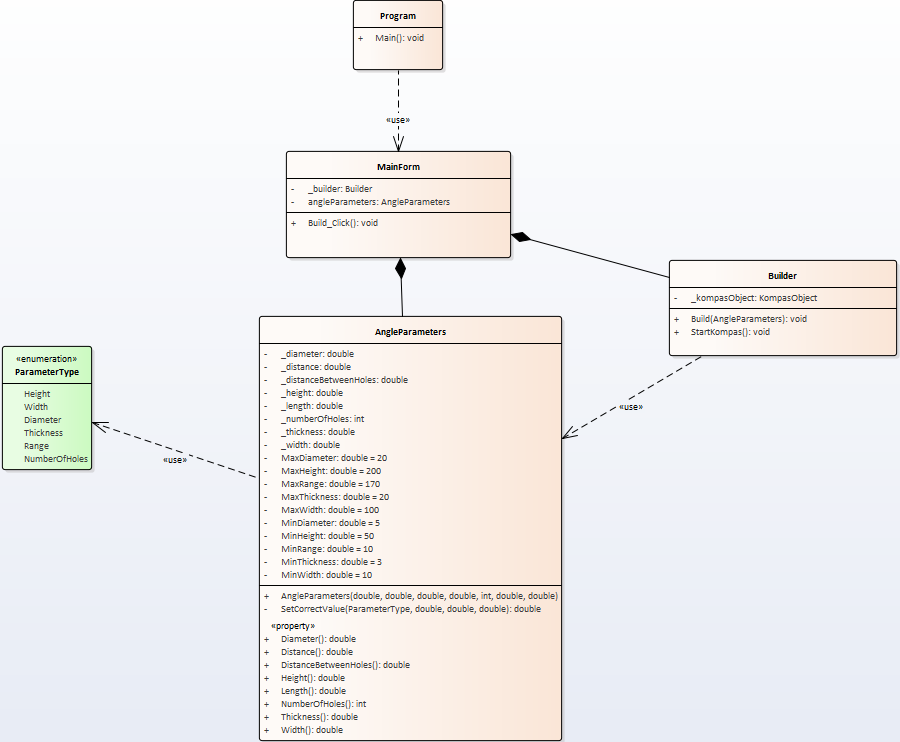


Рисунок 3.3 – Изначальная диаграмма классов плагина

Элемент модели это одна из структурных составляющих модели. В данном случае обязательными элементами являются параметры плоскостей и отверстий на них.

Параметром элемента модели является числовая величина, представляющая определенное геометрическое свойство элемента модели.

Класс AngleParameter хранит данные параметров элементов модели.

Класс MainForm является формой пользовательского интерфейса. Реализует методы, используемые для взаимодействия с пользователем.

Класс Builder хранит экземпляр класса построителя модели и реализует методы подключения к САПР КОМПАС-3D и построение металлического уголка.

В результате изменений, внесенных заказчиком, диаграмма классов подверглась незначительным изменениям.

Были добавлены методы BuildHole и BuildPlane в класс Builder, класс AngleParameters был переименован в PlaneParameters и был добавлен класс FormTools, для уменьшения вероятности ошибки при введении названий элементов, параметров и форм. Также в MainForm теперь хранится две сущности параметров PlaneParameters для каждой из двух плоскостей металлического уголка.

Рассмотрим измененную диаграмму классов плагина на рисунке 3.4

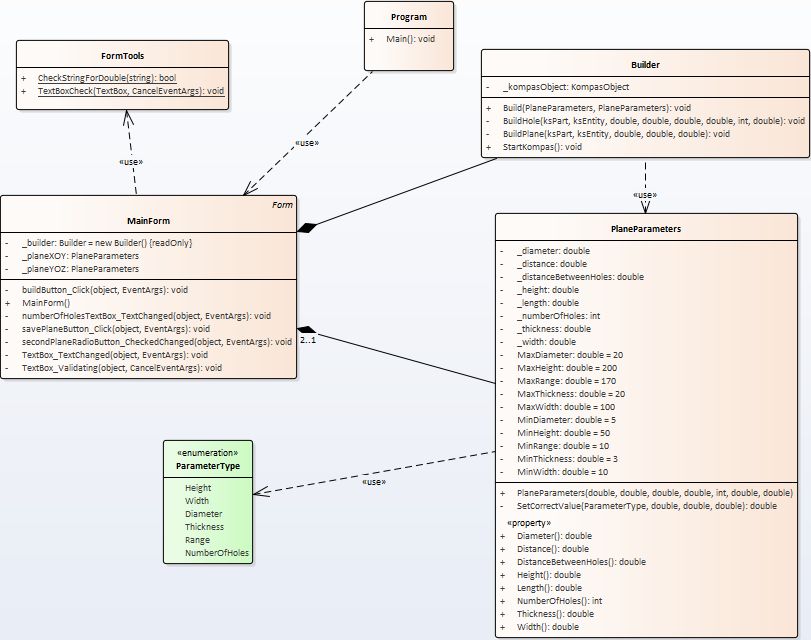


Рисунок 3.4 – Измененная диаграмма классов плагина

# 4 Описание программы для пользователя

Плагин представляет собой пользовательскую форму с ячейками для ввода параметров и выбора плоскости для построения модели. Запуск построения объекта осуществляется кнопкой «Построить». Макет пользовательского интерфейса, изображенный на рисунке 4.1, состоит из 4 блоков:

1. Блок задания параметров уголка;
2. Блок выбора плоскости построения;
3. Блок сохранения параметров в одну из плоскостей;
4. Блок для построения уголка в программе.

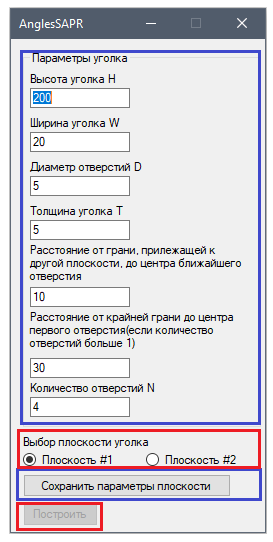


Рисунок 4.1 – Макет пользовательского интерфейса

Перед построением модели уголка пользователю необходимо задать значения его параметров во всех поля TextBox, на которые наложены ограничения:

1. Пользователь может ввести только положительные целочисленные или дробные значения;
2. При вводе значения, выходящего за допустимы диапазон, выводится сообщение о некорректном вводе с допустимыми значениями, показанное на рисунке 4.2;
3. Только при корректном заполнении всех полей кнопка «Построить» будет активна и выполнять назначенные ей действия.

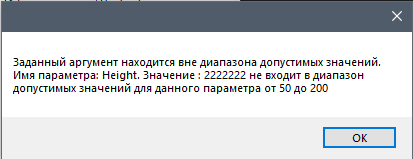


Рисунок 4.2 — Сообщение об ошибке

Нажатие кнопки “Построить” открывает систему автоматизированного проектирования «КОМПАС-3D» и создает новый пустой документ типа «Деталь» в котором производиться построение модели, если САПР «КОМПАС-3D» уже открыта, то произойдет создание только нового документа типа “Деталь”. Нажатие кнопки возможно в любой момент в течении времени заполнения параметров, начиная с момента запуска программы.

Выделенная фокусом кнопка “Построить” на диалоговом окне изображена на рисунке 4.3.

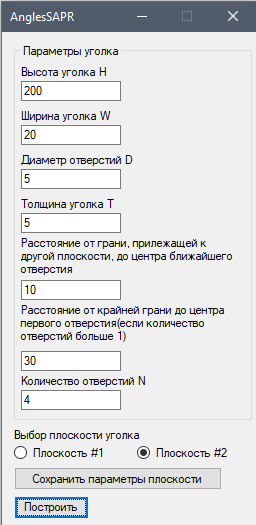


Рисунок 4.3 – Диалоговое окно с выделенной кнопкой “Построить”

Диалоговое окно документа САПР «КОМПАС-3D» после построения модели изображено на рисунке 4.4.

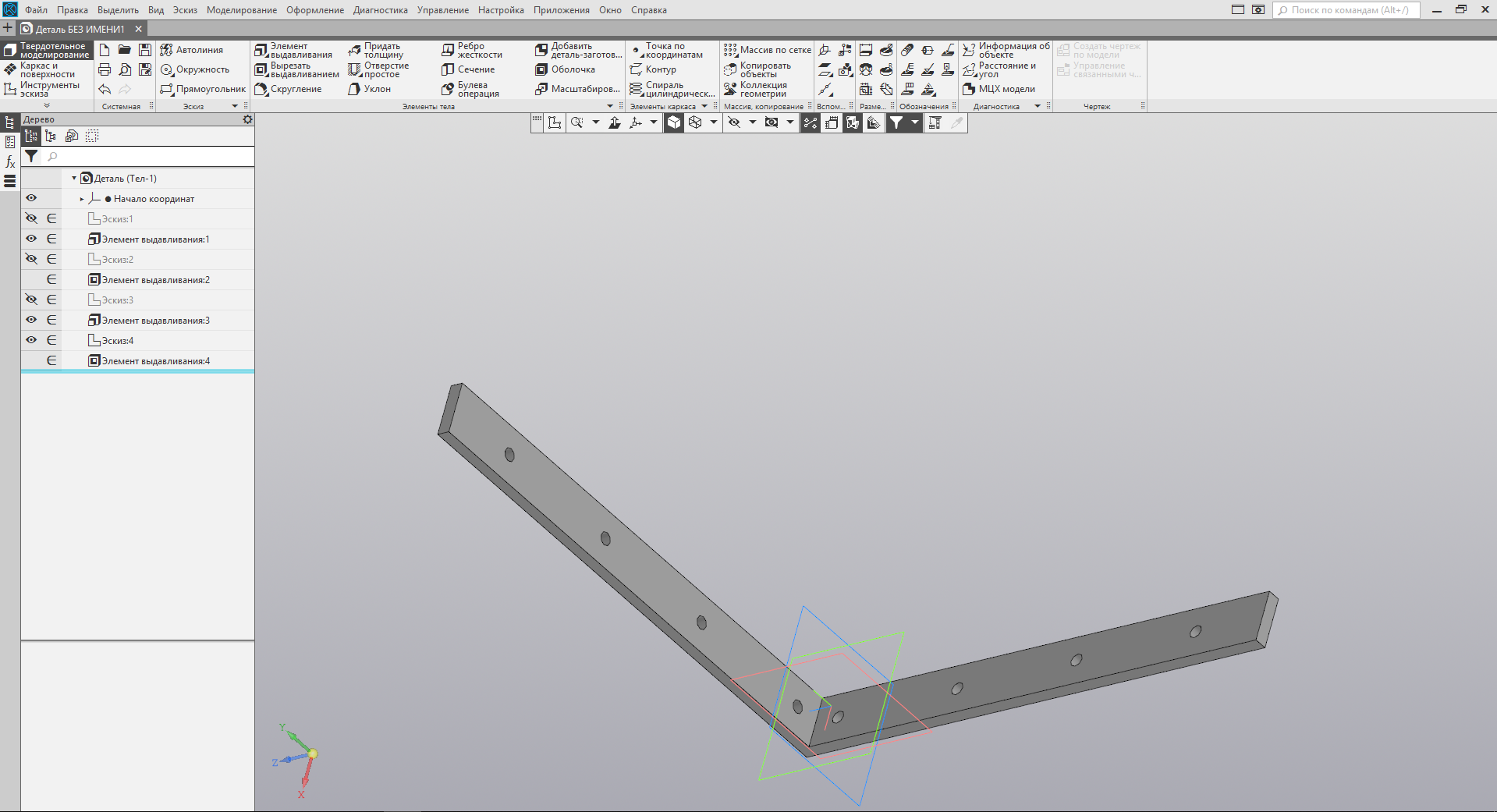


Рисунок 4.4 - Диалоговое окно документа САПР КОМПАС 3D после построения модели

# 5 Тестирование программы

Тестирование позволяет выявлять ошибки в программе в процессе разработки и при выпуске промежуточных и финальных версий приложения.

## 5.1 Функциональное тестирование

При функциональном тестировании [8] проверялась корректность работы плагина “Металлический уголок”, а именно, соответствие полученного результата в виде трехмерной модели, с входными параметрами.

Проведем тестирование максимальных и минимальных параметров модели.

Минимальные параметры:

1. Высота = 50;
2. Ширина = 11;
3. Толщина = 3;
4. Диаметр = 5;
5. Расстояния от граней = 10.

Максимальные параметры:

1. Высота = 200;
2. Ширина = 100;
3. Толщина = 20;
4. Диаметр = 20;
5. Расстояния от граней = 170.

Модель с минимальными параметрами представлена на рисунке 5.1.

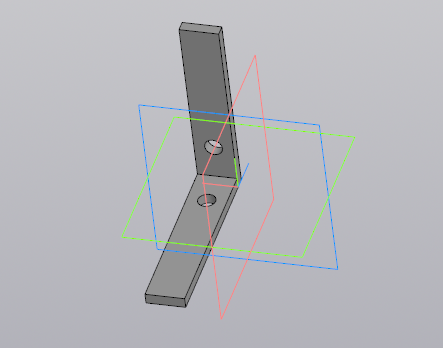


Рисунок 5.1 – Модель металлического уголка с минимальными возможными параметрами

Модель с максимальными параметрами представлена на рисунке 5.2.

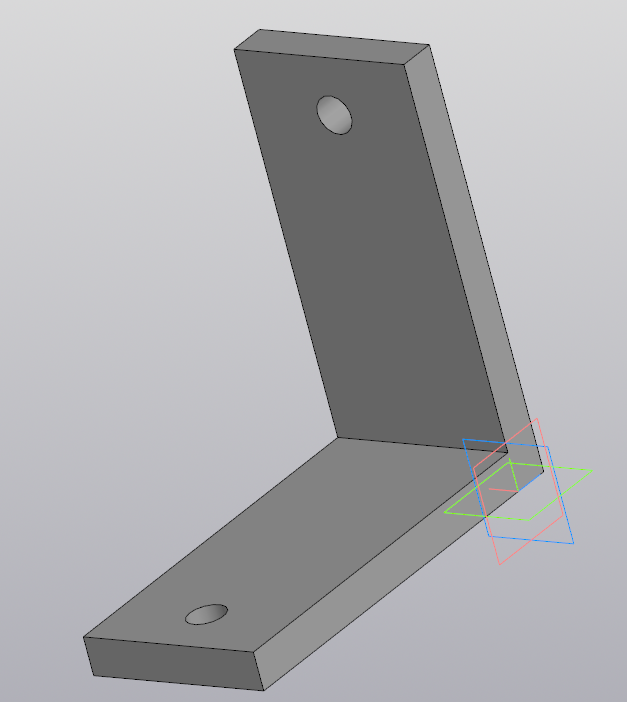


Рисунок 5.2 – Модель металлического уголка с максимальными возможными параметрами

Проведем тестирование построения модели с разными параметрами плоскостей. Результаты тестирования представлены на рисунке 5.3.

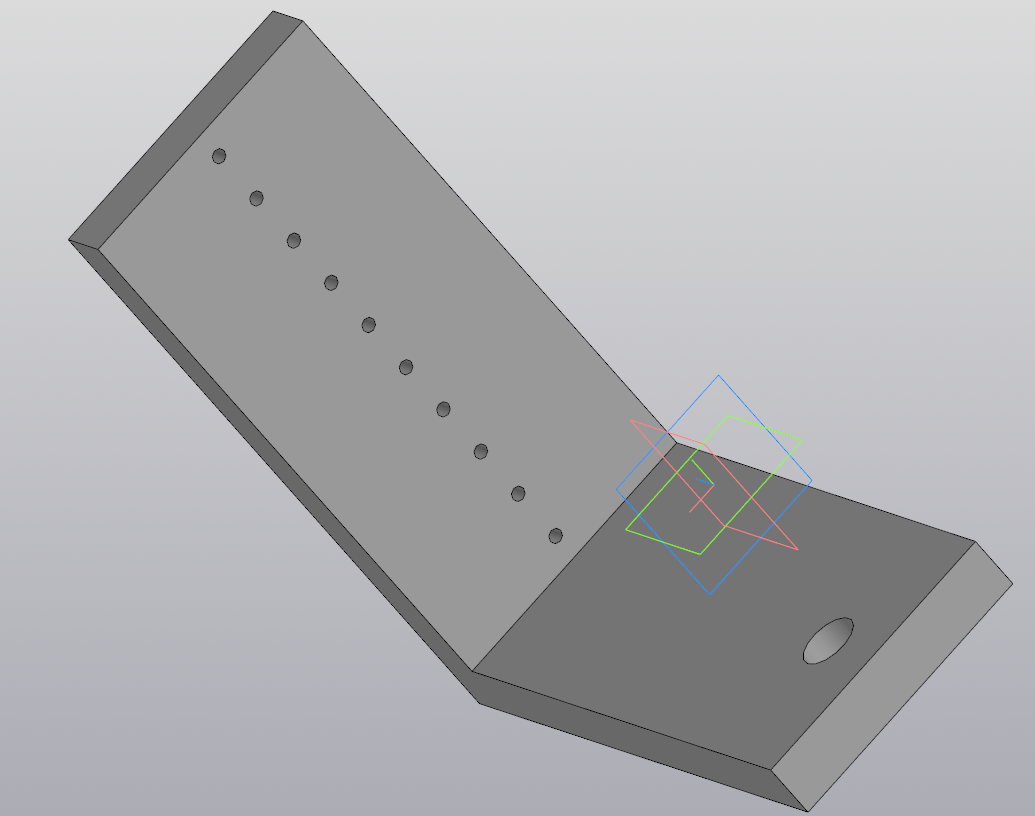


Рисунок 5.3 – Модель металлического уголка с разными параметрами отверстий на каждой плоскости

## 5.2 Модульное тестирование

В целях проверки корректности работы методов и свойств классов при помощи обозревателя тестов Visual Studio было проведено модульное тестирование [9], проверялись открытые поля и методы, для этого был создан тестовый класс:

* ModelTests – класс тестирующий свойства и методы класса PlaneParameters.

Диалоговое окно состояний запущенных тестов для классов ModelTests изображены на рисунке 5.4.

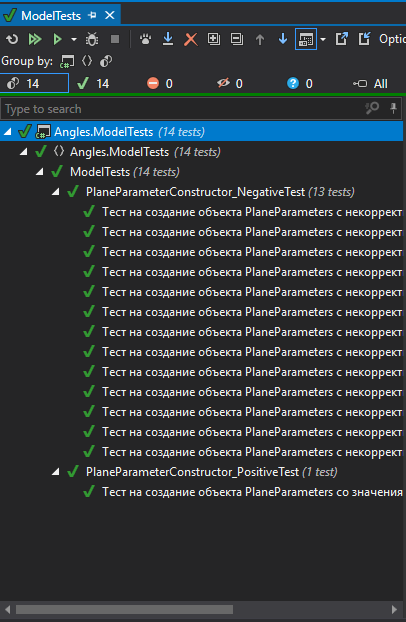


Рисунок 5.4 - Диалоговое окно состояний запущенных тестов для класса PlaneParameters

## 5.3 Нагрузочное тестирование

В целях проверки производительности работы плагина, было проведено нагрузочное тестирование [10]. Для проведения нагрузочного тестирования был добавлен таймер, который засекал время от начала до конца построения. Измерялись время, затрачиваемое на построение деталей, и потребляемая оперативная память.

Проведем тестирование для Windows 10 x64, установленной на SSD.

Конфигурация ПК:

1. Центральный процессор: Intel Core i5-6300HQ
2. ОЗУ: 8 ГБ
3. SSD 512ГБ
4. Графический процессор: GTX 960Ti

После построения 82 детали программа перестала отвечать и не возобновила работу.

Начиная с построения 18 детали, затраты оперативной памяти начали уменьшаться, за счет увеличения времени построения детали. В это же время наблюдалось значительное замедление работы интерфейса операционной системы в целом. Начиная с 16 детали общая загрузка диска достигла максимального значения. Время построения детали при построении первого экземпляра составляло 2 секунды, время построения 79 детали 3 секунды.

Графики потребляемой оперативной памяти относительно числа созданных деталей изображены на рисунке 5.9. Где y – количество используемой оперативной памяти, в мегабайтах, а x – число построенных деталей.

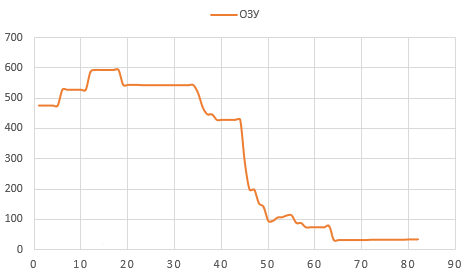


Рисунок 5.9 – Графики использования оперативной памяти относительно числа построенных деталей

Падение потребление ОЗУ в середине нагрузочного тестирования связано с тем, что плагин отсылает команды программе «Компас-3D», а она перестаёт отвечать на них также быстро, как при одиночном построении. Выходит, что ранее запущенные окна выгружаются из ОЗУ в раздел подкачки и происходит снижение потребление ОЗУ.

Графики зависимости времени построения от количества построенных деталей изображены на рисунке 5.10. Где y – время в секундах, а x – число построенных деталей.

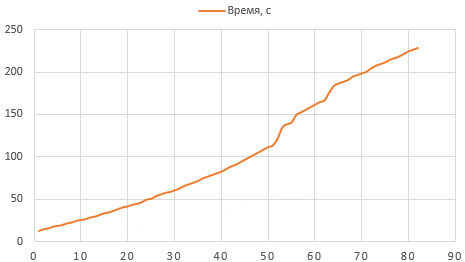


Рисунок 5.9 – Графики зависимости времени построения относительно числа построенных деталей

Так как алгоритм построения детали достаточно простой, то в результате мы видим, что разница во времени, затрачиваемом на построение детали, не сильно отличается при одноразовом построении и при пиковой нагрузке в нагрузочном тестировании. Частота обращений плагина к «Компас-3D» вызвала перезагрузку системы в виду перегрузки очереди исполнения инструкций.

# Заключение

В ходе выполнения лабораторных работ были изучены предметная область проектирования, предмет проектирования, аналоги предмета проектирования, API и на основании полученных данных были спроектированы архитектура и макет системы, создан плагин «Металлический уголок», проведены модульные, функциональные и нагрузочные тесты.

# Список использованных источников

1. КОМПАС – 3D [Электронный ресурс]. – URL: https://kompas.ru/ (дата обращения: 13.04.2020);
2. Microsoft Visual Studio 2019 [Электронный ресурс]. – URL: https://visualstudio.microsoft.com/ru/vs/ (дата обращения: 13.04.2020);
3. API 5,7. [Электронный ресурс]. – Режим доступа: https://forum.ascon.ru/index.php/board,4.0.html (дата обращения: 19.02.2020);
4. Базис-Шкаф. [Электронный ресурс]. – Режим доступа: http://mebelvopros.com/page/opisanie-moduley-bazisa (дата обращения: 19.02.2020);
5. Оборудование: Металлоконструкции. [Электронный ресурс]. – Режим доступа:https://kompas.ru/kompas-3d/application/machinery/steel-constructions-3d/ (дата обращения: 19.02.2020);
6. BricsCAD. [Электронный ресурс]. – Режим доступа: https://www.bricsys.com/ru-ru/ (дата обращения: 19.02.2020);
7. Мартин Фаулер. UML. Основы. Краткое руководство по стандартному языку объектного моделирования. / М. Фаулер – Изд: Символ-Плюс,2011, с.192 (3-е издание);
8. Функциональное тестирование [Электронный ресурс]. – URL: https://daglab.ru/funkcionalnoe-testirovanie-programmnogo-obespechenija/ (дата обращения: 13.04.2020);
9. Модульное тестирование [Электронный ресурс]. – URL: http://espressocode.top/unit-testing-software-testing/ (дата обращения: 13.04.2020);
10. Нагрузочное тестирование [Электронный ресурс]. – URL: https://daglab.ru/nagruzochnoe-testirovanie-programmnogo-obespechenija/ (дата обращения: 13.04.2020).

# Приложение А

(Справочное)

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

|  |  |
| --- | --- |
| Наименование | Описание |
| +public void PlaneParameterConstructor\_NegativeTest (double diameter, double distance, double height, double length, int numberOfHoles, double thickness, double width) | Метод для создания объекта PlaneParameters с некорректными значениями |
| + public void PlaneParameterConstructor\_PositiveTest (double diameter, double distance, double height, double length, int numberOfHoles, double thickness, double width) | Метод для создания объекта PlaneParameters с корректными значениями |

Таблица А.1 – Описание полей и методов класса ModelTests