

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра компьютерных систем в управлении и проектировании (КСУП)

РАЗРАБОТКА ПЛАГИНА «ИНЖЕКТОРНАЯ ПРОКЛАДКА» ДЛЯ СИСТЕМЫ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ «AUTODESK
INVENTOR»

Пояснительная записка к курсовому проекту по дисциплине
«Основы разработки САПР» (ОРСАПР)

Студент гр. 582-2

_____ Манаков В.В.

« ____ » _____ 2016 г.

Руководитель

м.н.с. ЛИКС каф. КСУП

_____ Калентьев А.А.

« ____ » _____ 2016 г.

Томск 2016

Реферат

Курсовой проект 36 с., 23 рис., 7 источников, 1 таблица.

INVENTOR, САПР, VISUAL STUDIO, ПЛАГИН, C#, INVENTOR API, ReSharper, NUnit.

Целью данной работы является разработка модуля для создания модели инжекторной прокладки в среде автоматизированного проектирования Autodesk Inventor 2016.

Результатом работы является модуль, осуществляющий построение модели прокладки, по заданным параметрам, с максимально возможной детализацией и реальными материалами.

Реализация программы выполнена с помощью Visual Studio 2015 Community. Редактирование исходного кода осуществлялась с помощью Visual Studio.

Пояснительная записка выполнена в Microsoft Word 2013.

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

Утверждаю
Зав. кафедрой КСУП

Шурыгин Ю.А.
« ____ » _____ 2016 г.

РАЗРАБОТКА ПЛАГИНА «ИНЖЕКТОРНАЯ ПРОКЛАДКА» ДЛЯ СИСТЕМЫ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ «AUTODESK
INVENTOR»

задание на курсовой проект по дисциплине
«Основы разработки САПР» (ОРСАПР)

Студент гр. 582-2

Манаков В.В.
« ____ » _____ 2016 г.

Руководитель

м.н.с. ЛИКС каф. КСУП

Калентьев А.А.
« ____ » _____ 2016 г.

Томск 2016

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра компьютерных систем в управлении и проектировании (КСУП)

Утверждаю
Зав. кафедрой КСУП

_____ Шурыгин Ю.А.
" ____ " _____ 2016 г.

ЗАДАНИЕ

на курсовой проект по дисциплине «ОРСАПР»

Студенту гр. 582-2 _____ Манакову Владиславу Владимировичу.

Тема: Разработка плагина «Инжекторная прокладка» для САПР «AutoDesk Inventor».

Срок сдачи студентом проекта: « 28 » _____ декабря _____ 2015 г.

Исходные данные к проекту

Разработать плагин для автоматического создания трёхмерных моделей различных типов прокладок для инжектора по заданным параметрам в системе автоматизированного моделирования «Autodesk Inventor».

Требования к программе

- модуль должен представлять собой библиотеку среды «Autodesk Inventor»;
- иметь графический интерфейс пользователя;
- иметь систему для контроля правильности ввода данных;
- при реализации системы использовать язык программирования высокого уровня C# (версия .NET Framework 4.0 и выше), в качестве среды разработки Visual Studio 2015;

- должна поддерживаться средой «Autodesk Inventor» начиная с 15-ой версии.
- 3D модель инжекторной прокладки (см. рисунок 1);

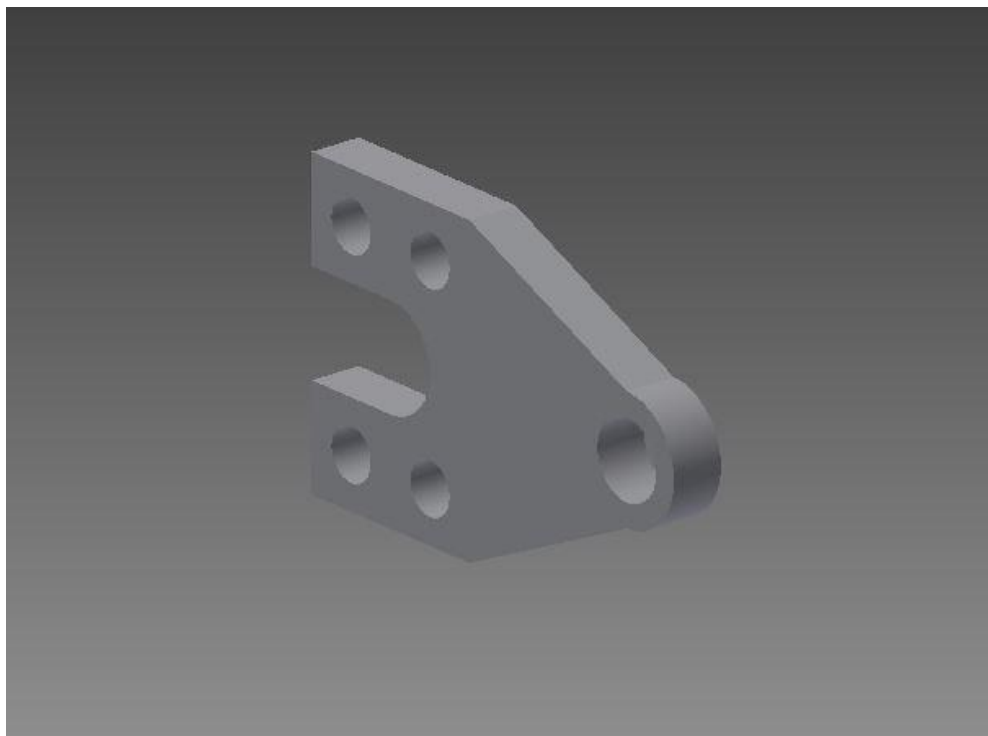


Рисунок 1 – 3D модель инжекторной прокладки

- параметры инжекторной прокладки (см. рисунок 2):
 $R1$ – радиус отверстия, находящийся на носу детали ($3:2$ к r и $2:3$ к $R2$);
 $R2$ – радиус выреза на корме детали ($3:2$ к $R1$ и $1:2$ к r);
 r – радиус отверстий по краям детали ($2:1$ к $R2$ и $3:2$ к $R1$);
 d – ширина детали;
 a – длина детали;
 h – высота детали;
 u – расстояние между центрами вырезов (не меньше r);
 t – расстояние от центра $R1$ до края детали. ($5:4$ к $R1$);
 f – расстояние от центра вырезов r до края детали;
 g – расстояние от центра выреза $R2$ до края детали;
 j – расстояние от начала детали до начала среза;
 p – расстояние ушка около выреза;
 k – расстояние от начала кормы детали до центра отверстия $R1$ на носу.

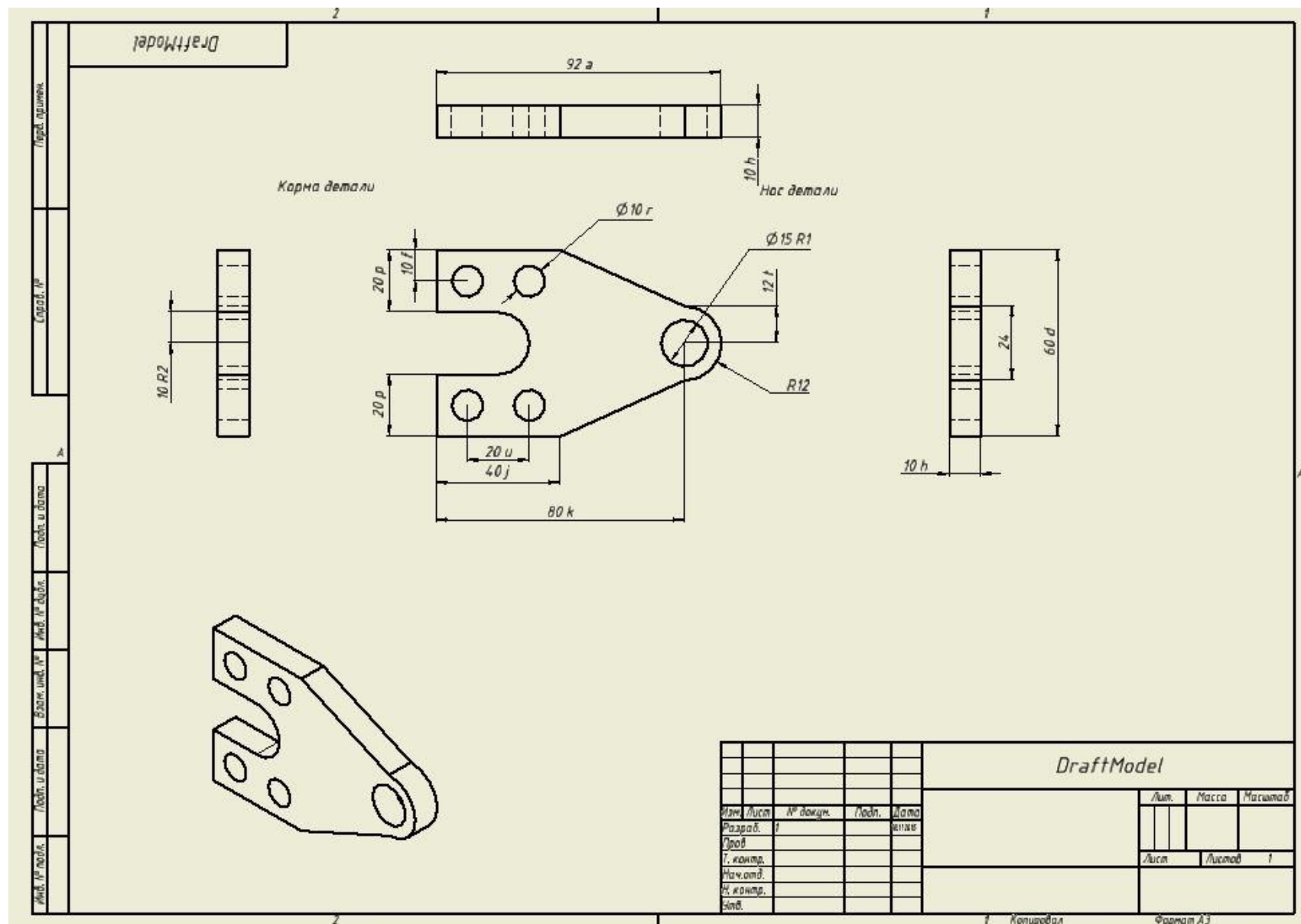


Рисунок 2 – Чертеж инжекторной прокладки

Системные и аппаратные требования к продукту

- операционная система: MS Windows 8.1 (x32), 10 (x64);
- аппаратная часть: оперативная память от 2Гб, Intel® Pentium® 4, 2 GHz или мощнее, Intel® Xeon®, Intel® Core™, AMD Athlon™ 64, или AMD Opteron™ процессор, или новее, Microsoft® Direct3D 10® или Direct3D 9® совместимые видеокарты.

График выполнения работ

Работы по созданию системы выполняются в три этапа:

- проектирование системы (14.10.2015);
- реализация проекта, прототип системы (21.12.2015);
- исправление недочетов, составление проектной документации (28.12.2015).

Содержание пояснительной записки

- реферат;
- оглавление;
- введение;
- обзор предметной области;
- обзор API;
- проект системы (UML диаграммы);
- тестирование программы;
- заключение;
- список литературы.

Дата выдачи задания: « 15 » _____ сентября _____ 2015 г.

Руководитель

м.н.с. ЛИКС, каф. КСУП

Калентьев А.А. _____

студент гр. 582-2

Манаков В.В. _____

Задание принял к исполнению

СОДЕРЖАНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ | 9 |
| 1 ОПИСАНИЕ САПР | 10 |
| 1.1 Описание API..... | 11 |
| 1.2 Описание аналогов разрабатываемого продукта..... | 14 |
| 2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ..... | 16 |
| 3 ПРОЕКТ СИСТЕМЫ | 17 |
| 3.1 Диаграмма вариантов использования..... | 17 |
| 3.2 Диаграмма классов | 18 |
| 3.3 Макеты пользовательского интерфейса | 21 |
| 4 РЕАЛИЗАЦИЯ СИСТЕМЫ | 22 |
| 4.1 Пользовательский интерфейс. | 22 |
| 4.2 Диаграмма классов..... | 23 |
| 5 ТЕСТИРОВАНИЕ | 25 |
| 5.1 Функциональное тестирование..... | 25 |
| 5.2 Модульное тестирование | 30 |
| 5.3 Нагрузочное тестирование..... | 32 |
| ЗАКЛЮЧЕНИЕ | 35 |
| СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ | 36 |

ВВЕДЕНИЕ

При проектировании определенных объектов, возникают такие ситуации, при которых имеющегося функционала, используемого программного обеспечения, не достаточно, тогда приходится прибегать к не тривиальным решениям данной проблемы.

Решением может послужить создание специализированного под конкретную задачу плагина[1], (модуля). Удобность применения плагина обусловлена тем, что плагин независимо компилируемый программный модуль, динамически подключаемый к основной программе, который можно осуществить при помощи технологии Application Programming Interface (API) [2]. API-технология предоставляет программисту набор процедур и функций для управления основной программой, но не дает прямого доступа к свойствам и методам объектов внутри основной программы.

Курсовой проект ориентирован на создание подсистемы системы автоматизированного проектирования (САПР) [3], в данном случае реализацией подсистемы будет плагин, который будет выполнять функции и задачи подсистемы на базе системы Autodesk Inventor[4].

1 ОПИСАНИЕ САПР

Autodesk Inventor — система трёхмерного твердотельного и поверхностного параметрического проектирования (САПР) компании Autodesk, предназначенная для создания цифровых прототипов промышленных изделий. Инструменты Inventor обеспечивают полный цикл проектирования и создания конструкторской документации[5]:

- 2D-/3D-моделирование;
- создание изделий из листового материала и получение их разверток;
- разработка электрических и трубопроводных систем;
- проектирование оснастки для литья пластмассовых изделий;
- динамическое моделирование;
- параметрический расчет напряженно-деформированного состояния деталей и сборок;
- визуализация изделий;
- автоматическое получение и обновление конструкторской документации (оформление по ЕСКД).

Предлагаемый функционал:

- Проектирование и тестирование в цифровом формате;
- прямое моделирование при построении деталей;
- динамический ввод данных при отрисовке эскизов;
- команда Сборка;
- проектирование пресс-форм;
- более эффективная совместная работа.

1.1 Описание API

Библиотека – это программный модуль, приложение, созданное для расширения стандартных возможностей системы Autodesk Inventor. Библиотека представляет собой ориентированную на конкретную задачу подсистему автоматизированного проектирования, которая после выполнения проектных расчетов формирует готовые конструкторские документы или их комплекты.

Доступ к внутренним функциям Autodesk Inventor обеспечивается двумя путями:

- через экспортные функции, оформленные в виде dll - модулей, которые разработчик подключает к своей программе, – при создании плоских чертежей; через использование COM-объектов – при программном формировании твердотельных моделей;
- с помощью технологии автоматизации, реализованной через API (Application Programming Interface – программный интерфейс приложения) системы Autodesk Inventor. Autodesk Inventor включает в свой состав 2D API и 3D API. 3D API обеспечивает доступ к системе Autodesk Inventor для создания и редактирования трехмерных моделей.

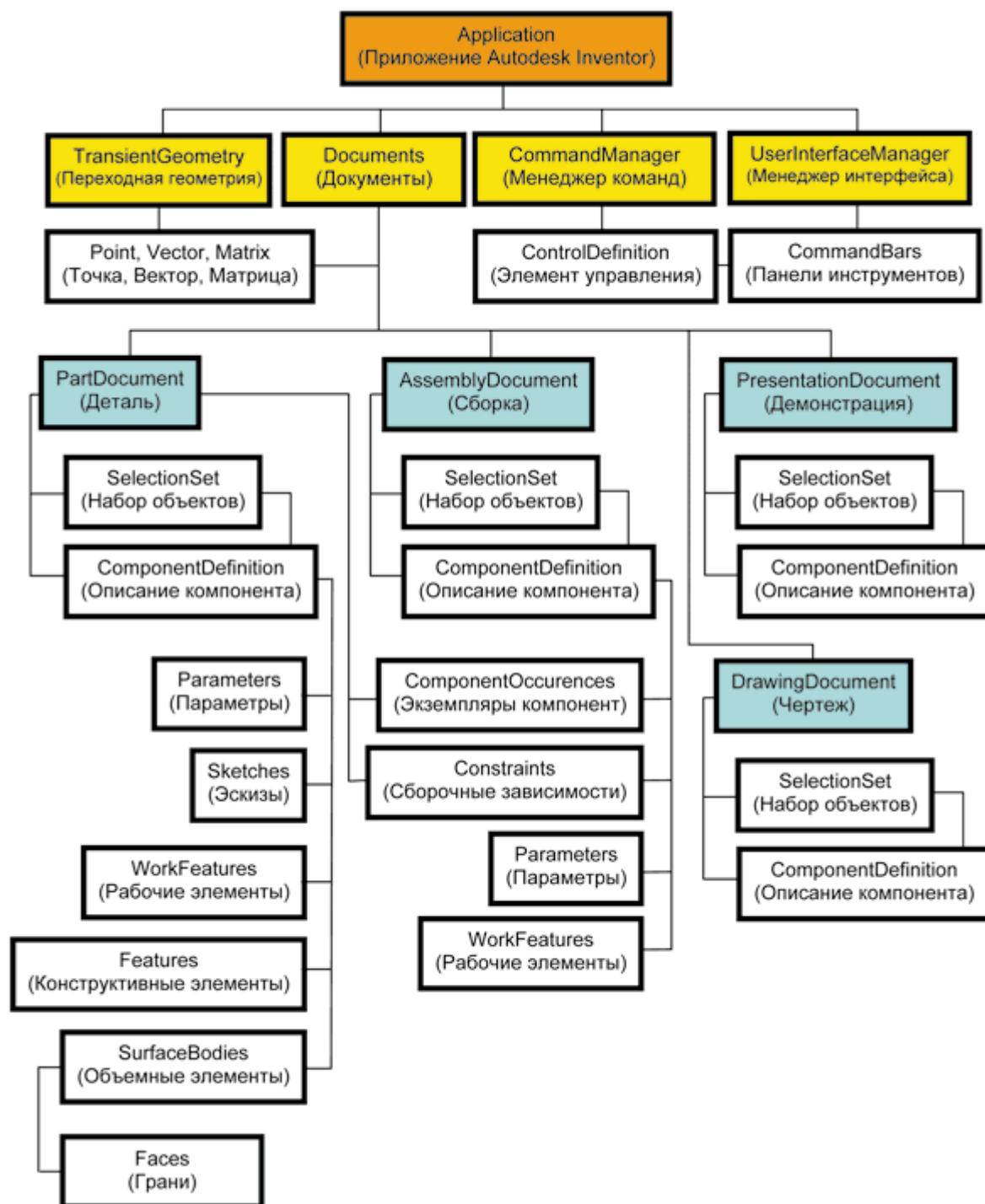


Рисунок 1.1 – Объектная модель Autodesk Inventor(AI)

Проект приложения AI (Application) представляет собой систему специальных документов САПР (Documents), переходной геометрии (TransientGeometry), менеджера команд (CommandManager) и менеджера пользовательского интерфейса (UserInterfaceManager). Основные типы документов AI: сборки (AssemblyDocument), деталь (PartDocument), демонстрация (PresentationDocument), чертеж (DrawingDocument).

AI — среда ассоциативного проектирования, поэтому каждый из типов документов, входящих в проект на всем протяжении разработки, сохраняет взаимосвязь с другими документами, в которых он используется.

Так, изменения в файле детали отражаются в документе сборки, содержащей эту деталь, или файле чертежа, имеющем виды этой детали.

Каждый из типов документов позволяет программно формировать и работать с наборами составляющих объектов (SelectionSet) и отдельными компонентами (ComponentDefinition). Для отдельного компонента доступны все характеризующие его объекты и свойства. Например, описание компонента (ComponentDefinition) детали (PartDocument) включает эскизы (плоские и трехмерные контуры), объемные элементы (вращения, выдавливания, сдвига, оболочки и др.) с опорными эскизами, конструктивные элементы (фаски, сопряжения, резьбы и др.), рабочие элементы (вспомогательные плоскости, оси, точки), таблицы параметров, управляющих геометрией. Описание компонента (ComponentDefinition) документа сборки (AssemblyDocument) в качестве экземпляра компонента (ComponentOccurrences) может включать документы деталей (PartDocument).

Элементы переходной геометрии (TransientGeometry) — это объекты (точки, векторы, матрицы), не имеющие графического выражения в моделях, но используемые при создании геометрии в документах AI.

Отдельно следует сказать об объектах типа сборочных зависимостей (Constraints), с помощью которых в документе сборки между деталями устанавливаются взаимные ограничения позиционирования и движения, что позволяет моделировать в сборке функциональность объекта-прототипа. Сборочные зависимости (Constraints) ассоциативно доступны как из объектов деталей, так и из применяющих их сборок.

1.2 Описание аналогов разрабатываемого продукта

Существует похожая сборка, по семантике аналогична проектированию инжекторной прокладки. Рассмотрим «Корпус редуктора лебедки подъема»[6] (рисунок 1.2).

Все выполняется в режиме сборки.

Изначально проектируется крышка редуктора лебедки, далее идёт сборка корпуса редуктора. После, отталкиваясь от заданных параметров крышки и корпуса редуктора лебедки (рисунок 1.3), проектируется прокладка, которая служит для более плотного прилегания крышки к лебедке и уменьшает время износа данной крышки (рисунок 1.4).

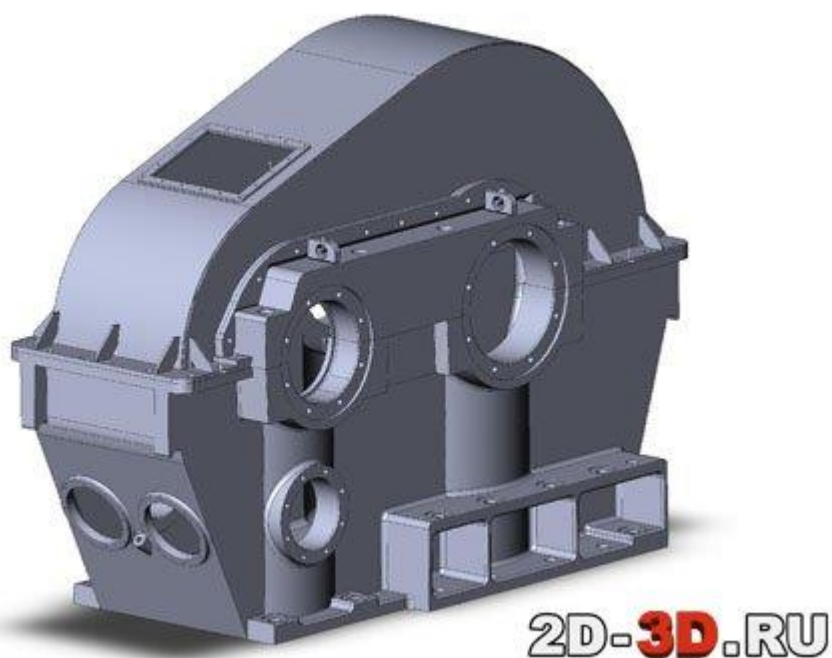


Рисунок 1.2 - «Корпус редуктора лебедки подъема»



Рисунок 1.3 – Крышка и корпус редукторной лебедки

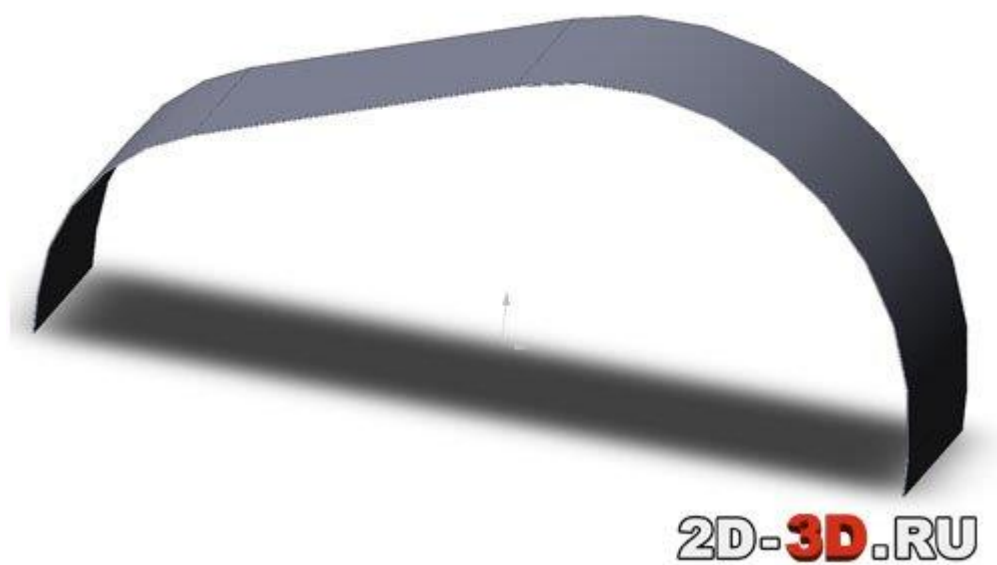


Рисунок 1.4 – Прокладка крышки редуктора лебедки подъема

2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ

Двигатель автомобиля – сложный механизм, состоящий из множества деталей. Внутри двигателя при работе непрерывной циркулируют различные технические жидкости, часть которых находится под большим давлением. Для поддержания давления на постоянном уровне и предупреждения выхода жидкостей наружу корпус двигателя должен быть герметичен. Для обеспечения герметичности служат прокладки разной формы, сделанные из различных материалов.

Инжекторная прокладка – это прокладка для инжекторного двигателя автомобиля, которая состоит из резиновой смеси, которую прессуют с добавлением асбестового волокна и порошка графита. Она служит для уплотнения мест прилегания деталей и противодействия внутреннему давлению в сочетании с высокой температурой.

Параметры инжекторной прокладки:

- a) радиус отверстия, находящийся на носу детали (не менее 2мм);
- b) радиус выреза на корме детали (не менее 5мм);
- c) радиус отверстий по краям детали (не менее 2мм);
- d) ширина (должна быть не менее 55мм);
- e) длина (должна быть не менее 80мм);
- f) высота (должна быть не менее 1мм и не более 25мм);
- g) расстояние между центрами вырезов (не меньше радиуса выреза);
- h) расстояние от центра вырезов (не меньше радиуса выреза);
- i) расстояние от начала детали до начала среза (не менее 35мм);
- j) расстояние ушка около выреза (не больше радиуса на корме выреза).

3 ПРОЕКТ СИСТЕМЫ

Для описания архитектуры, пользовательского сценария системы был выбран унифицированный язык моделирования (UML)[7]. На основе UML построены: диаграммы вариантов использования[7] и диаграммы классов[7].

Достаточно важным моментом при проектировании программы, является внешний вид программы. Внешний вид программы должен максимально отражать функциональные особенности программы, чтобы пользователь за минимальное время мог освоить программу и воспользоваться ей, подробное описание макетов пользовательского интерфейса и сами макеты приведены в подразделе 3.4.

3.1 Диаграмма вариантов использования

Диаграмма вариантов использования отражает возможный выбор действий (выбора состояния) пользователя внутри системы (Рисунок 3.1).

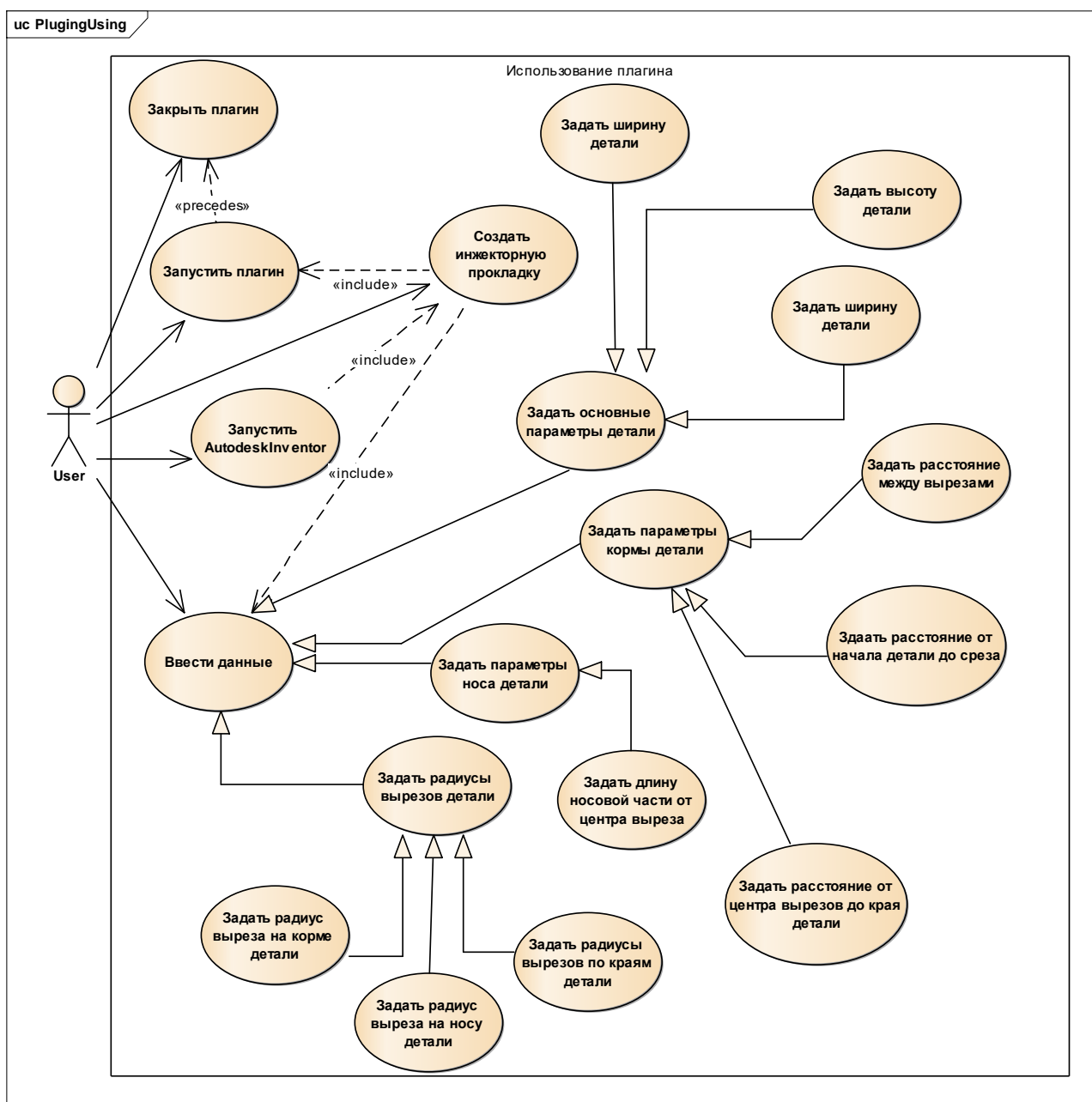


Рисунок 3.1— Диаграмма вариантов использования

3.2 Диаграмма классов

Предмет проектирования состоит из множества индивидуальных особенностей, которые определяют его в целом, поэтому удобно проектировать систему снизу-вверх, исходя из этого был выбран восходящий метод проектирования.

Для реализации подсистемы были спроектированы следующие классы (рисунок 3.2):

- `GasketUserInterface` – диалоговый класс, обеспечивающий взаимодействие между пользователем и программой через форму, подробное описание в Приложении А (Таблица А.1);
- `GasketInventorApi` – класс, обеспечивающий взаимодействие с методами Inventor API, подробное описание в Приложении А (Таблица А.2);
- `GasketProperties` – класс, хранящий в себе все параметры модели инжекторной прокладки, осуществляет проверку зависимых параметров, подробное описание в Приложении А (Таблица А.3);
- `PropertyValue` – класс предназначен для хранения и обработки параметров модели, подробное описание в Приложении А (Таблица А.4);
- `GasketModelCreator` – класс построения модели инжекторной прокладки в системе Autodesk Inventor, подробное описание в Приложении А (Таблица А.5);
- `GasketPropertiesType` – перечисление всех параметров модели инжекторной прокладки, подробное описание в Приложении А (Таблица А.6).

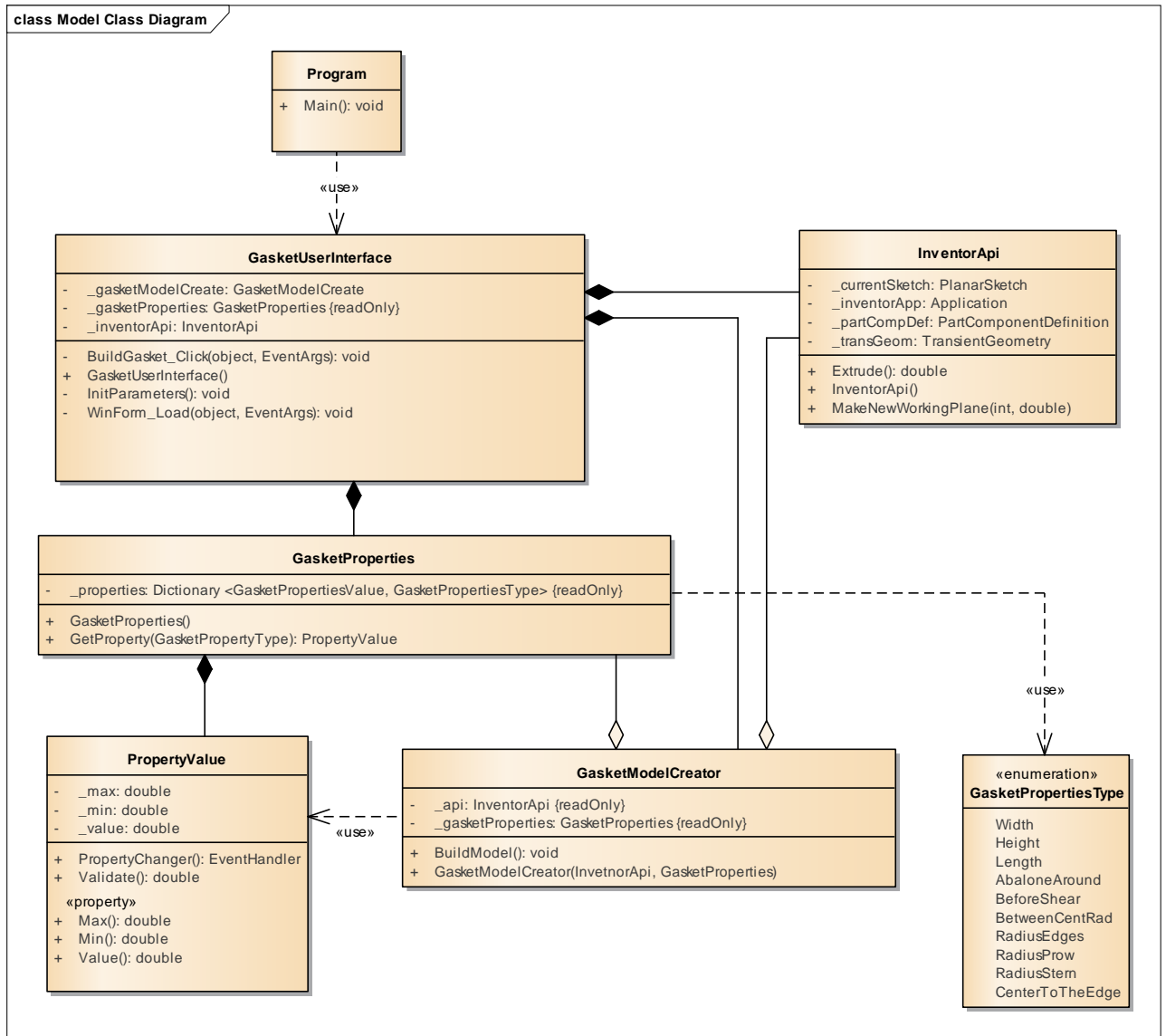


Рисунок 3.2 – UML диаграмма классов

3.3 Макеты пользовательского интерфейса

В текущем проекте будет использоваться Windows Forms. Ниже представлен макет графического интерфейса для ввода параметров модели. Программа будет состоять из одного диалогового окна с десятью пунктами и двумя кнопками «Запустить Inventor», «Построить деталь» (Рисунок 3.4).

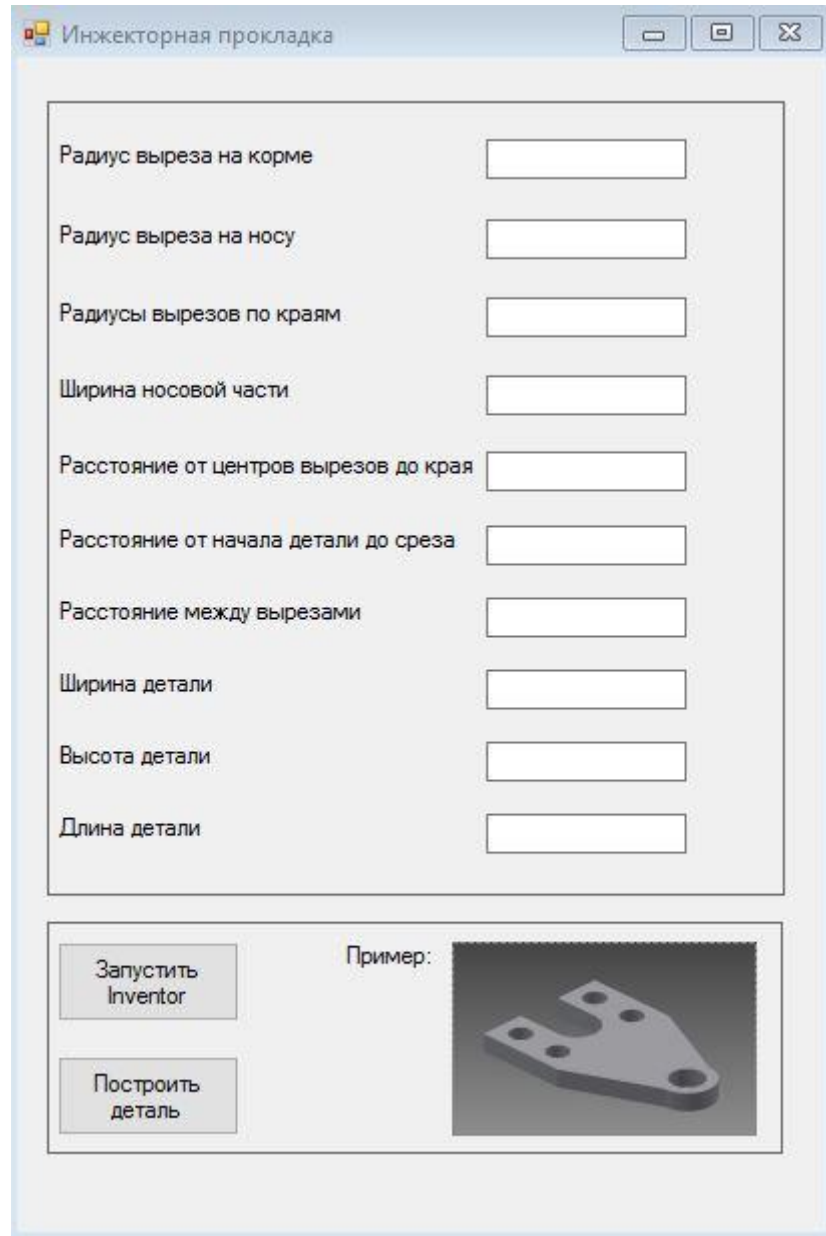


Рисунок 3.4 – Диалоговое окно при старте

Изначально при запуске плагина в параметрах уже будут содержаться значения, по которым можно построить модель.

4 РЕАЛИЗАЦИЯ СИСТЕМЫ

В процессе реализации спроектированная ранее система была неоднократно изменена и дополнена, улучшилась функциональность плагина, расширились возможности построения модели инжекторной прокладки, а также был сделан акцент на практичности выполненного модуля.

4.1 Пользовательский интерфейс.

Самым первым изменением претерпел пользовательский интерфейс программы. Были добавлены два новых поля с параметрами («Расстояние от начала, до центра выреза на носу детали»; «Глубина выреза на корме детали»), которые были не обходимы в процессе построения детали, а также поля заполнения параметров были заменены. Изначально параметры передавались через TextBox, но в целях удобства и практичности данные поля были заменены на UserControl, через который также было удобнее выполнять проверку данных.

После этого был добавлен ComboBox для расширения функциональности построения детали, в котором предоставляется выбор формы вырезов по краям детали (круг, прямоугольник, треугольник).

Также было добавлено автоматическое заполнение полей параметров при запуске плагина. Поля заполняются стандартными размерами инжекторной прокладки, которые подходят для большинства инжекторных двигателей.

Внесенные в плагин изменения пользовательского интерфейса можно увидеть на рисунке 4.1.

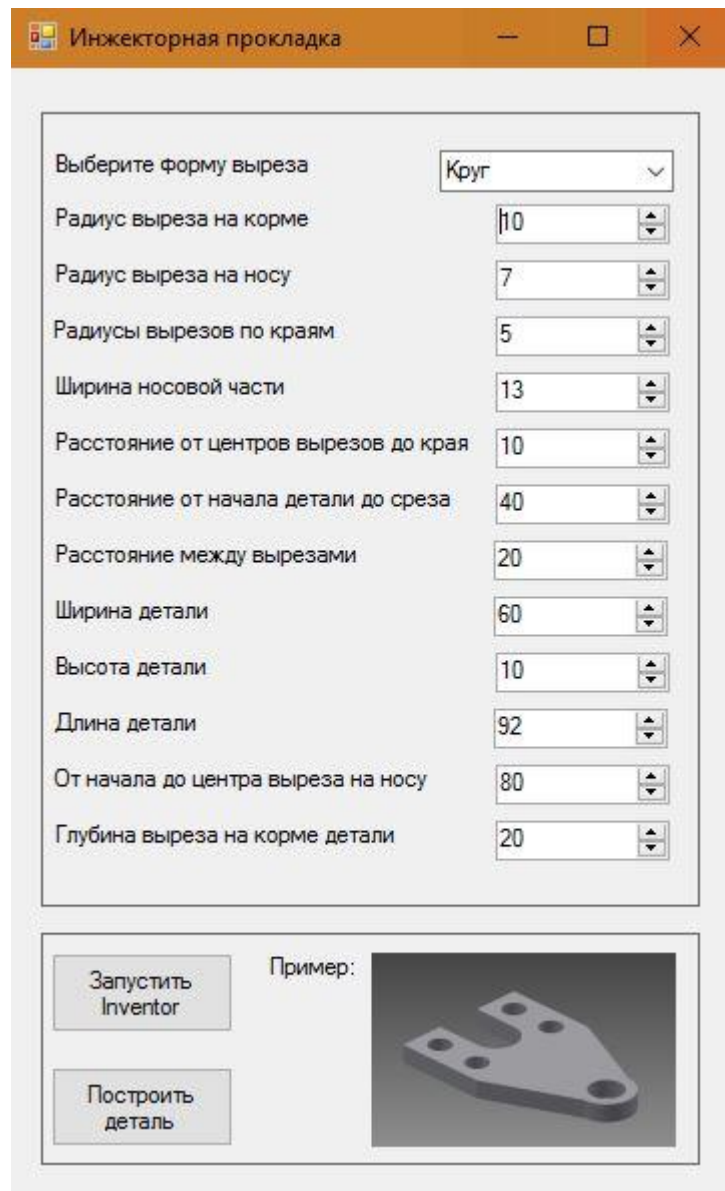


Рисунок 4.1 – Пользовательский интерфейс.

4.2 Диаграмма классов

Учитывая перечисленные выше изменения пользовательского интерфейса, диаграмма классов, описывающая архитектуру программы, также изменилась.

Был добавлен класс, описывающий поведение UserControl, дополнена проверка корректности данных, расширен набор методов построения детали.

Финальная версия диаграммы классов приведена на рисунке 4.2

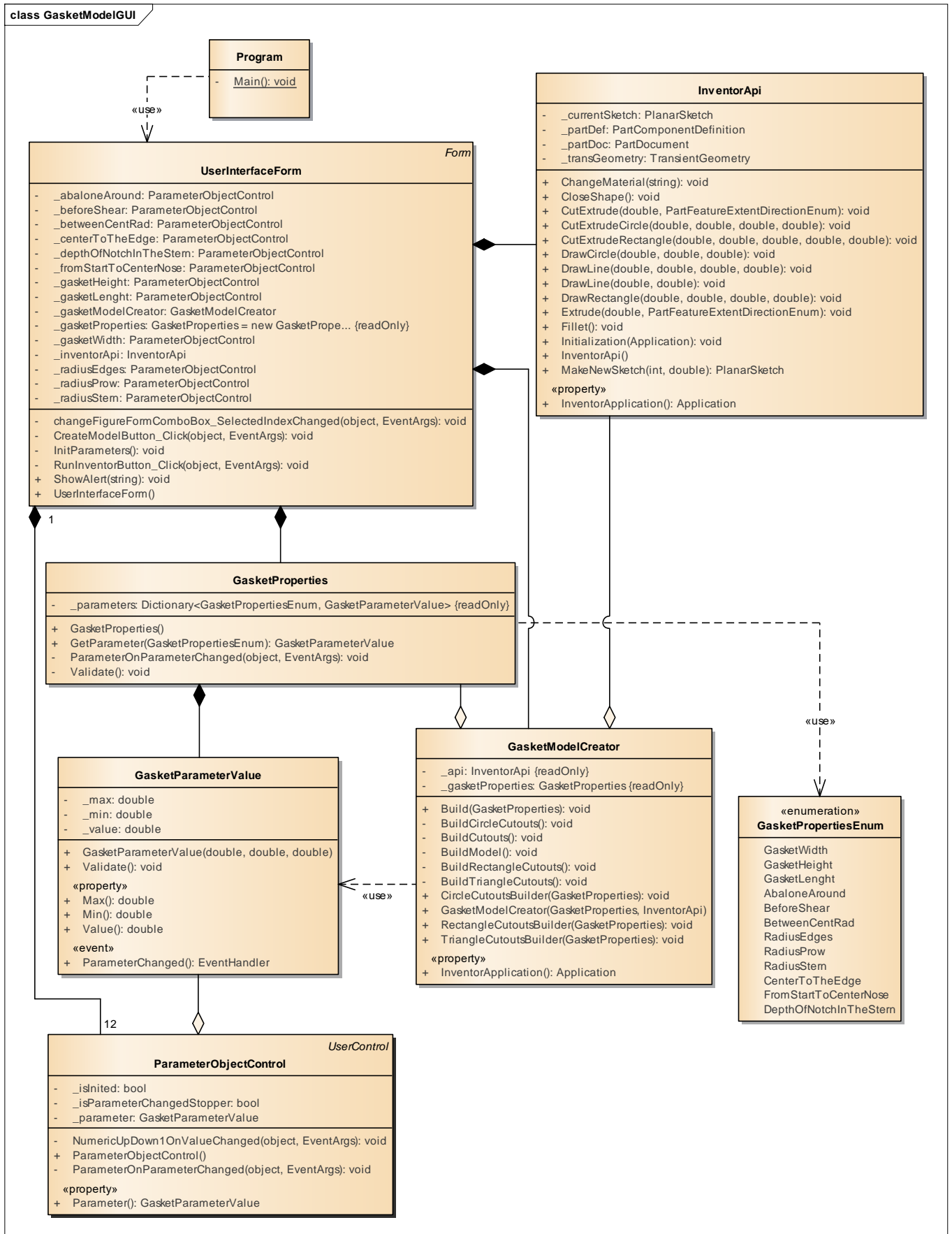


Рисунок 4.2 – Диаграмма классов.

5 ТЕСТИРОВАНИЕ

Тестирование — это проверка соответствия объекта желаемым критериям. Несоответствие объекта желаемым критериям называется ошибкой[7].

Существует много классификаций тестов по разным признакам. Тут же стоит отметить, что данные классификации пересекаются и один тест может одновременно относиться к нескольким классификациям[7].

В данном проекте будет применяться три вида тестирования: функциональное тестирование, модульное тестирование, нагрузочное тестирование.

5.1 Функциональное тестирование

Функциональное тестирование — тестирование функциональности объекта, т. е. правильно ли объект выполняет свои функции. Фактически, выполняется проверка правильности выходных данных при соответствующих входных[7]. В данном случае будет проверяться правильность построения детали при различных входных параметрах:

- Проверим построение модели инжекторной прокладки по заготовленным наборам параметров (см. рисунки 5.1-5.2).
- Проверим корректность построения детали при минимальных значениях параметров и при выборе формы вырезов по краям детали в виде прямоугольников (см. рисунки 5.3-5.4).
- Проверим корректность построения модели при максимальных значениях параметров и при выборе формы вырезов по краям детали в виде треугольников (см. рисунки 5.3-5.4).

Инжекторная прокладка

Выберите форму выреза: Круг

| | |
|---------------------------------------|----|
| Радиус выреза на корме | 10 |
| Радиус выреза на носу | 7 |
| Радиусы вырезов по краям | 5 |
| Ширина носовой части | 13 |
| Расстояние от центров вырезов до края | 10 |
| Расстояние от начала детали до среза | 40 |
| Расстояние между вырезами | 20 |
| Ширина детали | 60 |
| Высота детали | 10 |
| Длина детали | 92 |
| От начала до центра выреза на носу | 80 |
| Глубина выреза на корме детали | 20 |

Запустить Inventor

Построить деталь

Пример:




Рисунок 5.1 – Пользовательский интерфейс со стандартным набором параметров

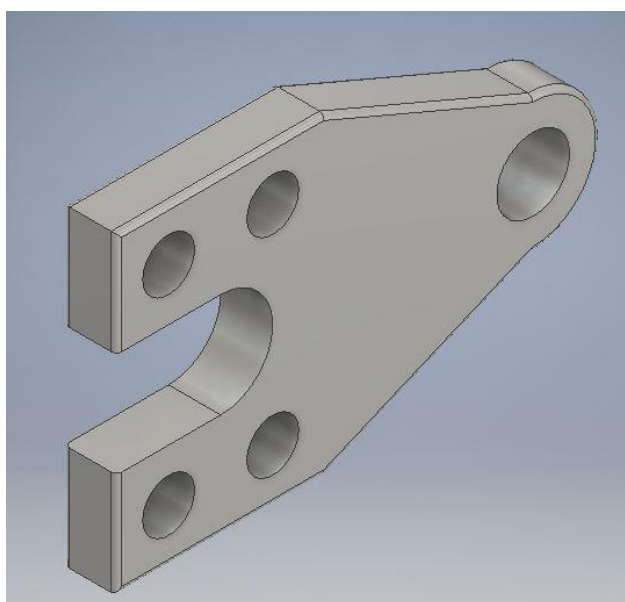


Рисунок 5.2 – Спроектированная модель со стандартным набором параметров

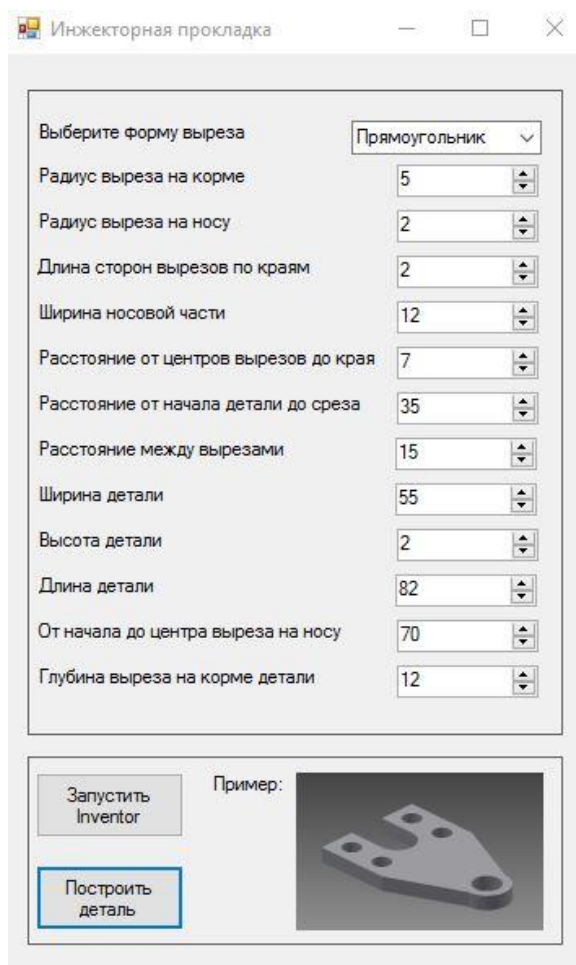


Рисунок 5.3 – Построение детали при минимальных значениях параметров и прямоугольными вырезами по краям

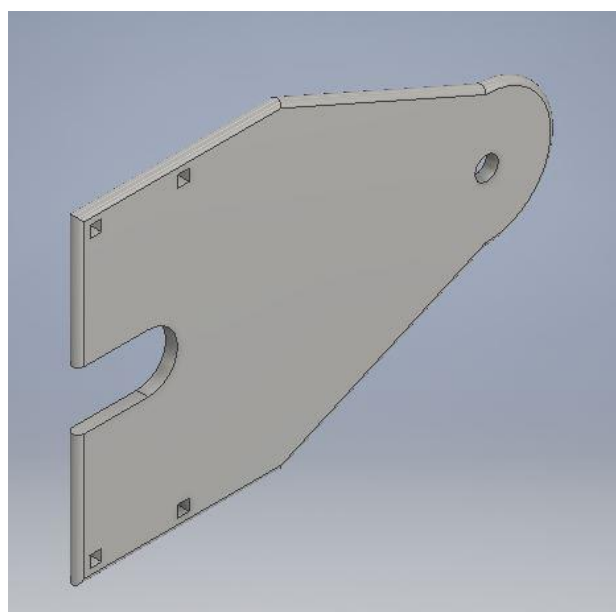


Рисунок 5.4 – Спроектированная модель при минимальных значениях параметров и прямоугольными вырезами по краям

Инжекторная прокладка

| | |
|---------------------------------------|-------------|
| Выберите форму выреза | Треугольник |
| Радиус выреза на корме | 12 |
| Радиус выреза на носу | 11 |
| Длина сторон вырезов по краям | 7 |
| Ширина носовой части | 20 |
| Расстояние от центров вырезов до края | 12 |
| Расстояние от начала детали до среза | 50 |
| Расстояние между вырезами | 30 |
| Ширина детали | 80 |
| Высота детали | 25 |
| Длина детали | 110 |
| От начала до центра выреза на носу | 90 |
| Глубина выреза на корме детали | 35 |

Запустить Inventor
 Построить деталь

Пример:

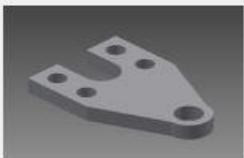


Рисунок 5.5 – Построение детали при максимальных значениях параметров и треугольными вырезами по краям

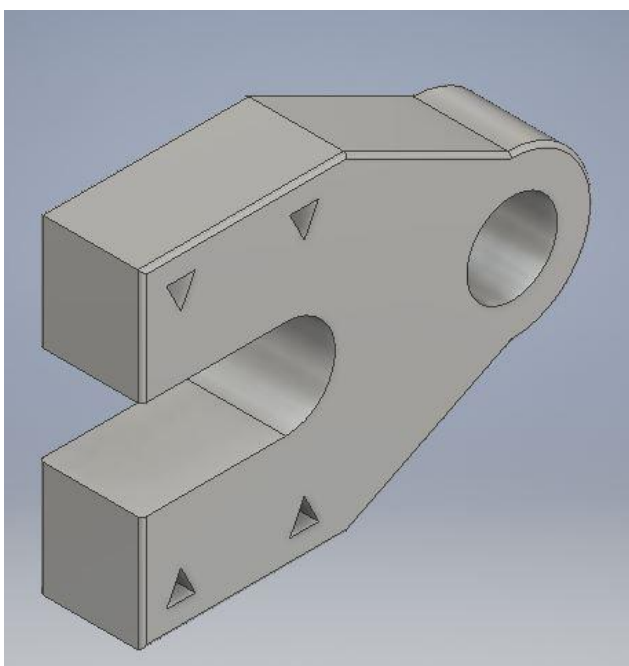


Рисунок 5.6 – Спроектированная модель при максимальных значениях параметров и треугольными вырезами по краям

Как видно из проделанных функциональных тестов, деталь, при различных итерациях входных параметров, строится корректно и соответствует необходимым требованиям.

Далее было выполнено тестирование плагина в операционной системе Windows 8.1 x86, для выполнения плагина использовался Visual Studio 2015 Community, построение детали выполнялось в Inventor 2015 (рисунок 5.7).

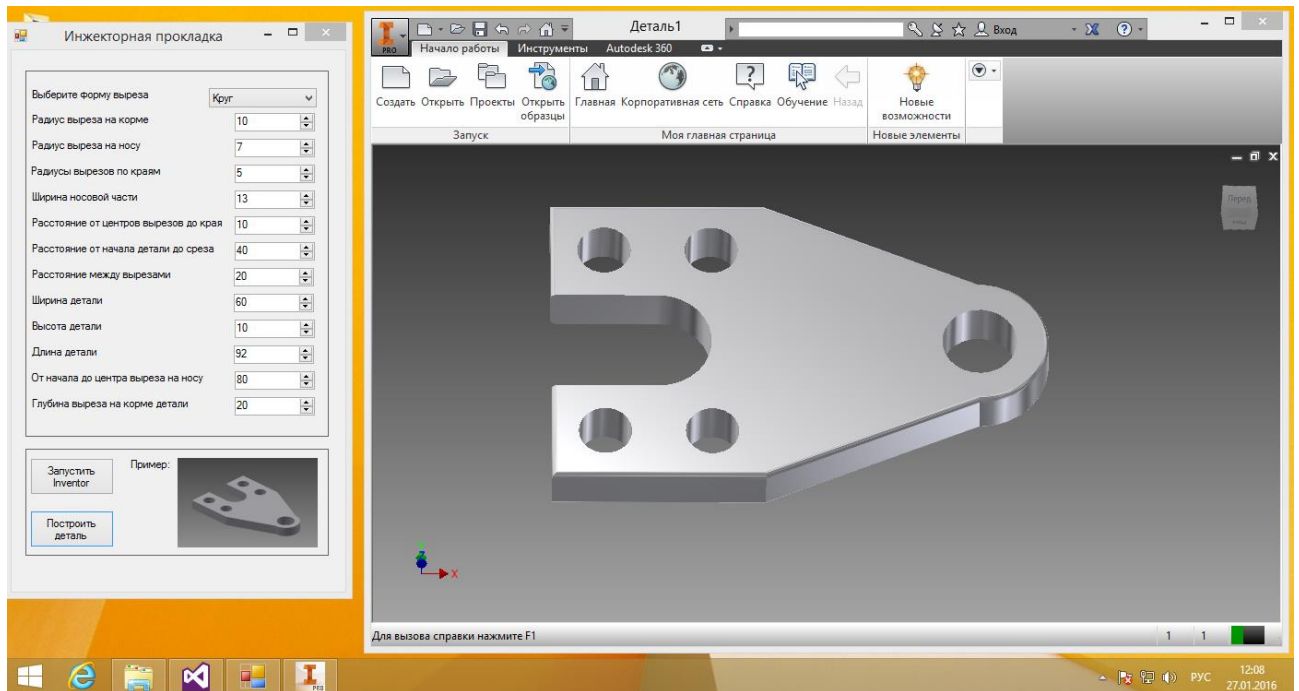


Рисунок 5.7 – Построение детали в Windows 8.1.

Исходя из проделанных выше тестов можно сделать вывод, что реализованный модуль корректно запускается и выполняет построение независимо от версии Windows и Inventor.

5.2 Модульное тестирование

Юнит-тестирование (англ. «*unit-testing*», или модульное тестирование) — тестирование отдельного элемента изолированно от остальной системы. Относительно парадигмы объектно-ориентированного программирования системой является вся программа, а отдельным элементом — класс или его метод. Юнит-тестирование предназначено для проверки правильности работы отдельно взятого элемента. Чтобы исключить из результатов тестирования влияние потенциальных ошибок других элементов, тестируемый элемент должен быть максимально изолирован, то есть не использовать объекты и методы других классов[7].

На основе следующих тестовых сценариев (таблица 5.1), проводилось тестирование корректности входных параметров и правильности работы плагина с Inventor 2016.

Таблица 5.1 – Тестовые сценарии

| Модуль | Название тестового метода | Тестовые сценарии |
|-----------------|--|---|
| ParametersTests | PositiveValueTest(double testValue) | При вводе корректных значений параметров, исключение не должно генерироваться. |
| | NotPositiveValueTest(double testValue) | При вводе значений, выходящих за допустимые пределы параметров, должно генерироваться исключение. |
| | ValueIsNotSaved(double testValue) | При вводе некорректного значения параметра, это значение не должно сохраняться. |

Продолжение таблицы 5.1

| Модуль | Название тестового метода | Тестовые сценарии |
|-----------------|--|--|
| ParametersTests | MaxValueTest(double testValue) | При вводе максимального значения параметра, должно генерироваться исключение т. к, параметр выходит за пределы объявления. |
| | MaxLimitTest(double testLimit1, double testLimit2) | При вводе максимального значения предела, должно генерироваться исключение т. к, лимит не проходит исходную валидацию. |
| | MinValueTest(double testValue) | При вводе минимального значения параметра, должно генерироваться исключение т. к, параметр меньше нуля и выходит за пределы. |
| | MinLimitTest(double testLimit1, double testLimit2) | При вводе минимального значения предела, должно генерироваться исключение т. к, лимит не проходит исходную валидацию. |
| InventorAPITest | void InitPositive() | При получении корректной ссылки на приложение, инициализация не должна генерировать исключение. |
| | InitNegative() | При получении некорректной ссылки на приложение, инициализация должна генерировать исключение. |

Тестирование проводилось с помощью фреймворка модульного тестирования NUnit 3.0.1 для языков платформы .NET. Выполнение тестов проводилось в дополнении ReSharper, интегрируемого в Visual Studio.

Ниже приведен рисунок успешного прохождения всех описанных модульных тестов (см. рисунок 5.8).

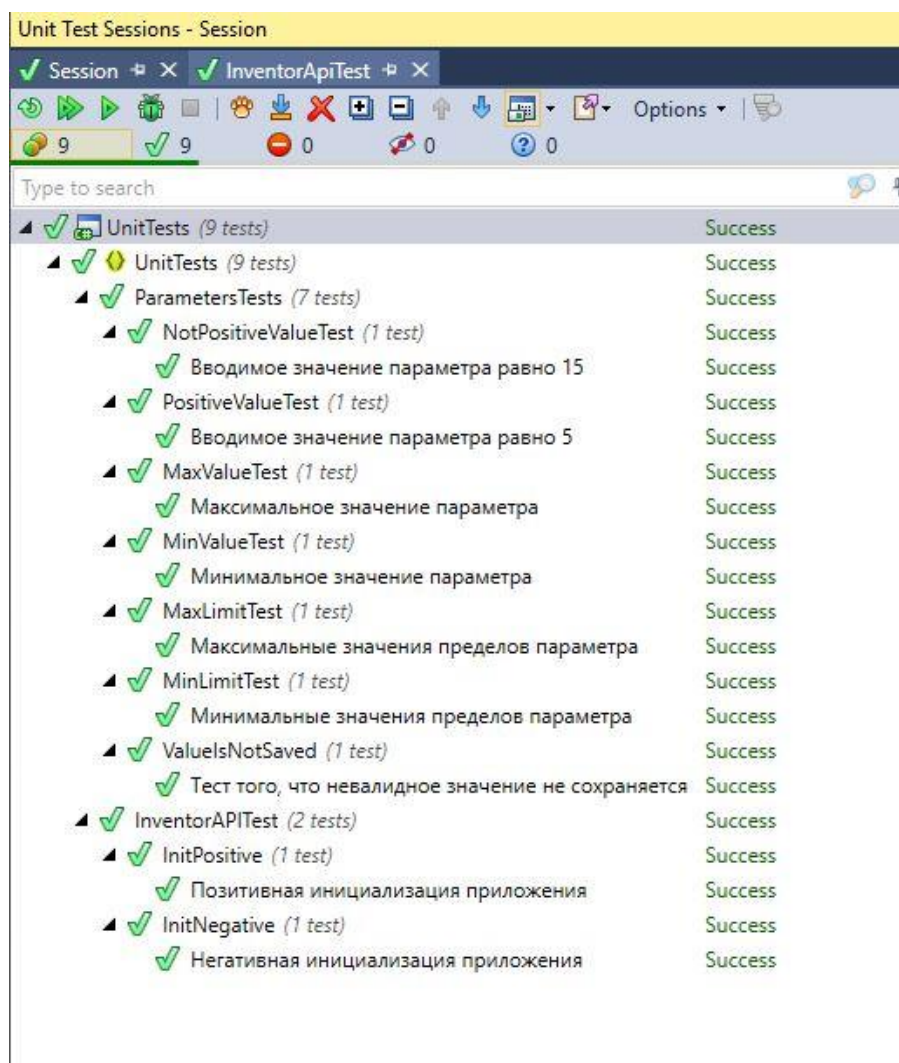


Рисунок 5.8 – Окно сессии модульного тестирования приложения.

5.3 Нагрузочное тестирование

Нагрузочное тестирование (англ. load testing) — подвид тестирования производительности, сбор показателей и определение производительности и времени отклика программно-технической системы или устройства в ответ на внешний запрос с целью установления соответствия требованиям, предъявляемым к данной системе (устройству).

Рассмотрим зависимость скорости построения модели с настройками по умолчанию от количества одновременно открытых моделей в программе «Inventor 2016» (см. рисунок 5.9).

Выполнение нагрузочного тестирования проводилось в программе HP LoadRunner. При попытке построить 100 деталей, программа выдала ошибку о нехватке оперативной памяти. Исходя из этого, для выполнения нагрузочного тестирования, количество построенных деталей было принято равным 97.



Рисунок 5.9 – Нагрузочное тестирование

Нагрузка на оперативную память в режиме ожидания при построенных 97 деталях инжекторной прокладки со стандартными параметрами в «Inventor 2016».

| Имя | 2% ЦП | 62% Память | 1% Диск | 0% Сеть |
|----------------------------|----------|---------------|------------|------------|
| Приложения (6) | | | | |
| > Autodesk® Inventor® 2016 | 0,5% | 1 146,3 МБ | 0 МБ/с | 0 Мбит/с |

Рисунок 5.10 – Нагрузка оперативной памяти в режиме ожидания

Нагрузка на центральный процессор при построении 97 деталей инжекторной прокладки со стандартными параметрами в «Inventor 2016».

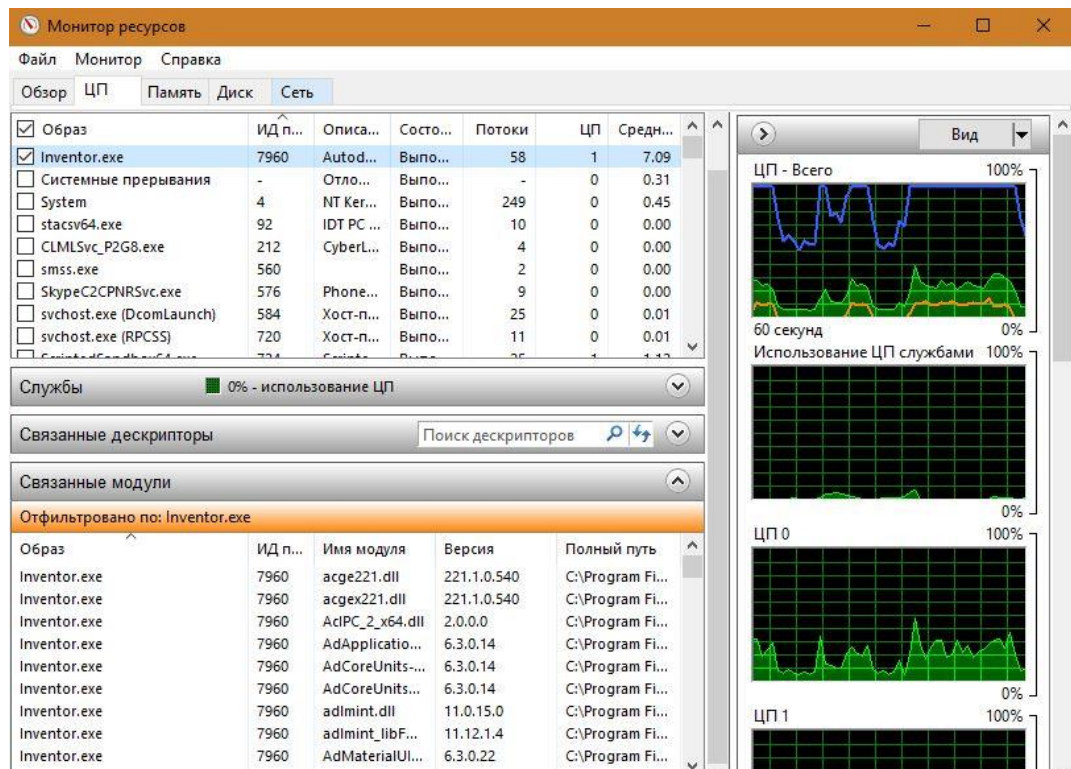


Рисунок 5.10 – Нагрузка центрального процессора

Нагрузка на оперативную память при построении 97 деталей инжекторной прокладки со стандартными параметрами в «Inventor 2016».

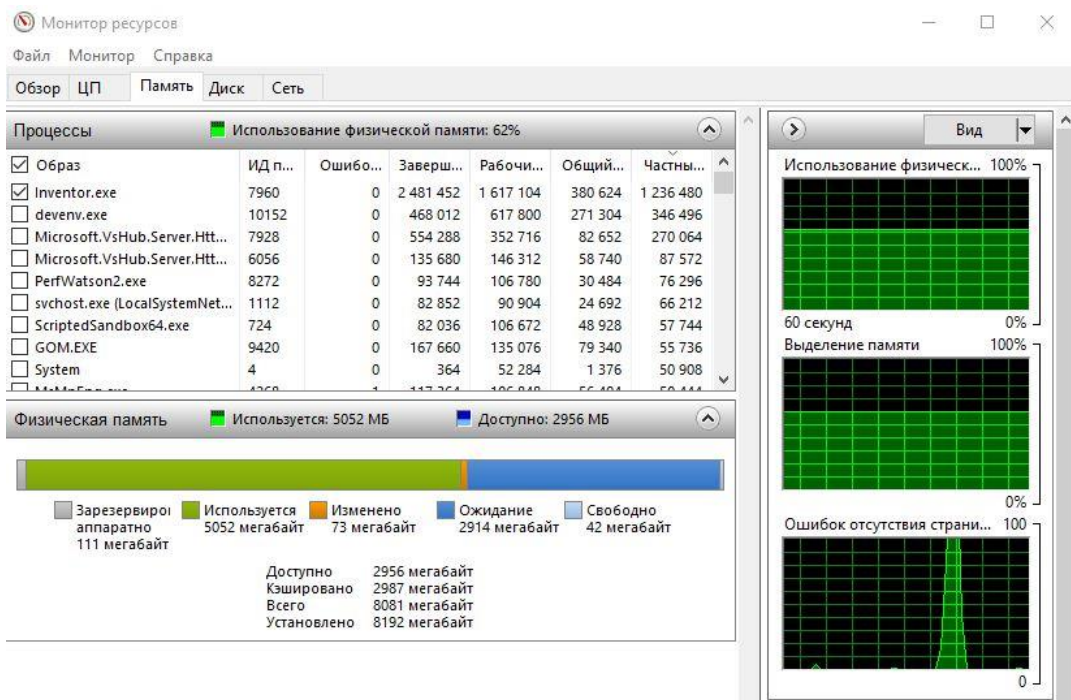


Рисунок 5.11 – Нагрузка оперативной памяти

ЗАКЛЮЧЕНИЕ

В ходе выполнения курсового проекта были изучены основные этапы проектирования программного продукта и его реализации, изучена предметная область объекта проектирования, Inventor API. Реализован модуль расширения «Построение инжекторной прокладки» для системы Inventor 2016. В модуле предусмотрено изменение параметров модели и реализована проверка корректности вводимых данных.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Плагин [Электронный ресурс]. – Режим доступа:
<http://ohostinge.com/dictionary/plug-in> Дата обращения (11.10.2015)
2. API [Электронный ресурс]. – Режим доступа:
<http://habrahabr.ru/post/201700/> Дата обращения (11.10.2015)
3. САПР [Электронный ресурс]. – Режим доступа:
http://www.tadviser.ru/index.php/Статья:САПР_Системы_автоматизированного_проектирования Дата обращения (28.01.2016)
4. Autodesk Inventor (САПР) [Электронный ресурс]. – Режим доступа:
www.autodesk.ru/products/inventor/features/all Дата обращения (11.10.2015)
5. Autodesk Inventor (САПР) [Электронный ресурс]. – Режим доступа:
<http://www.autodesk.ru/products/inventor/overview> Дата обращения (11.11.2015)
6. Корпус редуктора лебедки подъема [Электронный ресурс]. – Режим доступа:
<http://www.2d-3d.ru/3d-galereia/solidworks/2289-korpus-reduktora-lebedki-podema-ekg-8.html> Дата обращения (11.12.2015)
7. «Новые технологии в программировании» Учебно-методическое пособие / Томск. Томский государственный университет систем управления и радиоэлектроники, 176 стр.