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)

ПРОЕКТ СИСТЕМЫ

к курсовому проекту по дисциплине

«Основы разработки САПР»

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

Выполнил:

Студент гр. 582-2

Манаков В.В.

«\_\_»\_\_\_\_\_\_\_\_\_2015г.

Принял:

м.н.с ЛИКС каф. КСУП

Калентьев А.А.

«\_\_»\_\_\_\_\_\_\_\_2015г.

Томск 2015

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc432546834)

[1 ОПИСАНИЕ САПР 4](#_Toc432546835)

[1.1 Описание API 7](#_Toc432546836)

[1.2 Описание аналогов разрабатываемого продукта 8](#_Toc432546837)

[2 ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ 10](#_Toc432546838)

[3 ПРОЕКТ ПРОГРАММЫ 12](#_Toc432546839)

[3.1 Диаграмма вариантов использования 12](#_Toc432546840)

[3.2 Диаграмма классов 14](#_Toc432546841)

[3.3 Диаграмма пакетов 16](#_Toc432546842)

[3.4 Макеты пользовательского интерфейса 17](#_Toc432546843)

[ЗАКЛЮЧЕНИЕ 18](#_Toc432546844)

[Приложение A 20](#_Toc432546845)

# ВВЕДЕНИЕ

В процессе проектирования в зависимости от конкретного используемого программного продукта, могут возникнуть задачи. Когда стандартных средств и инструментов проектирования, содержащихся в данном программном продукте, для реализации конкретной (специфичной) задачи недостаточны. При возникновении таких задач, можно перейти к другому программному продукту. Который может содержать средства и функционал, позволяющий решить задачу. Но не всегда такой подход эффективен, так как полный переход к новому программному продукту влечет за собой определенные трудности.

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].

# ОПИСАНИЕ САПР

Autodesk Inventor (Инвентор) — система [трёхмерного](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D1%91%D1%85%D0%BC%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0) твердотельного и поверхностного [параметрического проектирования](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%B0%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) ([САПР](https://ru.wikipedia.org/wiki/%D0%A1%D0%90%D0%9F%D0%A0)) компании [Autodesk](https://ru.wikipedia.org/wiki/Autodesk" \o "Autodesk) (Автодеск), 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](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%B0%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) напряженно-деформированного состояния деталей и сборок;
* [визуализация](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%B7%D1%83%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) изделий;
* автоматическое получение и обновление конструкторской документации (оформление по [ЕСКД](https://ru.wikipedia.org/wiki/%D0%95%D0%A1%D0%9A%D0%94)).

Предлагаемый функционал:

* **Проектирование и тестирование в цифровом формате**

Autodesk Inventor обладает интуитивной параметрической рабочей средой для создания концептуальных эскизов и кинематических моделей деталей и изделий. В Inventor автоматически формируются интеллектуальные представления таких компонентов, как пластмассовые детали, металлические рамы, вращающиеся компоненты, трубопроводные системы, электрические провода и кабели. Уделяя меньше времени работе с геометрией, конструкторы могут сконцентрироваться на построении цифровых прототипов для проверки функционирования изделия и оптимизации производственных затрат[5].

* **Прямое моделирование при построении деталей**

В Autodesk Inventor используются принципы прямого моделирования, которые существенно упрощают процесс проектирования и стимулируют творческий подход[5].

* **Динамический ввод данных при отрисовке эскизов**

Динамический ввод использует специальный интерфейс (heads-up display – HUD). При включенном динамическом вводе рядом с курсором появляется поле ввода значения, которое динамически обновляется при перемещении курсора. Использование технологии HUD позволяет задавать геометрию эскиза, не вынося нанесение размеров в отдельный этап. Такой способ построения эскизов полностью совпадает с отрисовкой в AutoCAD®[5].

* **Команда Сборка**

При использовании обновленной команды Сборка нужно выбрать сопрягаемый элемент первой детали и перетащить его на сопрягаемый элемент второй детали для их сборки. Inventor автоматически определит необходимую сборочную зависимость и создаст между ними сопряжение. Если пользователь собирается изменить положение компонентов, новое диагностическое средство автоматически определяет, какие изменения следует внести в зависимости Inventor, чтобы выполнить перестановку с сохранением замысла. Новый инструмент диагностики сборочных зависимостей позволяет убедиться в том, что при их изменении все нужные зависимости в изделии будут сохранены[5].

* **Проектирование пресс-форм**

Семейство Autodesk Inventor содержит средства автоматизированного проектирования пресс- форм, исходные данные для которых берутся из 3D-моделей пластмассовых деталей. Технология расчета пластических течений Autodesk® Moldflow® позволяет оптимизировать процесс с точки зрения технологичности изготовления деталей и уменьшения числа операций[5].

* **Более эффективная совместная работа**

Формирование документации на основе всесторонне протестированного цифрового 3D-прототипа сокращает количество ошибок и последующих извещений об изменениях в проекте. В Inventor обеспечивается быстрый и качественный выпуск рабочих чертежей, информация для которых берется непосредственно из 3D-модели. В комплексы программных средств вместе с Inventor включается AutoCAD® Mechanical – программа для высокопроизводительного 2D-черчения. Inventor тесно интегрирован со средствами управления данными, разработанными компанией Autodesk. Благодаря этому обеспечивается эффективный и безопасный обмен проектной информацией, а также еще на ранних этапах закладываются основы сотрудничества между конструкторскими и производственными подразделениями[5].

## 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 СОМ-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.

Объектная модель Autodesk Inventor API разделяется на:

* объекты общего назначения;
* объекты пользовательского интерфейса;
* объекты документа детали;
* объекты документа сборки;
* объекты документа рисования.

## Описание аналогов разрабатываемого продукта

Существует похожая сборка, по семантике аналогична проектированию инжекторной прокладки. Рассмотрим «Корпус редуктора лебедки подъема»[6] (рисунок 1.1).

Все делается в режиме сборки.

Изначально проектируется крышка редуктора лебедки, далее идёт сборка корпуса редуктора. После, отталкиваясь от заданных параметров крышки и корпуса редуктора лебедки (рисунок 1.2), проектируется прокладка, которая служит для более плотного прилегания крышки к лебедке и уменьшает время износа данной крышки (рисунок 1.3).

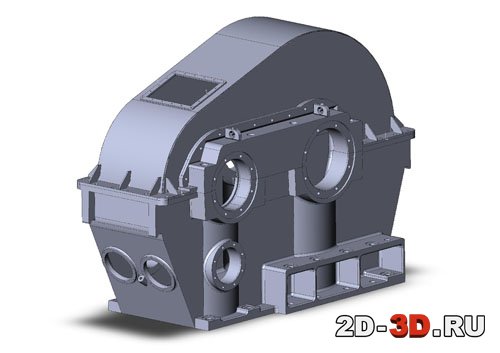


Рисунок 1.1 - «Корпус редуктора лебедки подъема»

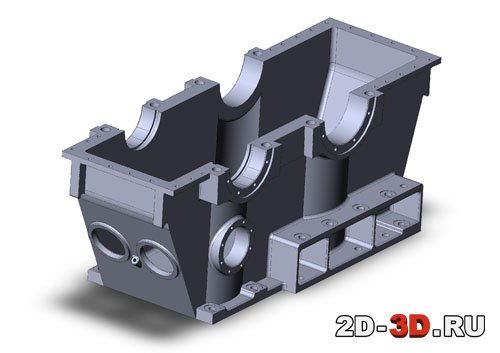
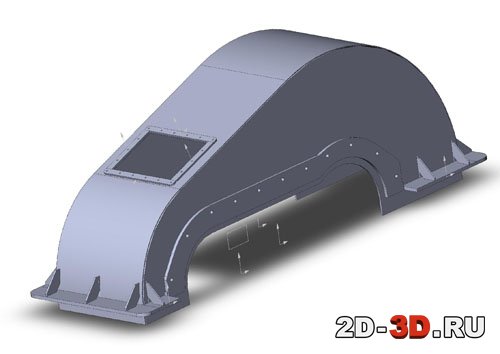


Рисунок 1.2 – Крышка и корпус редукторной лебедки



Рисунок 1.3 – Прокладка крышки редуктора лебедки подъема

# 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:

1. радиус отверстия, находящийся на носу детали (не менее 7мм);
2. радиус выреза на корме детали (не менее 9мм);
3. радиус отверстий по краям детали (не менее 4мм);
4. ширина (должна быть не менее 58мм и не более 62мм);
5. длина (должна быть не менее 90мм и не более 94мм);
6. высота (должна быть не менее 8мм и не более 11мм);
7. расстояние между центрами вырезов (не меньше радиуса выреза);
8. расстояние от центра вырезов (не меньше радиуса выреза);
9. расстояние от начала детали до начала среза (не менее 37мм);
10. расстояние ушка около выреза (не больше радиуса на корме выреза).

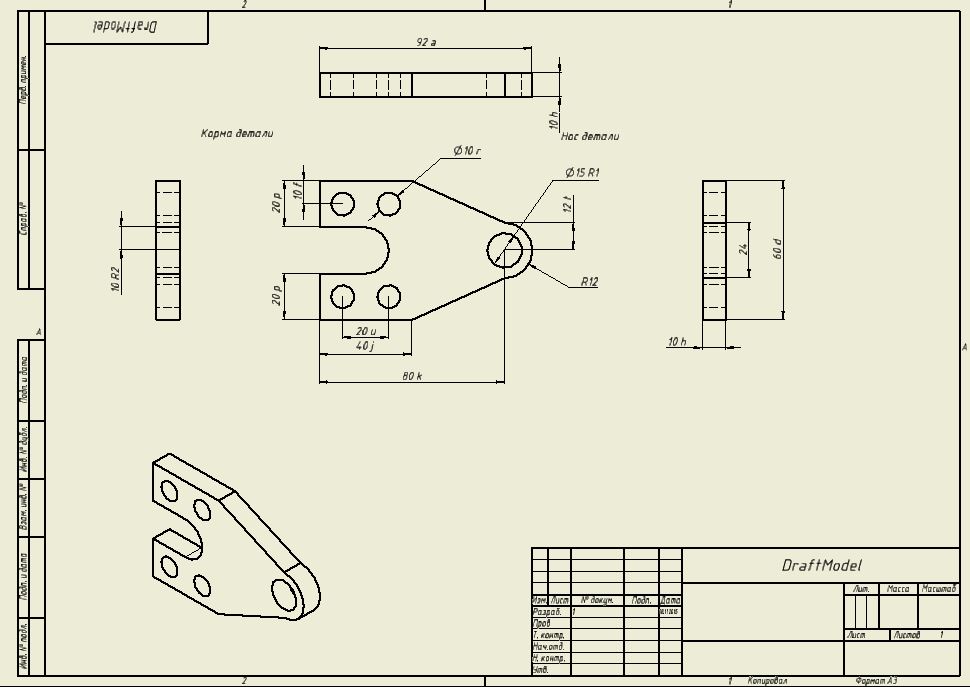


Рисунок 2.1 – Инжекторная прокладка.

# 

# 3 ПРОЕКТ ПРОГРАММЫ

Для формального[7] описания архитектурной особенности, пользовательского сценария системы был выбран унифицированный язык моделирования (UML)[8]. На основе UML построены: диаграммы вариантов использования[9] диаграммы классов[10], диаграммы пакетов[11].

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, поэтому 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 A (Таблица A.1);
* GasketInventorApi − класс, обеспечивающий взаимодействие с методами Inventor API, подробное описание в Приложении A (Таблица A.2);
* GasketProperties − класс, хранящий в себе все параметры модели инжекторной прокладки, осуществляет проверку зависимых параметров, подробное описание в Приложении A (Таблица A.3);
* GasketPropertiesValue − класс предназначен для хранения и обработки параметров модели, подробное описание в Приложении A (Таблица A.4);
* GasketModelCreate − класс построения модели инжекторной прокладки в системе Autodesk Inventor, подробное описание в Приложении A (Таблица A.5);
* GasketPropertiesType – перечисление всех параметров модели инжекторной прокладки, подробное описание в Приложении A (Таблица A.6).



Рисунок 3.2 – UML диаграмма классов

### 3.3 Макеты пользовательского интерфейса

Программа будет состоять из одного диалогового окна с десятью пунктами и двумя кнопками «Запустить Inventor», «Построить деталь» (Рисунок 3.4).

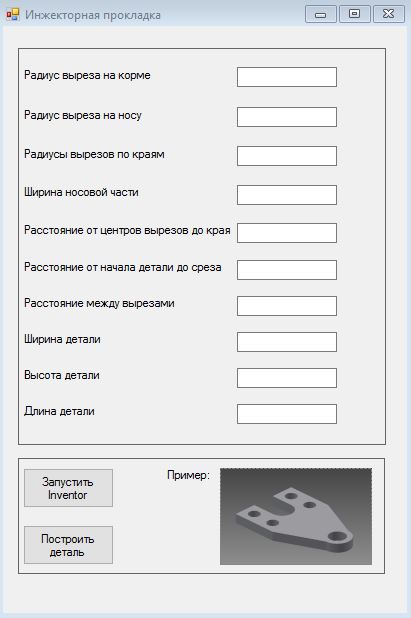


Рисунок 3.4 – Диалоговое окно при старте

# ЗАКЛЮЧЕНИЕ

В ходе проектирования системы были изучены предметная область проектирования, предмет проектирования, аналоги предмета проектирования, API и на основании полученных данных была спроектированы архитектура системы и макет системы. Таким образом, на основе, полученных результатов можно реализовать плагин «Инжекторная прокладка».

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

* + 1. Плагин [Электронный ресурс]. − Режим доступа:

<http://ohostinge.com/dictionary/plug-in> дата обращения (11.10.2015)

* + 1. API [Электронный ресурс]. − Режим доступа:

<http://habrahabr.ru/post/201700/> дата обращения (11.10.2015)

* + 1. САПР [Электронный ресурс]. − Режим доступа:

<https://ru.wikipedia.org/wiki/Система_автоматизированного_проектирования> дата обращения (11.10.2015)

* + 1. Autodesk Inventor (САПР) [Электронный ресурс]. − Режим доступа:

https://ru.wikipedia.org/wiki/Autodesk\_Inventor дата обращения (11.10.2015)

* + 1. Autodesk Inventor (САПР) [Электронный ресурс]. – Режим доступа:

<http://www.autodesk.ru/products/inventor/overview>

дата обращения (11.11.2015)

* + 1. 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 (11.11.2015)

* + 1. Формальный язык [Электронный ресурс]. − Режим доступа:

<http://inf1.info/language> дата обращения (11.10.2015)

* + 1. UML. [Электронный ресурс]. - Режим доступа:

<http://www.uml.org/> дата обращения (11.10.2015)

* + 1. Диаграмма вариантов использования [Электронный ресурс]. – Режим доступа: <http://habrahabr.ru/post/47940/> дата обращения (11.10.2015)
    2. Диаграмма пакетов [Электронный ресурс]. − Режим доступа:

<http://blogs.rsdn.ru/effective/5545045> дата обращения (11.10.2015)

* + 1. Диаграмма классов [Электронный ресурс]. − Режим доступа: <http://www.informicus.ru/Default.aspx?SECTION=6&id=73&subdivisionid=3> обращения (11.10.2015)

# Приложение A

В таблицах приложения для обозначения модификаторов доступа полей приняты следующие условные знаки:

* «−» − обозначение private (открытого) поля;
* «+» − обозначение public (открытого) поля.

Таблица А.1 – Описание полей и методов класса GasketUserInterface

|  |  |
| --- | --- |
| Класс\поля\методы | Описание |
| + GasketUserInterface () | Конструктор класса GasketUserInterface |
| –InitParameters():void | Метод инициализация параметров |
| –WinForm\_Load(object, EventArgs):void | Метод загрузки формы |
| –BuildGasket\_Click(object, EventArgs):void | Метод, обрабатывающий функции построениe инжекторной прокладки |
| –\_inventorApi: InventorApi | Интерфейс API Inventor |
| –\_gasketModelCreate: GasketModelCreate | Модель инжекторной прокладки |
| –\_gasketProperties: GasketProperties = new GasketProperties() {readOnly} | Параметры модели инжекторной прокладки |

Таблица А.2 – Описание полей и методов класса InventorApi

|  |  |
| --- | --- |
| Класс\поля\методы | Описание |

Таблица А. 3– Описание полей и методов класса GasketProperties

|  |  |
| --- | --- |
| Класс\поля\методы | Описание |
| +GasketProperties() | Класс, хранящий в себе все параметры модели, осуществляет проверку зависимых параметров |
| +GetProperty(GasketPropertiesType): GasketPropertiesValue | Возвращает указанный параметр |
| –\_properties: ParameterValue <GasketPropertiesValue, GasketPropertiesType> {readOnly} | Значения параметров вала |

Таблица А.4 – Описание полей и методов класса GasketPropertiesValue

|  |  |
| --- | --- |
| Класс\поля\методы | Описание |
| +Parameter(): double | Класс предназначен для хранения и обработки параметра инжекторной прокладки |
| +Validate(): void | Данный метод меняет значение параметра на корректное в случае выхода его за пределы допустимого диапазона |
| +PropertyChanged(): EventHandler | Данное событие вызывается в случае, если значение параметра было изменено |
| +Value(): double | Данное свойство позволяет установить или запросить текущее значение параметра |
| +Min(): double | Данное свойство позволяет установить или запросить минимальное значение параметра |
| +Max(): double | Данное свойство позволяет установить или запросить максимальное значение параметра |
| –\_max: double | Максимально возможное значение параметра |
| –\_min: double | Минимальное возможное значение параметра |
| –\_value: double | Значение параметра |

Таблица А.5– Описание полей и методов класса GasketModelCreate

|  |  |
| --- | --- |
| Класс\поля\методы | Описание |
| +GasketModelCreate (GasketProperties, InventorApi) | Класс для построения модели |
| +BuildModel(): void | Метод, строящий модель прокладки |
| –\_api: InventorApi{readOnly} | Интерфейс API Inventor |
| –\_ gasketProperties: GasketProperties {readOnly} | Параметры модели инжекторной прокладки |

Таблица А.6– Описание полей и методов класса GasketPropertiesType

|  |  |
| --- | --- |
| Класс\поля\методы | Описание |