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)

# РАЗРАБОТКА ПЛАГИНА «МАСТЕР ШЕСТЕРЁНОК» ДЛЯ «AutoCAD»

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил: студент гр. 580-2

Лубов Г.П.

« » 2023 г.

Руководитель: к.т.н., доцент каф. КСУП

Калентьев А.А.

« » 2023 г.

Томск, 2023

# Лабораторная 3. Проект системы.

Оглавление

1. [Описание САПР 3](#_bookmark1)
   1. [Информация о выбранной САПР 3](#_bookmark2)
   2. [Описание API 3](#_bookmark3)
   3. [Обзор аналогов плагина 6](#_bookmark4)
2. [Описание предмета проектирования 10](#_bookmark5)
3. [Проект системы 11](#_bookmark6)
   1. [Диаграмма классов 11](#_bookmark7)
   2. [Макеты пользовательского интерфейса 13](#_bookmark8)
4. [Список используемых источников 14](#_bookmark9)

# Описание САПР

* 1. Информация о выбранной САПР

Autodesk AutoCAD — система автоматизированного проектирования (САПР) для создания трёх- и двухмерных моделей. Позволяет выполнять построение 3D-моделей деталей, объединять их в сборки, а также выполнять чертежи и инженерные расчёты физических характеристик. AutoCAD и специализированные приложения на его основе применяются в области машиностроения, строительства, архитектуры и т.д. Программа имеет русскую локализацию [1].

Прямым аналогом разрабатываемого плагина является инструмент “Цилиндрическая зубчатое зацепление” в Autodesk Inventor.

Косвенными аналогами разрабатываемого плагина являются САПР Autodesk Fusion 360 и Kompas-3D.

* 1. Описание API

API (Application Program Interface) – программный интерфейс приложения, набор функций, позволяющий взаимодействовать с программой через другие программы. Для AutoCAD есть API на двух языка программирования: C#/.NET и Python. Для разработки плагина, рассматриваемого в данной работе, будет использоваться API для языка C#/.NET [2].

Основные библиотеки API представлены ниже:

1. AcDbMgd.dll. Используется для работы с объектами файла чертежа;
2. AcMgd.dll. Используется для работы с самим приложением AutoCAD;
3. AcCui.dll. Используется для работы с файлами пользовательских настроек;

Некоторые используемые методы и классы API представлены в таблицах 1.1 – 1.6.

Таблица 1.1 – Некоторые используемые классы API

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| DocumentManager | DocumentManager | Класс, хранящий коллекцию открытых документов AutoCAD |
| Database | Database | Класс, хранящий все графические и большинство неграфических объектов AutoCAD |
| TransactionManager | TransactionManager | Класс, обрабатывающий все транзакции и работающий с реестром операционной системы |
| Line | Line | Класс, определяющий прямую линию |
| Point3d | Point3d | Класс, определяющий точку в трёхмерном пространстве |
| Circle | Circle | Класс, определяющий окружность |
| Solid3d | Solid3d | Класс, позволяющий создавать тело |

Таблица 1.2 – Некоторые используемые методы API

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Line() | Point3d, Point3d | Line | Создаёт линию через 2 точки в трёхмерном пространстве |
| Circle.Center() | Point3d | void | Устанавливает центр окружности |
| Circle.Radius() | double | void | Устанавливает радиус окружности |
| Solid3d() | Array<Point3d> | Solid3d | Создаёт твёрдое тело по точкам в пространстве |
| Object. ArrayPolar() | NumberOfObjects, AngleToFill, CenterPoint | Array<Object> | Создаёт полярный массив из элементов |

Таблица 1.3 – Основные методы интерфейса DocumentManager

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых  данных | Описание |
| MdiActiveDocument() | - | Document | Возвращает созданный документ чертежа |
| MdiActiveDocument() | - | Editor | Возвращает редактор чертежа |

Таблица 1.4 – Основные методы интерфейса TransactionManager

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых  данных | Описание |
| StartManager() | - | Transaction | Реализует работу с примитивами |
| Transaction.Commit() | - | Editor | Завершает работу с примитивами |

Таблица 1.5 – Основные методы класса BlockTableRecord

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых  данных | Описание |
| AppendEntity() | object | void | Реализует работу с примитивами |

Таблица 1.6 – Основные методы класса Solid3d

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых  данных | Описание |
| CreateWedge() | Double, double | void | Создаёт объект по заданной длине, ширине, высоте |
| Extrude() | Int, double, double | void | Выполняет выдавливание области на заданную длину |

* 1. Обзор аналогов плагина

“Цилиндрическое зубчатое зацепление” ­­– инструмент в программе Autodesk Inventor, позволяющий создавать зубчатые передачи ременного, цепного, червячного, цилиндрического и конического зацепления косозубого и прямозубого типов по заданным параметрам [3]. С помощью данного инструмента можно в автоматизированном режиме получить готовый узел зубчатой передачи по заданным параметрам. В данный момент данный программный продукт на территории РФ не распространяется. Пользовательский интерфейс инструмента представлен на рисунке 1.1.

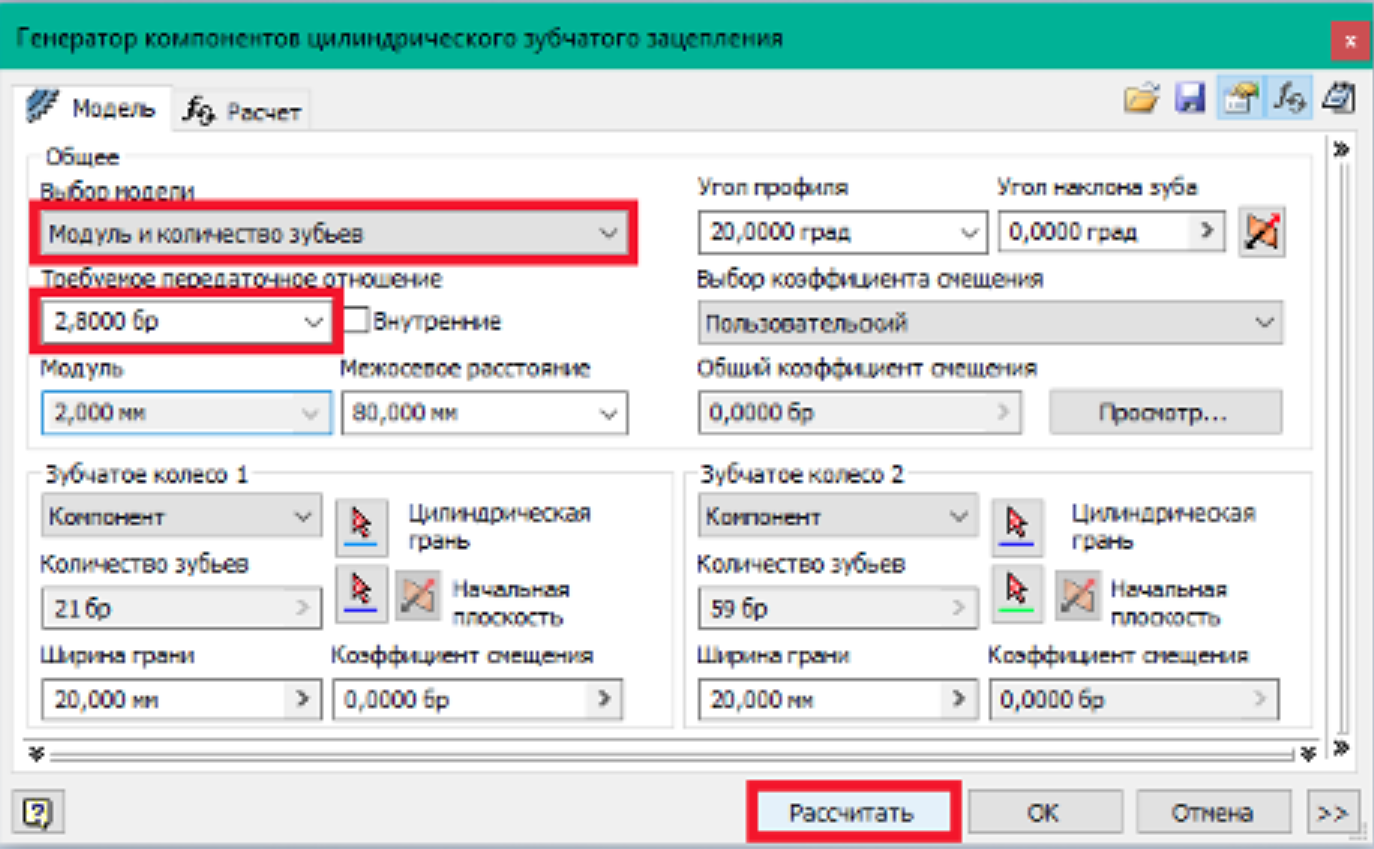


Рисунок 1.1 – Пользовательский интерфейс инструмента “Цилиндрическое зубчатое зацепление” в программе Autodesk Inventor

Autodesk Fusion 360 – система автоматизированного проектирования, включающая в себя модули автоматизированного геометрического моделирования, инженерных расчётов, производства, проектирования печатных плат и автоматизации формирования конструкторской документации. Преимуществом данного программного продукта является простой пользовательский интерфейс и возможность работы в облаке с конструкторской документацией [4]. С помощью данного программного продукта можно вручную создать модель шестерни. В данный момент этот программный продукт на территории РФ не распространяется. Пользовательский интерфейс программы представлен на рисунке 1.2.

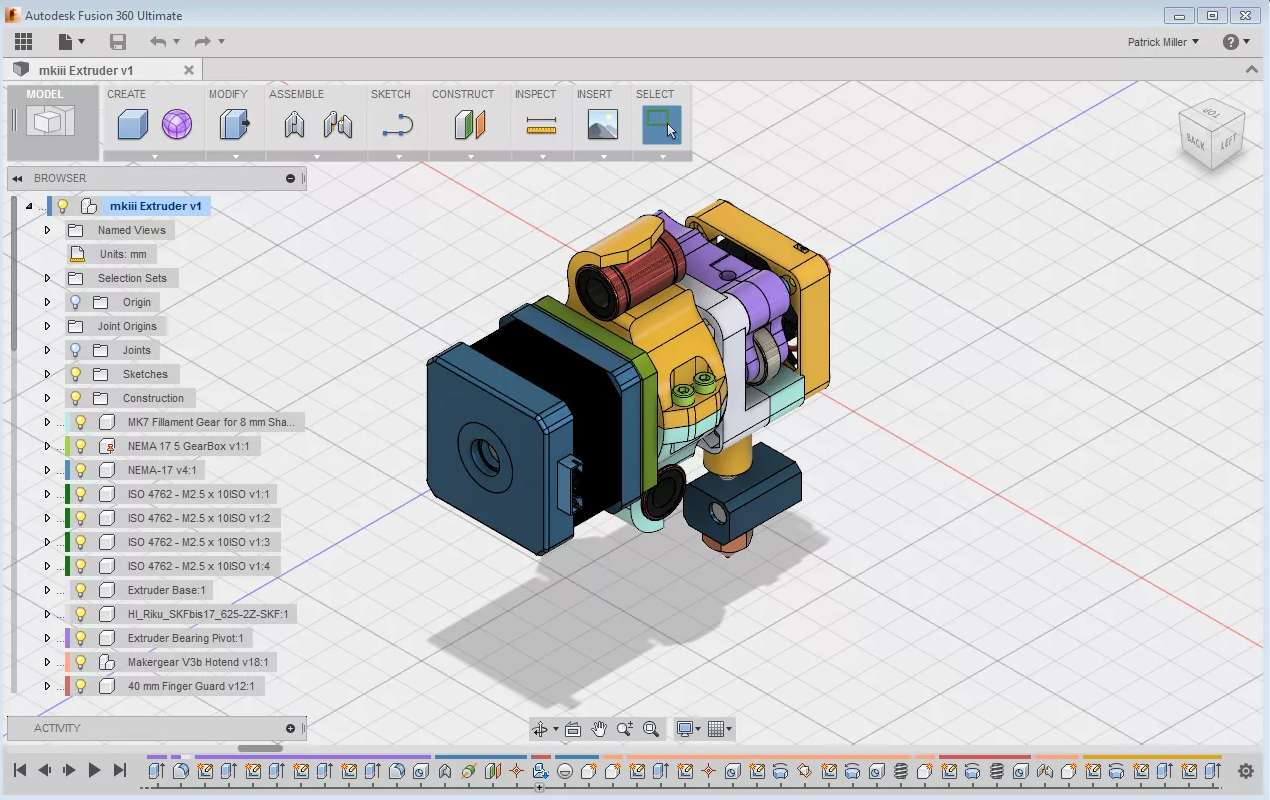


Рисунок 1.2 – пользовательский интерфейс Autodesk Fusion 360

Kompas-3D – система автоматизированного проектирования отечественной разработки, включающая в себя модули автоматизированного геометрического моделирования, инженерных расчётов, производства, проектирования печатных плат, автоматизации и формирования конструкторской документации [5]. Изначально система была ориентирована на создание конструкторской документации в соответствии с ЕСКД, ЕСТД, СПДС и международными стандартами. Преимуществом данного программного обеспечения является его доступность в РФ, в отличие от импортных САПР. С помощью данного программного продукта можно вручную создать модель шестерни. Пользовательский интерфейс программы представлен на рисунке 1.3.

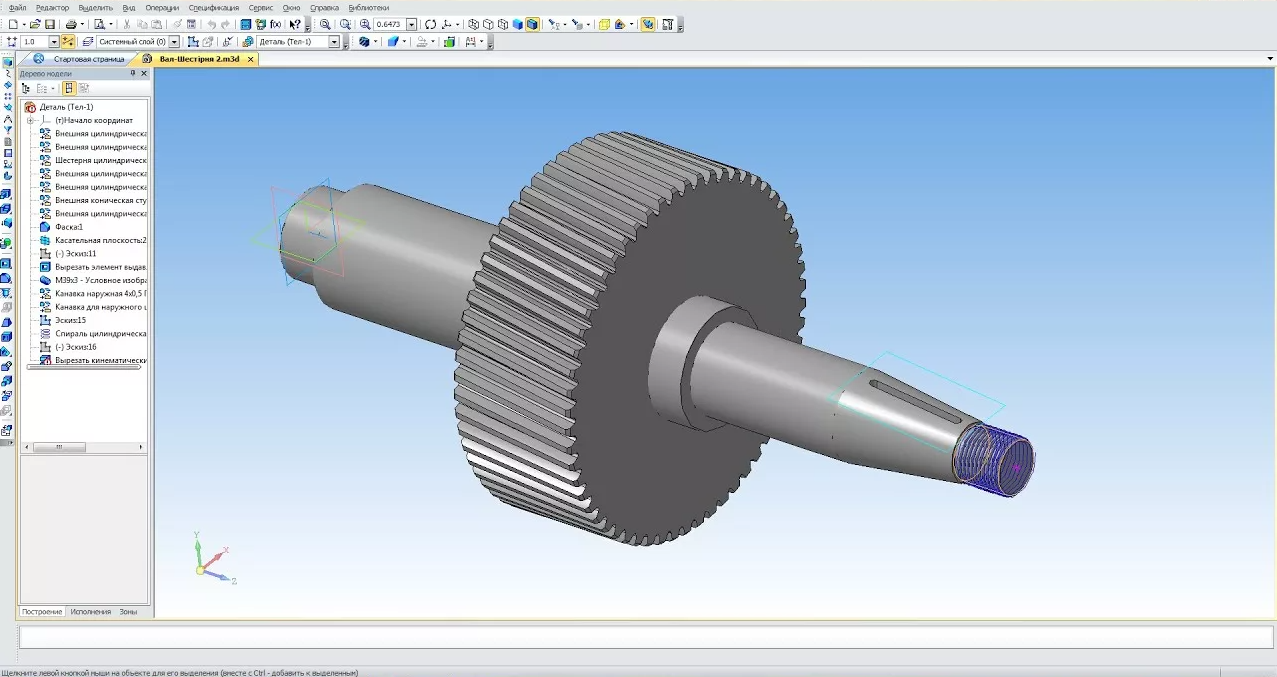


Рисунок 1.3 – пользовательский интерфейс Kompas-3D

# Описание предмета проектирования

Зубчатое колесо (шестерня) — основная деталь зубчатой передачи в виде диска с зубьями на цилиндрической или конической поверхности, входящими в зацепление с зубьями другого зубчатого колеса [6]. Чертёж шестерни представлен на рисунке 2.1.



Рисунок 2.1 – чертёж шестерни

Шестерня имеет следующие параметры:

– внешний диаметр шестерни D (1 — 1000 мм, но не менее d);

– диаметр посадочного отверстия d (1 — 999 мм, но не более, чем 9D/10);

– толщина шестерни S (1 — 1000 мм)

– количество зубов N (3 — 1000 шт.);

– высота зуба h (0,1 — 999 мм, но не более, чем диаметр посадочного отверстия d).

1. **Проект системы**

3.1 Диаграмма классов

UML-диаграмма классов — тип статической структурной диаграммы, описывающей структуру системы посредством обозначения классов, их атрибутов, методов, связей на диаграмме [7-10].

На рисунке 3.1 отображена диаграмма классов приложения.

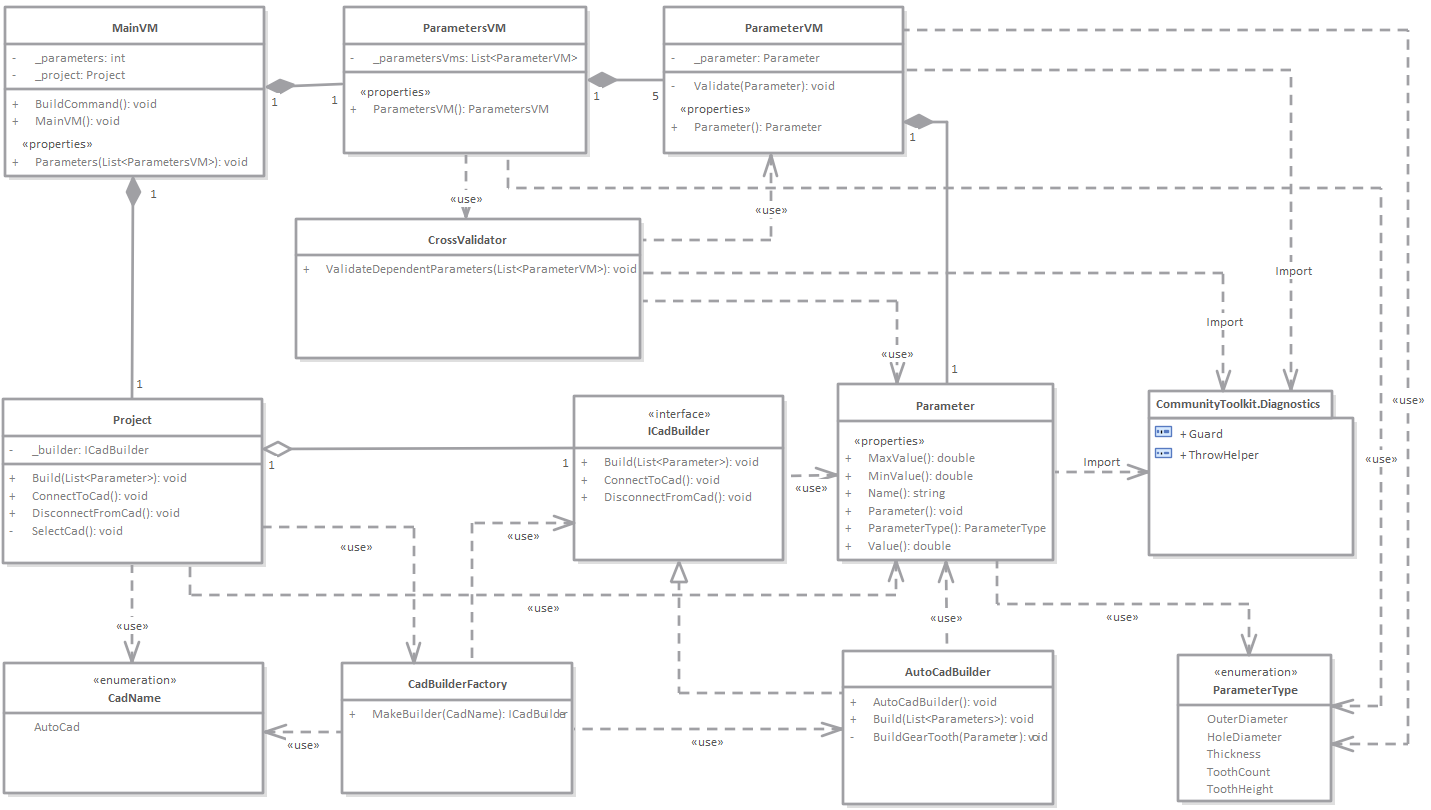


Рисунок 3.1 – Пример UML-диаграммы классов

Архитектура приложения реализована по паттерну MVVM.

Класс MainVM представляет собой объект, через который будет осуществляться обработка пользовательского ввода и передача его в модель (таблица 3.1).

Таблица 3.1 – Описание класса MainVM

|  |  |
| --- | --- |
| Название | Описание |
| \_project | Хранит экземпляр проекта |
| \_parameters | Хранит параметры шестерни |
| BuildCommand | Вызывает команду построения шестерни |
| MainVM | Конструктор |
| ParamatersVM | Хранит все параметры шестерни |

Класс Parameters представляет группу параметров моделируемого объекта (таблица 3.2).

Таблица 3.2 – Описание класса ParametersVM

|  |  |
| --- | --- |
| Название | Описание |
| \_parametersVms | Параметры шестерни |
| ParametersVM | Конструктор |

Класс ParameterVM является представлением параметра, который отвечает за первичную валидацию параметра и дальнейшую отправку этого параметра на уровень модели (таблица 3.3).

Таблица 3.3 – Описание класса ParameterVM

|  |  |
| --- | --- |
| Название | Описание |
| Parameter | Хранит параметр |
| \_parameter | Хранит параметр |
| Validate | Выполняет валидацию текущего фрагмента |

Класс CrossValidator осуществляет перекрёстную валидацию между зависимыми параметрами (таблица 3.4).

Таблица 3.4 – Описание класса CrossValidator

|  |  |
| --- | --- |
| Название | Описание |
| ValidateDependentParameters | Валидирует зависимые параметры |

Класс Project является главным классом модели. Через него происходит взаимодействие с САПР, выполняется подключение, отключение, выбор целевой САПР и выполняется построение модели (таблица 3.5).

Таблица 3.5 – Описание класса Project

|  |  |
| --- | --- |
| Название | Описание |
| \_builder | Хранит экземпляр обёртки над построителем |
| Build | Выполняет построение модели |
| ConnectToCad | Выполняет подключение к САПР |
| DisconnectFromCad | Выполняет отключение от САПР |
| SelectCad | Выбирает конкретную САПР для использования при построении |

Интерфейс ICadBuilder абстрагирует Project от конкретной реализации построителя (таблица 3.6).

Таблица 3.6 – Описание интерфейса ICadBuilder

|  |  |
| --- | --- |
| Название | Описание |
| Build | Выполняет построение модели в САПР |
| ConnectToCad | Выполняет подключение к САПР |
| DisconnectFromCad | Выполняет отключение от САПР |

Класс Parameter выполняет хранение информации об одном из параметров модели (таблица 3.7).

Таблица 3.7 – Описание класса Parameter

|  |  |
| --- | --- |
| Название | Описание |
| MaxValue | Хранит максимальное допустимое значение параметра |
| MinValue | Хранит минимальное допустимое значение параметра |
| Name | Хранит название параметра |
| Parameter | Конструктор |
| Value | Значение параметра |
| Type | Тип параметра |

Перечисление CadName отображает названия САПР, с которыми в настоящий момент времени может работать плагин (таблица 3.8).

Таблица 3.8 – Описание перечисления CadName

|  |  |
| --- | --- |
| Название | Описание |
| AutoCad | Перечисление для Autodesk AutoCAD |

Класс CadBuilderFactory предоставляет конкретный экземпляр ICadBuilder классу Project. Данное решение применено для упрощения модификации плагина в будущем на тот случай, если потребуется подключить этот плагин к другой САПР (таблица 3.9).

Таблица 3.9 – Описание класса CadBuilderFactory

|  |  |
| --- | --- |
| Название | Описание |
| MakeBuilder | Создаёт экземпляр конкретного построителя модели в соответствующей САПР |

Класс AutoCadBuilder предоставляет конкретную реализацию построителя для САПР AutoCAD (таблица 3.10).

Таблица 3.10 – Описание класса AutoCadBuilder

|  |  |
| --- | --- |
| Название | Описание |
| AutoCadBuilder | Конструктор |
| Build | Выполняет построение модели в AutoCAD |
| BuildGearTooth | Выполняет построение одного зуба в AutoCAD |

Библиотека CommunityToolkit.Guard предоставляет реализацию методов валидации без необходимости реализовывать их самостоятельно.

Таблица 3.11 – Описание класса ParametersVM

|  |  |
| --- | --- |
| Название | Описание |
| Guard | Реализует методы валидации значений на минимум, максимум, т.п. |
| ThrowHelper | Реализует методы для облегчения выбрасывания исключений |

Запуск плагина предполагается выполнять из САПР через командную строку.

* 1. Макеты пользовательского интерфейса

Пример макета пользовательского интерфейса представлен на рисунке 3.2.

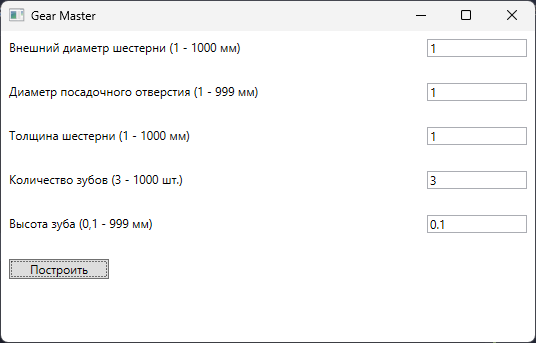


Рисунок 3.2 — Пользовательский интерфейс

Валидация некорректных данных представлена на рисунках 3.3, 3.4.

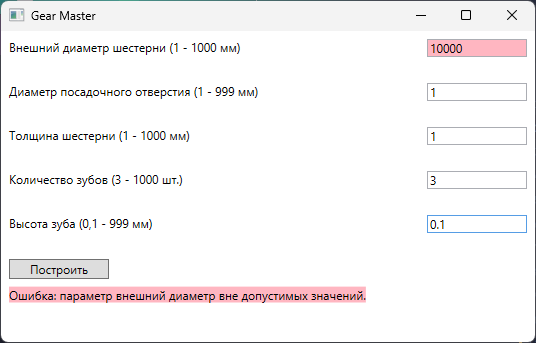


Рисунок 3.3 — Интерфейс с неправильно введенным значением параметра

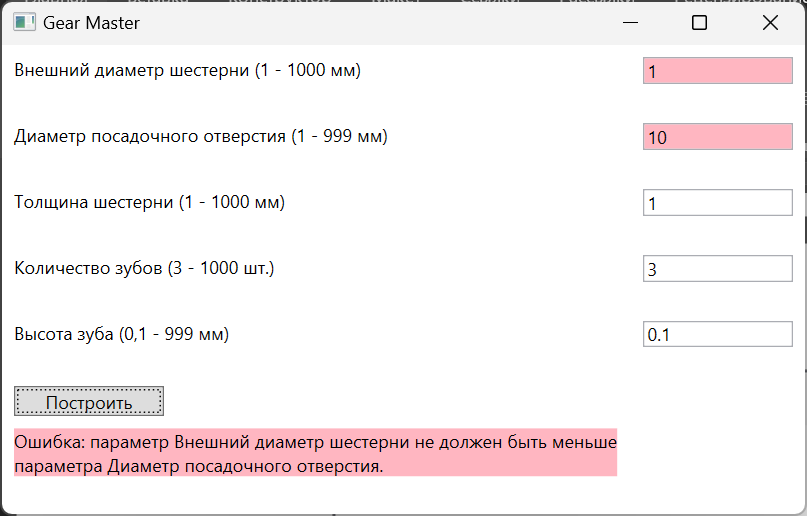


Рисунок 3.4 — Интерфейс с неправильно введенными значениями параметров

# Список используемых источников

1. Языковые пакеты для САПР Autodesk AutoCAD. [Электронный ресурс]. Режим доступа: свободный (дата образения 16.10.23). https://www.autodesk.com/support/technical/article/caas/tsarticles/ts/2edY5Oczsv93tDF1ugCKHU.html
2. Документация AutoCAD .NET API для разработчиков. [Электронный ресурс]. Режим доступа: свободный (дата обращения 16.10.23), https://help.autodesk.com/view/OARX/2022/ENU/?guid=GUID-390A47DB-77AF-433A-994C-2AFBBE9996AE
3. Обучающая статья “Autodesk Inventor. Построение зубчатой передачи”. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), https://www.pointcad.ru/novosti/autodesk-inventor.-postroenie-zubchatoj-peredachi
4. Официальный сайт САПР Autodesk Fusion 360. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), https://www.autodesk.com/products/fusion-360/overview?term=1&tab=subscription
5. Официальный сайт САПР Kompas-3D. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), <https://kompas.ru/kompas-3d/about/>
6. Энциклопедия “Академик”. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), <https://dic.academic.ru/dic.nsf/ruwiki/1200290>
7. Руководство “What is Class Diagram”. [Электронный ресурс]. Режим доступа: свободный (дата обращения 09.10.23), https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/
8. UML. Основы. Краткое руководство по стандартному языку объектного моделирования. Изд: Символ-Плюс, 2011, с.192 (3-е издание)
9. Язык UML. Руководство пользователя. Изд: ДМК Пресс, 2015, с.496 Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления, Томск 2021 г., 52 с.
10. Применение UML 2.0 и шаблонов проектирования. Введение в объектно- ориентированный анализ, проектирование и итеративную разработку. Изд: Вильямс, 2013, с.739 (3-е издание)