Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ

УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**РАЗРАБОТКА ПЛАГИНА «Киприч» ДЛЯ «КОМПАС-3D»**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил:

студент гр. 582-1

\_\_\_\_\_\_\_\_\_ Ершов Г. А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Проверил:

к.т.н., доцент каф. КСУП \_\_\_\_\_\_\_\_\_\_ Калентьев А.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

1 Описание САПР

# Описание программы

КОМПАС-3D – это российская импортонезависимая система трёхмерного проектирования, ставшая стандартом для тысяч предприятий и сотен тысяч профессиональных пользователей.

КОМПАС-3D широко используется для проектирования изделий основного и вспомогательного производств в таких отраслях промышленности, как машиностроение (транспортное, сельскохозяйственное, энергетическое, нефтегазовое, химическое и т.д.), приборостроение, авиастроение, судостроение, станкостроение, вагоностроение, металлургия, промышленное и гражданское строительство, товары народного потребления и т. д.[1]

Данная САПР позволяет проектировать модели и сборки разного уровня сложности, благодаря разнообразному функционалу, включающего в себя работу как с 2-мерными эскизами, так и с 3D-моделями. В САПР есть возможность работать со всеми основными примитивами необходимыми для создания эскизов и моделей, а также существует достаточное количество инструментов для работы с 3D-моделями (вытягивание, вращение, вырезание и др.).

Компас 3D имеет множество прямых аналогов на рынке, среди них встречаются Autodesk Inventor, SOLIDWORKS и др.

В рамках дисциплины выбор данной САПР объясняется наличием описания API на русском языке, доступность учебной версии САПР без необходимости получать одобрения от компании, а также большим количеством информации на сторонних ресурсах на русском языке, позволяющим детальнее узнать о возможностях работы с САПР.

# Описание API

**API** (Application Programming Interface) – интерфейс программирования приложений, описывающий способы взаимодействия программы с внешними компонентами.

Для подключения и работы с API на C# потребуется выполнить ряд следующих действий:

1. Включить в свойствах проекта функцию Register for COM Interop;
2. Создать DLL-обёртку для TLB Компас API с помощью Tlblmp.exe;
3. Подключить созданный DLL к проекту;
4. Зарегистрировать библиотеку в системе КОМПАС (а именно реализовать статический метод типа .htmSample с рядом настроек)
5. Зарегистрировать библиотеку на компьютере пользователя, воспользовавшись утилитой RegAsm.exe

В таблице 1.1 представлены интерфейсы, которые будут использованы при разработке плагина.

Таблица 1.1 – Интерфейсы, используемые при разработке

|  |  |
| --- | --- |
| Интерфейс | Описание |
| KompasObject | Корневой объект API КОМПАС, предоставляет доступ к документам и сервисным функциям приложения. |
| ksEntity | Базовый элемент модели (эскиз, операция и т.д.), оболочка над объектом параметров Definition. |
| ksDocument2D | 2D-документ/редактор, используемый для построения геометрических примитивов. |
| ksSketchDefinition | Управляет началом и завершением редактирования эскиза. |
| ksDocument3D | 3D-документ, в котором создаются детали и сборки. |

Окончание таблицы 1.1 – Интерфейсы, используемые при разработке

|  |  |
| --- | --- |
| Интерфейс | Описание |
| ksPart | Часть модели (деталь или сборка), через неё создаются объекты и операции. |
| ksBaseExtrusionDefinition | Определяет параметры операции выдавливания. |
| ksBossRotatedDefinition | Определяет параметры операции вращения (вращательное тело). |
| ksCutExtrusionDefinition | Определяет параметры операции выреза выдавливанием. |
| ksEdgeFilletDefinition | Определяет параметры скруглений и фасок кромок. |

Таблица 1.2 – Методы интерфейса KompasObject

|  |  |  |
| --- | --- | --- |
| Метод | Возвращаемое значение | Описание |
| Document3D() | ksDocument3D | Получает указатель на 3D-документ (деталь/сборку). |
| ActivateControllerAPI() | bool | Активирует контроллер API КОМПАС-3D. |
| Visible() | bool | Возвращает или задает видимость окна КОМПАС-3D. |

Таблица 1.3 – Методы интерфейса ksEntity

|  |  |  |
| --- | --- | --- |
| Метод | Возвращаемое значение | Описание |
| Create() | bool | Создает объект в модели. |
| GetDefinition() | IUnknown | Возвращает интерфейс параметров (Definition) связанного объекта. |

Таблица 1.4 – Методы интерфейса ksDocument2D

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| ksCircle(xc, yc, rad, style) | int | xc, yc – координаты центра;  rad – радиус; style – стиль линии | Создает окружность на активном эскизе. |

Таблица 1.5 – Методы интерфейса ksSketchDefinition

|  |  |  |
| --- | --- | --- |
| Метод | Возвращаемое значение | Описание |
| BeginEdit() | bool | Открывает режим редактирования эскиза. |
| EndEdit() | bool | Завершает редактирование эскиза. |

Таблица 1.6 – Методы интерфейса ksDocument3D

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| Create(invisible, typeDoc) | bool | invisible – скрытый режим; typeDoc – тип документа (деталь/сборка) | Создает новый 3D-документ. |
| GetPart(type) | ksPart | type – тип части (pTop\_Part и др.) | Возвращает компонент (деталь или сборку) указанного типа. |

Таблица 1.7 – Методы интерфейса ksPart

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| GetDefaultEntity(objType) | ksEntity | objType – тип объекта | Возвращает объект, созданный системой по умолчанию (например, плоскость, ось). |
| NewEntity(objType) | ksEntity | objType – тип создаваемого объекта | Создает новый объект (эскиз, операция и т.д.). |

Таблица 1.8 – Методы интерфейса ksBaseExtrusionDefinition

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| SetSideParam(forward, type, depth, draftValue, draftOutward) | bool | forward – направление; type – тип; depth – глубина; draftValue – уклон; draftOutward – направление уклона | Задает параметры выдавливания. |
| SetSketch(sketch) | bool | sketch – эскиз операции | Назначает эскиз для операции выдавливания. |

Таблица 1.9 – Методы интерфейса ksBossRotatedDefinition

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| SetSketch(sketch) | bool | sketch – эскиз профиля | Назначает профиль для операции вращения. |
| SetAxis(axis) | bool | axis – ось вращения | Устанавливает ось вращения. |
| SetAngle(angle) | bool | angle – угол вращения (в градусах) | Задает угол вращения (обычно 360°). |

Таблица 1.10 – Методы интерфейса ksCutExtrusionDefinition

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| SetSketch(sketch) | bool | sketch – эскиз выреза | Назначает эскиз для операции выреза. |
| SetSideParam(forward, type, depth, draftValue, draftOutward) | bool | forward – направление; type – тип; depth – глубина; draftValue – уклон; draftOutward – направление уклона | Задает параметры выреза. |

Таблица 1.11 – Методы интерфейса ksEdgeFilletDefinition

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Возвращаемое значение | Параметры | Описание |
| SetRadius(radius) | bool | radius – радиус скругления | Задает радиус скругления кромок. |
| AddEdge(edge) | bool | edge – ссылка на ребро модели | Добавляет ребро, к которому применяется скругление. |
| Create() | bool | – | Создает операцию скругления с заданными параметрами. |

# Обзор аналогов плагина

Программа автоматического построения 3D моделей и разверток по заданным значениям в AutoCAD «Лекало». Расчет и построение механических передач

Данная программа позволяет создавать следующие 3D модели в AutoCAD посредством ввода размеров с клавиатуры:

* металлопрокат;
* механические соединения;
* механические передачи;
* элементы гидро- и пнемвоприводов;
* построение конструктивных элементов [4].

На рисунке 1.1 представлен пользовательский интерфейс программы «Лекало» для построения втулки.



Рисунок 1.1 – Пользовательский интерфейс программы «Лекало» для построения втулки

Вторым аналогом является плагин Плагин «Fasteners» для программы FreeCAD.

Данный плагин предназначен для моделирования метрических болтов и гаек по стандартам ISO [5].



Рисунок 1.2 – Пользовательский интерфейс плагина «Fasteners»

# Описание предмета проектирования

Кирпич — это строительный материал, предназначенный для возведения стен, перегородок и других конструктивных элементов зданий. Кирпич имеет стандартную форму параллелепипеда и может содержать сквозные отверстия для уменьшения веса и повышения теплоизоляционных свойств.

Изменяемые параметры (из ТЗ на рис. 2.1):

* Длина кирпича L (100 — 1000мм);
* Общая высота кирпича H (30 — 300мм);
* Ширина кирпича W (50 — 500мм);
* Радиус r в отверстиях кирпича (от 2 до 30мм и не должен превышать 25% ширины кирпича);
* Количество отверстий n (от 0 до максимально возможного k, зависит от r, отверстия не должны заходить за рамки W\_a и L\_a (обозначены на рис. 2.2)).

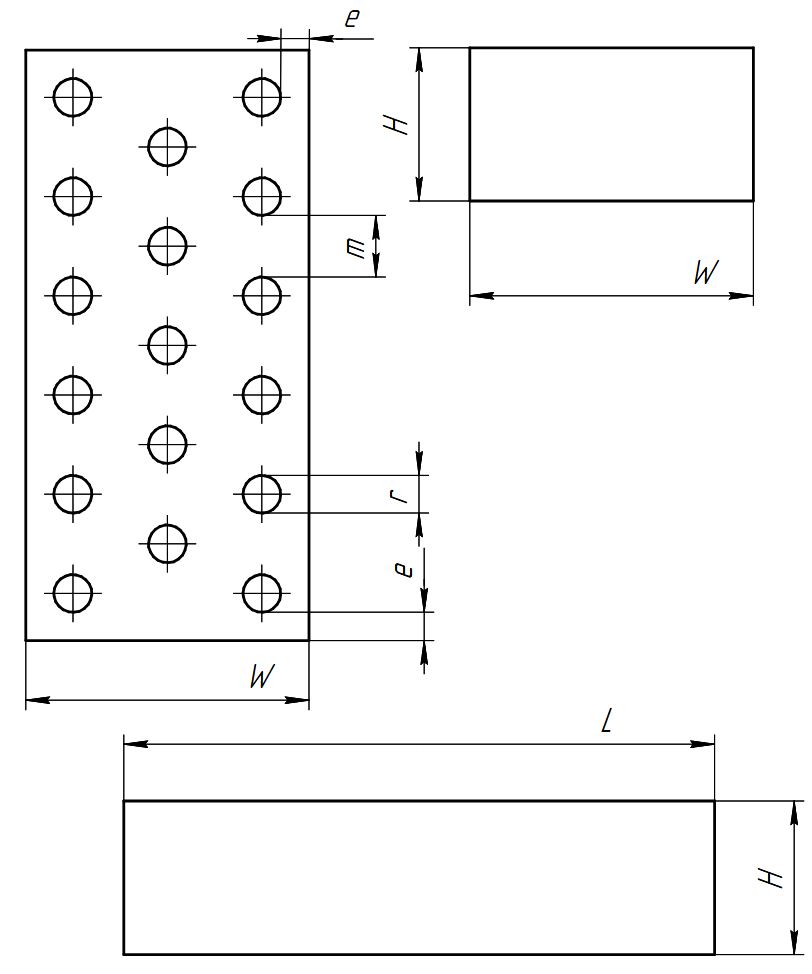


Рисунок 2.1 – Параметризованная модель кирпича с обозначением размеров.

Неизменяемые параметры для плагина (нужны для определения максимально возможного количество отверстий, обозначены на рис. 2.2):

* Допустимая максимальная ширина для установки отверстий W\_a (W - 2e = W - 2(r + 5));
* Допустимая максимальная длина для установки отверстий L\_a (L\_a = L - 2e = L - 2(r + 5));
* Ширина кирпича W (50 — 500мм);
* Минимальное расстояние до края кирпича e (e = r +5мм);
* Минимальное расстояние между отверстиями m (m = 2r + max(0.25r, 5 мм) (максимум между двумя величинами 0.25r и 5 мм)).

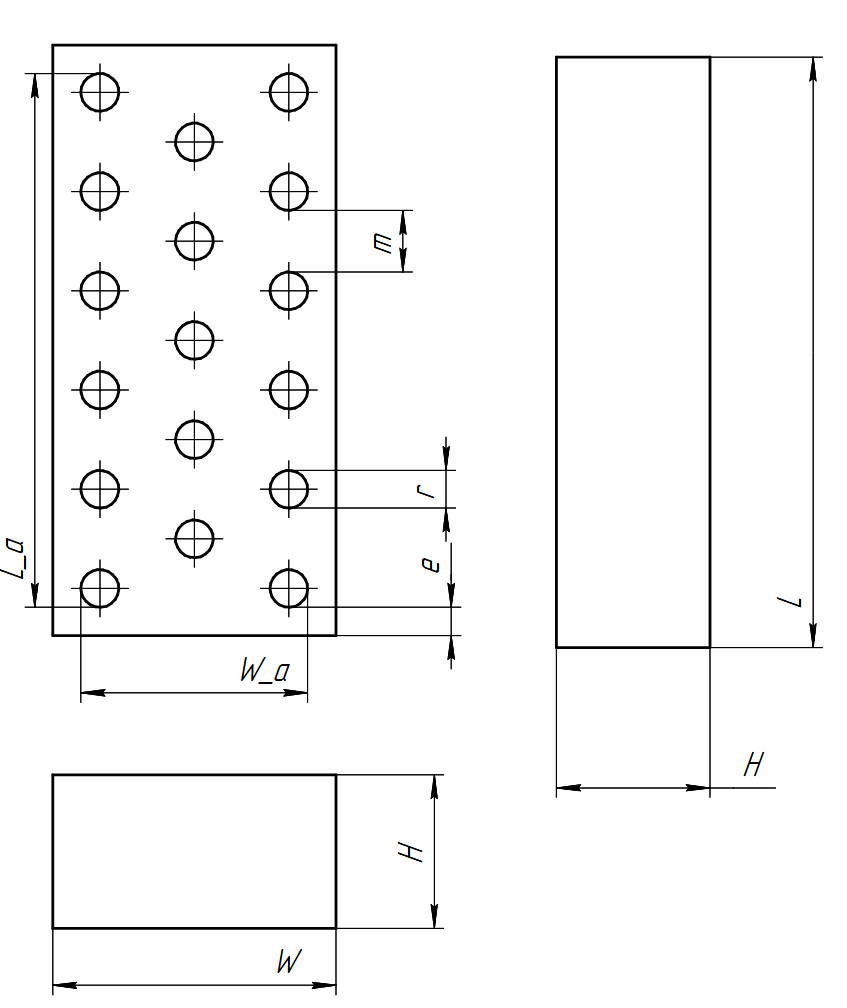


Рисунок 2.2 — Параметризованная модель кирпича с обозначением неизменяемых размеров.

# Проект системы

# UML‑диаграмма классов

UML – стандартный язык визуального моделирования для спецификации, проектирования и документирования систем. В проекте используется диаграмма классов для описания структуры плагина.

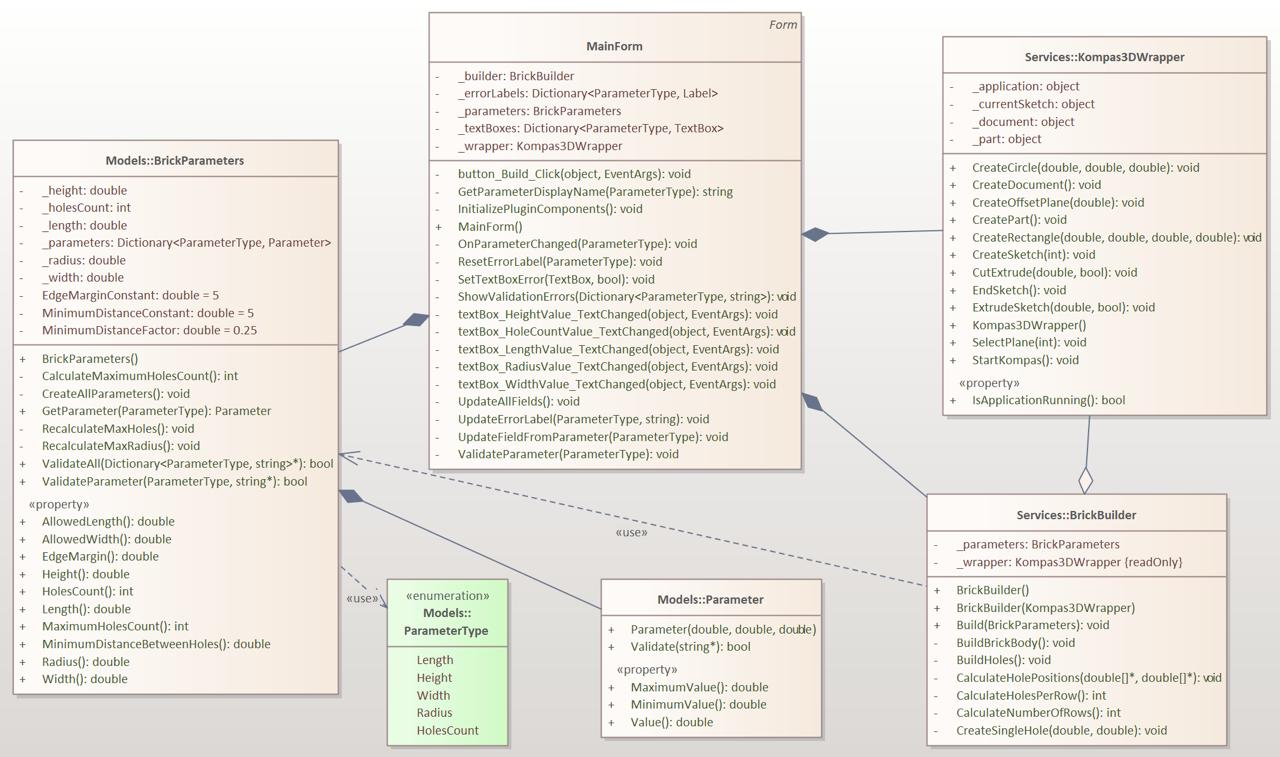


Рисунок 3.1 – UML‑диаграмма классов плагина «Кирпич».

Таблица 3.1 – Поля класса Parameter

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Value | double | Текущее значение параметра. |
| MinimumValue | double | Минимально допустимое значение параметра. |
| MaximumValue | double | Максимально допустимое значение параметра. |

Таблица 3.2 – Используемые методы класса Parameter

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Parameter | minimum: double, maximum: double, defaultValue: double | — | Конструктор. Инициализирует параметр с заданными ограничениями и значением по умолчанию. |
| Validate | errorMessage: out string | bool | Проверяет, находится ли текущее значение в допустимом диапазоне. Возвращает false и сообщение об ошибке при нарушении ограничений. |

Продолжение таблицы 3.3 – Используемые методы класса BrickBuilder

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| BrickBuilder | wrapper: Kompas3DWrapper | — | Конструктор с параметром. Инициализирует строитель с заданной обёрткой API. |
| Build | parameters: BrickParameters | void | Основной метод построения модели кирпича. Управляет вызовом методов построения тела и отверстий. |
| BuildBrickBody | – | void | Создаёт основное тело кирпича (параллелепипед). |
| BuildHoles | – | void | Создаёт отверстия в кирпиче согласно заданным параметрам. |

Окончание таблицы 3.3 – Используемые методы класса BrickBuilder

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| CalculateHolePositions | xPositions: out double[], yPositions: out double[] | void | Вычисляет координаты центров отверстий на верхней грани кирпича. |
| CreateSingleHole | x: double, y: double | void | Создаёт одно цилиндрическое отверстие в указанной позиции. |
| CalculateHolesPerRow | – | int | Вычисляет количество отверстий в одном ряду. |
| CalculateNumberOfRows | – | int | Вычисляет количество рядов отверстий. |

Таблица 3.3 – Используемые методы класса MainForm

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| MainForm | – | – | Конструктор формы. Инициализирует компоненты и обновляет поля. |
| InitializePluginComponents | – | void | Инициализирует внутренние компоненты плагина: параметры, обёртку API и строитель. |
| OnParameterChanged | parameterType: ParameterType | void | Обрабатывает изменение параметра: обновляет зависимые параметры и выполняет валидацию. |
| UpdateFieldFromParameter | parameterType: ParameterType | void | Обновляет текстовое поле на форме значением из объекта параметров. |

Продолжение таблицы 3.3 – Используемые методы класса MainForm

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| ValidateParameter | parameterType: ParameterType | void | Выполняет валидацию параметра и обновляет визуальное отображение ошибок. |
| SetTextBoxError | textBox: TextBox, hasError: bool | void | Устанавливает или сбрасывает визуальную индикацию ошибки для текстового поля (изменяет цвет фона). |
| UpdateErrorLabel | parameterType: ParameterType, errorMessage: string | void | Обновляет метку ошибки для указанного параметра. |
| ResetErrorLabel | parameterType: ParameterType | void | Сбрасывает метку ошибки. |

Продолжение таблицы 3.3 – Используемые методы класса MainForm

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| UpdateAllFields | — | void | Обновляет все текстовые поля формы значениями по умолчанию из параметров. |
| button\_Build\_Click | sender: object, e: EventArgs | void | Отображает диалоговое окно с перечнем ошибок валидации. |
| ShowValidationErrors | errors: Dictionary<ParameterType, string> | void | Обновляет метку ошибки для указанного параметра. |

Окончание таблицы 3.3 – Используемые методы класса MainForm

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| GetParameterDisplayName | type: ParameterType | string | Возвращает отображаемое имя параметра. |
| textBox\_LengthValue\_TextChanged | sender: object, e: EventArgs | void | Обработчик изменения текста в поле длины кирпича. |
| textBox\_WidthValue\_TextChanged | sender: object, e: EventArgs | void | Обработчик изменения текста в поле ширины кирпича. |
| textBox\_HeightValue\_TextChanged | sender: object, e: EventArgs | void | Обработчик изменения текста в поле высоты кирпича. |
| textBox\_RadiusValue\_TextChanged | sender: object, e: EventArgs | void | Обработчик изменения текста в поле радиуса отверстий. |
| textBox\_HoleCountValue\_TextChanged | sender: object, e: EventArgs | void | Изменения текста в поле количества отверстий. |

Таблица 3.4 – Используемые поля класса MainForm

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_parameters | BrickParameters | Объект, хранящий параметры кирпича. |
| \_builder | BrickBuilder | Объект, отвечающий за построение модели. |
| \_wrapper | Kompas3DWrapper | Объект-обёртка для взаимодействия с API КОМПАС-3D. |
| \_textBoxes | Dictionary<ParameterType, TextBox> | Словарь, связывающий типы параметров с текстовыми полями для ввода. |
| \_errorLabels | Dictionary<ParameterType, Label> | Словарь, связывающий типы параметров с метками для отображения ошибокю |

Таблица 3.5 – Используемые поля класса BrickParameters

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| EdgeMarginConstant | const double | Константа для вычисления минимального отступа от края (5 мм). |
| MinimumDistanceFactor | const double | Коэффициент для вычисления минимального расстояния между отверстиями (0.25). |
| MinimumDistanceConstant | const double | Минимальное абсолютное расстояние между отверстиями (5 мм). |

Окончание таблицы 3.5 – Используемые поля класса BrickParameters

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_parameters | Dictionary<ParameterType, Parameter> | Словарь, хранящий объекты параметров. |
| \_length | double | Приватное поле для хранения длины кирпича. |
| \_width | double | Приватное поле для хранения ширины кирпича. |
| \_height | double | Приватное поле для хранения высоты кирпича. |
| \_radius | double | Приватное поле для хранения радиуса отверстий. |
| \_holesCount | int | Приватное поле для хранения количества отверстий. |

Таблица 3.6 – Используемые методы класса BrickParameters

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| BrickParameters | — | — | Конструктор. Создаёт все параметры с значениями по умолчанию и вычисляет зависимые ограничения. |

Продолжение таблицы 3.6 – Используемые методы класса BrickParameters

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| GetParameter | type: ParameterType | Parameter | Возвращает объект параметра по его типу. |
| ValidateParameter | type: ParameterType, errorMessage: out string | bool | Выполняет валидацию указанного параметра. Возвращает true, если параметр корректен. |
| ValidateAll | errors: out Dictionary<ParameterType, string> | bool | Выполняет валидацию всех параметров. Возвращает true, если все параметры корректны. |
| CreateAllParameters | — | void | Создаёт словарь параметров с начальными значениями и диапазонами. |
| CalculateMaximumHolesCount | — | int | Вычисляет максимум отверстий. |

Окончание таблицы 3.6 – Используемые методы класса BrickParameters

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| RecalculateMaxRadius | — | void | Пересчитывает максимально допустимый радиус отверстий (не более 25% от ширины кирпича). |
| ValidateParameter | — | void | Пересчитывает максимальное количество отверстий при изменении размеров кирпича или радиуса. |

Таблица 3.7 – Используемые поля класса Kompas3Dwrapper

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| \_application | object | Объект приложения КОМПАС-3D. |
| \_document | object | Объект текущего документа. |
| \_part | object | Объект детали. |
| \_currentSketch | object | Объект текущего эскиза. |

Таблица 3.8 – Используемые методы класса Kompas3DWrapper

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Kompas3DWrapper | – | – | Конструктор. Создаёт обёртку для работы с API КОМПАС-3D. |
| StartKompas | – | void | Запускает приложение КОМПАС-3D или подключается к уже запущенному. |
| CreateDocument | – | void | Создаёт новый 3D-документ (деталь). |
| CreatePart | – | void | Получает объект детали для работы. |
| CreateSketch | planeType: int | void | Создаёт новый эскиз на указанной плоскости. |
| EndSketch | – | void | Завершает редактирование эскиза. |
| IsApplicationRunning | – | bool | Возвращает true, если приложение КОМПАС-3D запущено и доступно |

Окончание таблицы 3.8 – Используемые методы класса Kompas3DWrapper

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| CreateRectangle | x1: double, y1: double, x2: double, y2: double | void | Создаёт прямоугольник в эскизе по двум противоположным углам. |
| CreateCircle | centerX: double, centerY: double, radius: double | void | Создаёт окружность в эскизе с заданным центром и радиусом. |
| ExtrudeSketch | depth: double, forward: bool | void | Выполняет операцию выдавливания эскиза на заданную глубину. |
| CutExtrude | depth: double, throughAll: bool | void | Выполняет операцию вырезания выдавливанием |
| SelectPlane | planeType: int | void | Выбирает базовую плоскость для построения |
| CreateOffsetPlane | offset: double | void | Создаёт смещённую плоскость на заданное расстояние от базовой |

# Макеты пользовательского интерфейса

На рисунке 3.2 представлен ввод параметров с единицами измерений и подсказками по допустимым диапазонам. Кнопка «Построить кирпич».

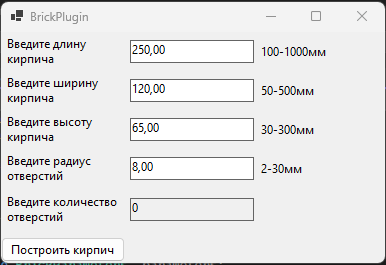


Рисунок 3.2 – Макет UI.

На рисунке 3.3 представлена ошибка валидации и подсвечивание поля в месте ошибки с последующим изменеием текста рядом с полем, а так же ошибка при преднамеренном нажатии кнопки «Построить кирпич» с неправильными параметрами.

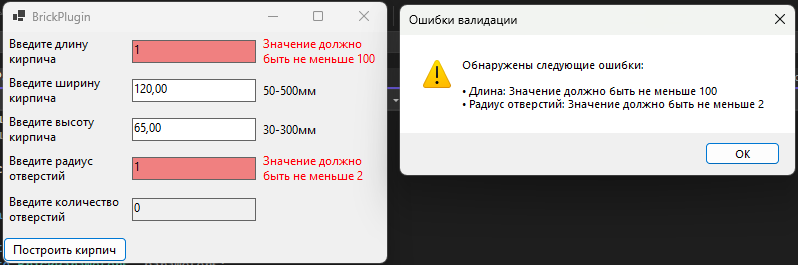


Рисунок 3.3 – Пример сообщения об ошибке.

4 Список источников

1. КОМПАС-3D [Электронный ресурс]. − Режим доступа https://kompas.ru/kompas-3d/about/ (дата обращения 13.10.2025)
2. API [Электронный ресурс]. − Режим доступа https://habr.com/ru/articles/464261/ (дата обращения 13.10.2025)
3. UML [Электронный ресурс]. − Режим доступа https://www.uml-diagrams.org/ (дата обращения 13.10.2025)

4. Программа автоматического построения 3D моделей и разверток по заданным значениям в AutoCAD «Лекало». Расчет и построение механических передач [Электронный ресурс]. – Режим доступа: https://www.2d-3d.ru/3d-galereia/autocad/811-programma-dlya-autocad-lekalo.html (дата обращения: 13.10.2025)

5. Болты, гайки, шайбы... Быстрое моделирование [Электронный ресурс]. – Режим доступа: https://3dtoday.ru/blogs/3dlab/bolts-nuts-washers-quick-.modeling (дата обращения: 13.10.2025)