Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**РАЗРАБОТКА ПЛАГИНА “ВОЗДУШНЫЙ ВИНТ” ДЛЯ САПР КОМПАС 3D**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР (ОРСАПР)»

Выполнил:  
Студент гр. 580-1

\_\_\_\_\_\_\_\_\_\_\_\_/ Белошицкий А.П.

(подпись)

«\_\_\_» \_\_\_\_\_\_\_\_ 2023г.

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

\_\_\_\_\_\_\_\_\_\_\_\_ / Калентьев А.А.

(подпись)

«\_\_\_» \_\_\_\_\_\_\_\_ 2023г.

# Оглавление

[Оглавление 1](#_Toc788153484)

[1 Описание САПР 2](#_Toc1694143596)

[1.1 Информация о выбранной САПР 3](#_Toc1288236784)

[1.2 Описание API 4](#_Toc1348515865)

[1.3 Обзор аналогов 8](#_Toc863792039)

[2 Описание предмета проектирования 9](#_Toc1862462247)

[3 Проект системы 11](#_Toc1903258118)

[3.1 Диаграмма классов 12](#_Toc1618891531)

[3.2 Макеты пользовательского интерфейса 13](#_Toc1144402060)

[Список использованных источников 14](#_Toc1435003588)

# 1 Описание САПР

## 1.1 Информация о выбранной САПР

Проектирование новых видов и образцов машин, оборудования, устройств, аппаратов, приборов и других изделий представляет сложный и длительный процесс, включающий в себя разработку исходных данных, чертежей, технической документации, необходимых для изготовления опытных образцов и последующего производства, и эксплуатации объектов проектирования.

***Проектирование*** — это комплекс работ с целью получения описаний нового или модернизируемого технического объекта, достаточных для реализации или изготовления объекта в заданных условиях. В процессе проектирования возникает необходимость создания описания, необходимого для построения еще не существующего объекта. Получаемые при проектировании описания бывают окончательными или промежуточными. Окончательные описания представляют собой комплект конструкторско-технологической документации в виде чертежей, спецификаций, программ для ЭВМ и автоматизированных комплексов и т.д. [1]

Основной целью автоматизации является повышение качества исполнения процесса. Автоматизированный процесс обладает более стабильными характеристиками, чем процесс, выполняемый в ручном режиме. Во многих случаях автоматизация процессов позволяет повысить производительность, сократить время выполнения процесса, снизить стоимость, увеличить точность и стабильность выполняемых операций.

***Система автоматизированного проектирования (САПР)***— это организационно-техническая система, состоящая из комплекса средств автоматизации проектирования, взаимодействующего с подразделениями проектной организации и выполняющая автоматизированное проектирование.

Для реализации плагина будет использоваться программа   
«КОМПАС-3D» версии 22.

***Компас-3D*** – это система трехмерного моделирования деталей и сборок, используемая для проектирования изделий в машиностроении и строительстве — от изделий народного потребления до авиа-, судостроения и продукции военного назначения. [2]

Система «КОМПАС-3D» была выбрана потому-что, отличается проектированием изделий любой сложности, простотой освоения, бесплатной технической поддержкой, автоматизацией отраслевых задач и многим другим. Бесплатная учебная версия распостряется и ее довольно легко получить, в отличии от других импортных САПР.

В качестве аналога «КОМПАС-3D» в котором можно разработать подобный плагин можно привести «Autodesk Inventor». Данная САПР имеет схожий функционал и интерфейс, благодаря чему при переходе между этими САПР не должны возникать проблемы с освоением системы.

## 1.2 Описание API

Аббревиатура ***API*** расшифровывается как «Application Programming Interface» (интерфейс программирования приложений, программный интерфейс приложения). [3]

Сегодня встречаются задачи, решение которых не реализованы в CAD-системах. Чаще всего это очень узкоспециализированные задачи, которые встречаются на каком-то конкретном предприятии или подотрасли. Для решения подобных задач и их автоматизации используется API.

***API КОМПАС-3D*** — это ориентированные на прикладного программиста инструментальные средства разработки приложений (библиотек конструктивов, прикладных САПР) на базе системы КОМПАС. API КОМПАС-3D включает в свой состав API 5 и API 7.

Главным интерфейсом API системы КОМПАС является ***KompasObject.***[4] Методы этого интерфейса, реализуют наиболее общие функции работы с документами системы, системными настройками, файлами, а также дают возможность получить указатели на другие интерфейсы (интерфейсы динамического массива, работы с математическими функциями, библиотек моделей или фрагментов и различных структур параметров определенного типа).

Ниже в таблице 1.1 представлены основные свойства и методы интерфейса KompasObject.

Таблица 1.1 — Методы и свойства интерфейса KompasObject

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Document3D() |  | ksDocument | Метод для получения указателя на интерфейс трехмерного графического документа (детали или сборки) |
| GetParamStruct(short structType) | structType – тип интерфейса параметров | StructType2D | Метод для получения указателя на интерфейс графического документа (чертежа или фрагмента) |
| Visible |  | bool | Свойство видимости приложения |
| Quit() |  |  | Метод для закрытия активного окна приложения КОМПАС |

В таблице 1.2 представлены методы интерфейса ksEntity, необходимые для разработки плагина.

Таблица 1.2 — Методы и свойства интерфейса ksEntity

|  |  |  |
| --- | --- | --- |
| Название | Тип возвращаемых данных | Описание |
| Create() | bool | Создать объект в модели |
| GetDefinition() | IUnkown | Получить указатель на интерфейс параметров объектов и элементов |
| Update() | bool | Изменить свойства объекта (используя ранее установленные свойства) |

В таблице 1.3 представлены свойства и методы интерфейса ksDocument2D, необходимые для разработки плагина

Таблица 1.3 — Методы интерфейса ksDocument2D

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| ksRectangle(  ksRectangleParam param, int style) | param – параметры прямоугольника.  style – стиль линии. | int | Получить указатель на прямоугольник на двумерной плоскости либо 0 в случае ошибки |

В таблице 1.4 представлены свойства и методы интерфейса ksDocument3D, необходимые для разработки плагина.

Таблица 1.4 — Методы интерфейса ksDocument3D

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Create (bool invisible, bool \_typeDoc) | invisible – признак режима редактирования документа (true – невидимый режим, false –  видимый режим), typeDoc - тип документа (true- деталь, fasle- сборка) | bool | Создать документ-модель (деталь или сборку) |
| GetPart(int type) | type – тип компонента из перечисления Типы компонентов. | ksPart | Получить указатель на интерфейс компонента в соответствии с заданным типом |

В таблице 1.5 представлены методы интерфейса ksPart, необходимые для разработки плагина.

Таблица 1.5 — Свойства и методы интерфейса ksPart.

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| EntityCollection(short objType) | objType – тип объектов, содержащихся в массиве. | ksEnintyCollection | Формирует массив объектов и возвращает указатель на его интерфейс |

Продолжение таблицы 1.5

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| GetDefaultEntity(short objType) | objType – тип объекта | ksEntity | Получить указатель на интерфейс объекта, создаваемого системой по умолчанию |
| GetPart(int type) | Type – тип компонента | ksPart | Получить указатель на интерфейс компонента в соответствии с заданным типом |
| NewEntity(short objType) | objType – тип объекта | ksEntity | Создать новый интерфейс объекта и получить указатель на него |

В таблице 1.6 представлены типы объектов документа-модели, необходимые для разработки плагина.

Таблица 1.6 — Некоторые типы объектов документа-модели

|  |  |  |
| --- | --- | --- |
| Идентификатор объекта | Название объекта | Интерфейс параметров |
| o3d\_unknown | Неизвестный (включает все объекты) |  |
| o3d\_planeXOZ | Плоскость XOZ | ksPlaneParam |
| o3d\_planeYOZ | Плоскость YOZ | ksPlaneParam |
| o3d\_planeXOY | Плоскость XOY | ksPlaneParam |
| o3d\_sketch | Эскиз | ksSketchDefinition |
| o3d\_face | Грань | ksFaceDefinition |
| o3d\_baseExtrusion | Базовая операция выдавливания | ksBaseExtrusionDefinition |

## 

## 1.3 Обзор аналогов

Плагин ***OpenCutList by L'Air du Bois***

В качестве примера для аналога плагина “Воздушный винт” был выбран плагин “*Propeller”* для программы *“3D MAX 2020”* , которая тоже является программой для 3D-моделирования и дизайнинга. Сам плагин предоставляет из себя готовое решение с удобным способом задания параметров, которое является альтернативой нашего плагина, для создания воздушного винта, т.к. при некоторых параметрах лопастей можно получить воздушный винт.

Плагин доступен на официальном сайте компании “High Technology Craft, LLC” его можно скачать напрямую от туда. [5]

Устанавливается данный плагин непосредственно с официального сайта компании “High Technology Craft, LLC”. Интерфейс плагина представлен на рисунке 1.1.

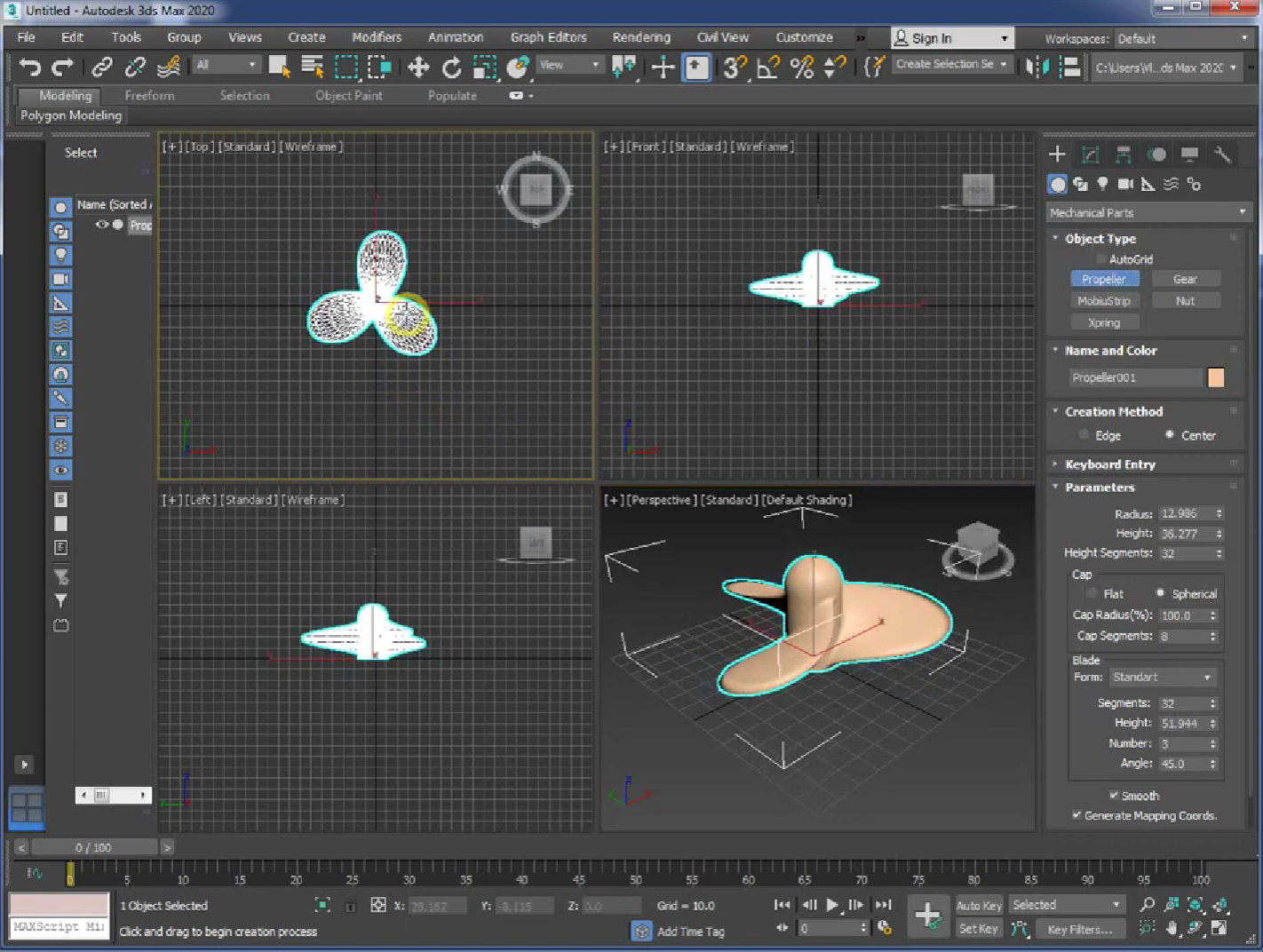


Рисунок 1.1 — Интерфейс плагина *Propeller*

# 2 Описание предмета проектирования

Предметом проектирования является Воздушный винт.

Воздушный винт (пропе́ллер) — лопастной движитель, создающий при вращении тягу за счёт отбрасывания воздуха назад с некоторой дополнительной скоростью, приводимый во вращение двигателем и преобразующий крутящий момент двигателя в силу тяги.

Основа воздушного винта - некое цилиндрическое тело, имеющее сквозное отверстие вдоль своей центральной оси, для крепления с валом двигателя.

К цилиндрической основе крепятся лопасти. Они имеют совершенно различную форму, оправдываемую применением винта.

У пользователя есть возможность изменять параметры в винте которые приведенные ниже:

* + ширина лопасти винта Влопасти (15 — 60мм);
  + длинна лопасти винта Lлопасти (100 — 300мм);
  + внутренний радиус окружности основания винта Rвнутр (От Lлопасти / 10, до Lлопасти / 2);
  + внешний радиус окружности основания винта Rвнеш (От Rвнутр + 10%, до Rвнутр \* 2);
  + форма лопасти (не менее двух вариантов на усмотрение разработчика);
  + количество лопастей на окружности основания (От 2, до 15).

На рисунке 2.1 показан общий вид воздушного винта:

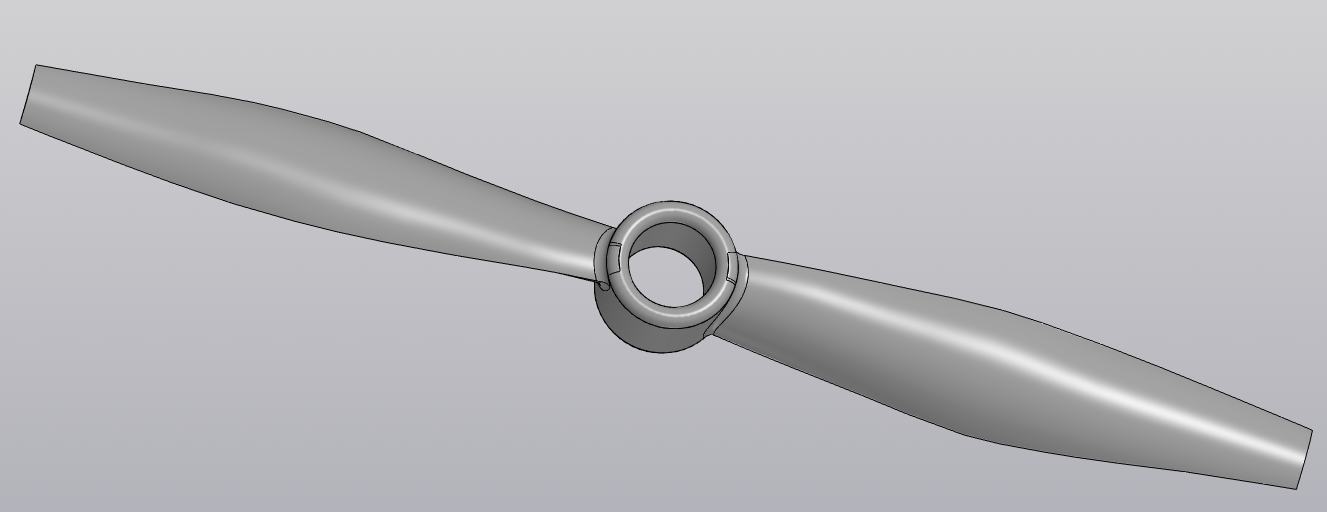


Рисунок 2.1 — Общий вид воздушного винта

На рисунке 2.3 представлен вид сверху с указанными параметрами:

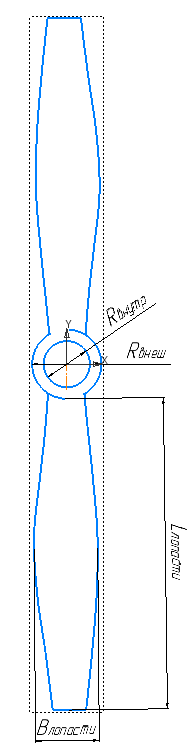


Рисунок 2.2 — Вид воздушного винта сверху

# 3 Проект системы

## 3.1 Диаграмма классов

Диаграмма классов определяет типы классов системы и различного рода статические связи, которые существуют между ними. На диаграммах классов изображаются также атрибуты классов, операции классов и ограничения, которые накладываются на связи между классами.[7] Целью создания диаграммы классов является графическое представление статической структуры элементов системы. Диаграмма классов плагина представлена на рисунке 3.1.

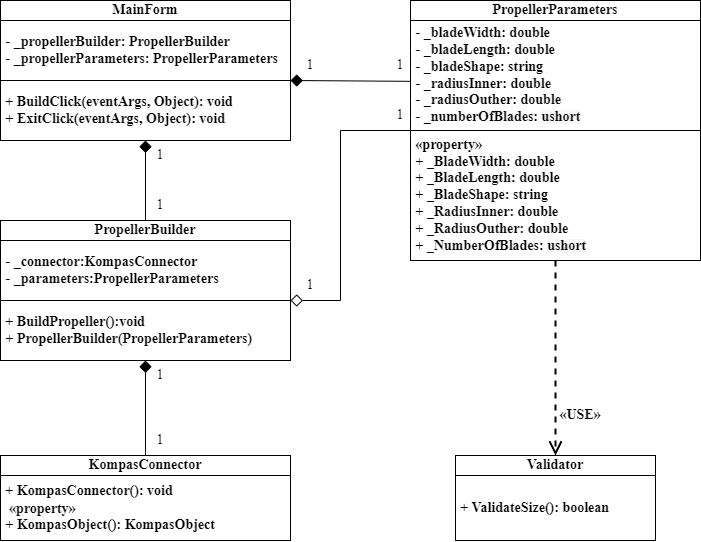


Рисунок 3.1 — Диаграмма классов плагина «Воздушный винт»

* MainForm является главным элементом управления для обработки действий в графическом интерфейсе;
* PropellerBuilder – выполняет построение детали;
* PropellerParameters – содержит в себе параметры воздушного винта, которые проверяются на правильность с помощью класса Validator;
* Validator – имеет метод для сравнивания параметра с максимальным и минимальным доступным значением;
* KompasConnector – класс связи с КОМПАС – 3D.

## 3.2 Макеты пользовательского интерфейса

Пользовательский интерфейс представляет собой форму для ввода параметров. На форме присутствует чертёж с параметрами для демонстрации параметров пропеллера и поля для ввода. Пользователь вводит значения самостоятельно, опираясь на подсказки, отображенные около полей. При нажатии на кнопку «Построить» проводится проверка зависимых параметров и, если условия соблюдены, строится 3D-модель воздушного винта. На рисунке 3.2 представлен макет пользовательского интерфейса.

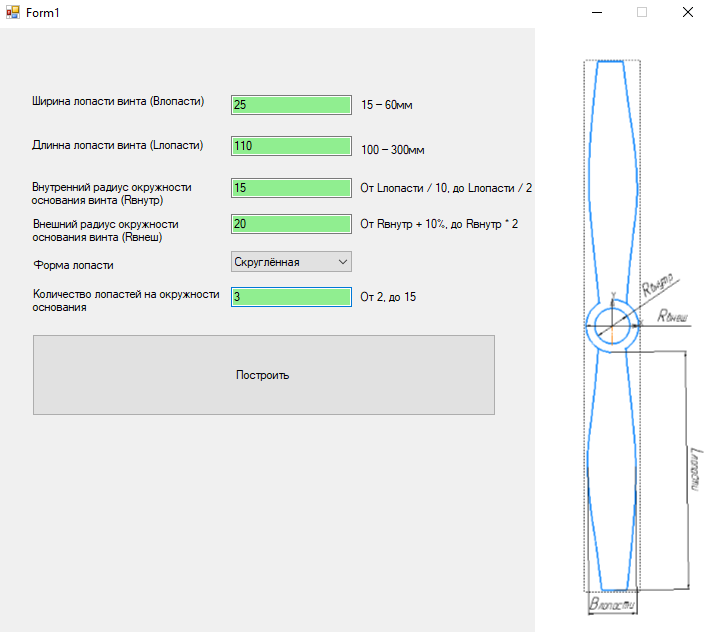


Рисунок 3.2 — Макет пользовательского интерфейса

Проверка правильности ввода значений проводится по ходу заполнения полей. Если поле заполнено неправильно, то есть пользователь ввел значение, превышающее границы, то оно подсвечивается красным цветом, сигнализирующем об ошибке.

Если же введены некорректные значения, и пользователь решил построить модель, несмотря на них, кнопка построения будет неактивна, пока не будут введены корректные значения (рисунок 3.3).

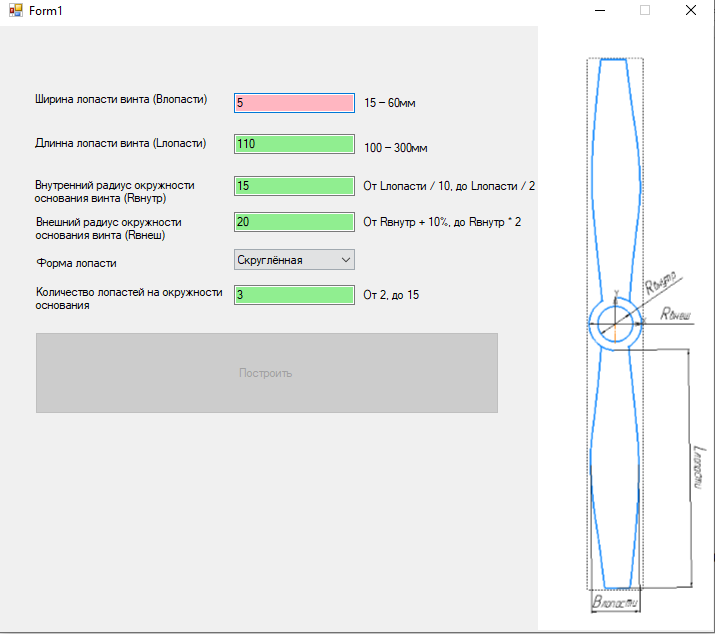


Рисунок 3.3 — Пример обработки ошибок при построении модели

# Список использованных источников

1. Общие сведения о САПР [Электронный ресурс]. – Режим доступа: http://www.hi-edu.ru/e-books/xbook116/01/part-002.htm, свободный (дата обращения: 06.11.2023).
2. Обзор популярных систем автоматического проектирования (CAD) [Электронный ресурс]. – Режим доступа: https://www.pointcad.ru/novosti/obzor-sistem-avtomatizirovannogo-proektirovaniya, свободный (дата обращения: 06.11.2023).
3. КОМПАС-3D: О программе [Электронный ресурс]. – Режим доступа: https://kompas.ru/kompas-3d/about/, свободный (дата обращения: 06.11.2023).
4. Что такое API? [Электронный ресурс]. – Режим доступа: https://dev.by/news/chto-takoe-api-prostym-yazykom, свободный (дата обращения: 06.11.2023).
5. apps.autodesk.com [Электронный ресурс]. — Режим доступа: https://apps.autodesk.com/3DSMAX/en/Detail/Index?id=8738716343823844227&appLang=en&os=Win64 (дата обращения 06.11.2023).
6. Воздушный винт [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/%D0%92%D0%BE%D0%B7%D0%B4%D1%83%D1%88%D0%BD%D1%8B%D0%B9\_%D0%B2%D0%B8%D0%BD%D1%82, свободный (дата обращения: 06.11.2023).
7. Фаулер М. UML. Основы. 3-е издание / М.Фаулер. – 3-е изд., пер. с англ. – СПб.: Символ-Плюс, 2019. – 192 с.