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**

**Элементы оглавления не найдены.**

# **1 Введение**

Автоматизация моделирования имеет огромное значение для развития науки, техники и производства в современном обществе. В настоящее время автоматизация – основной способ повышения производительности и эффективности труда инженерно-технических работников, занимающихся моделированием сложных устройств. Использование автоматизации в проектировании позволяет создавать все более сложные технические объекты и гибко реагировать на появление новых решений и технологий в той или иной области техники. Она позволяет значительно повысить точность расчетов, выбрать наилучшие варианты для реализации на основе строгого математического анализа всех или большинства вариантов проекта с оценкой технических, технологических и экономических характеристик производства и эксплуатации проектируемого объекта, значительно повысить качество конструкторской документации, существенно сократить сроки проектирования и передачи конструкторской документации в производство, эффективнее использовать технологическое оборудование с программным управлением [1].

Таким образом, целью данной работы является разработка плагина, автоматизирующего построение модели «Воздушный винт» для системы автоматизированного проектирования КОМПАС-3D с помощью интегрированной среды разработки Visual Studio 2022 Community [2]

Интегрированная среда разработки Visual Studio — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения.

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. [3]

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.

# **2 Постановка и описание задачи**

В данном проекте была поставлена задача реализовать плагин для САПР (Компас 3D), на языке программирования C#. Объектом построения был воздушный винт, с гибко-изменяемыми параметрами.

Сроком реализации данного проекта был целый семестр, с начала сентября до конца декабря, где на разработку и тестирования плагина ушло больше месяца, все остальное время заняло проектирование, документация.

В ходе анализа реализации плагина были выявлены “подводные камни” в виде сложно читаемой документации по API выбранной САПР, а так же ошибки допущенные, как в проектирование зависимостей между параметрами, так и в допустимых пределах этих параметров. Из положительных сторон можно выделить что есть упоминание всех методов и свойств данной API, и отрицательных сторон можно отметить что нет явных примеров как правильно их использовать, и зачастую некоторые решения реализовывались “методом тыка”.

# **3 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 (в ходе разработки, рамки параметров были измены):

* + 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 3.1 показан общий вид воздушного винта:

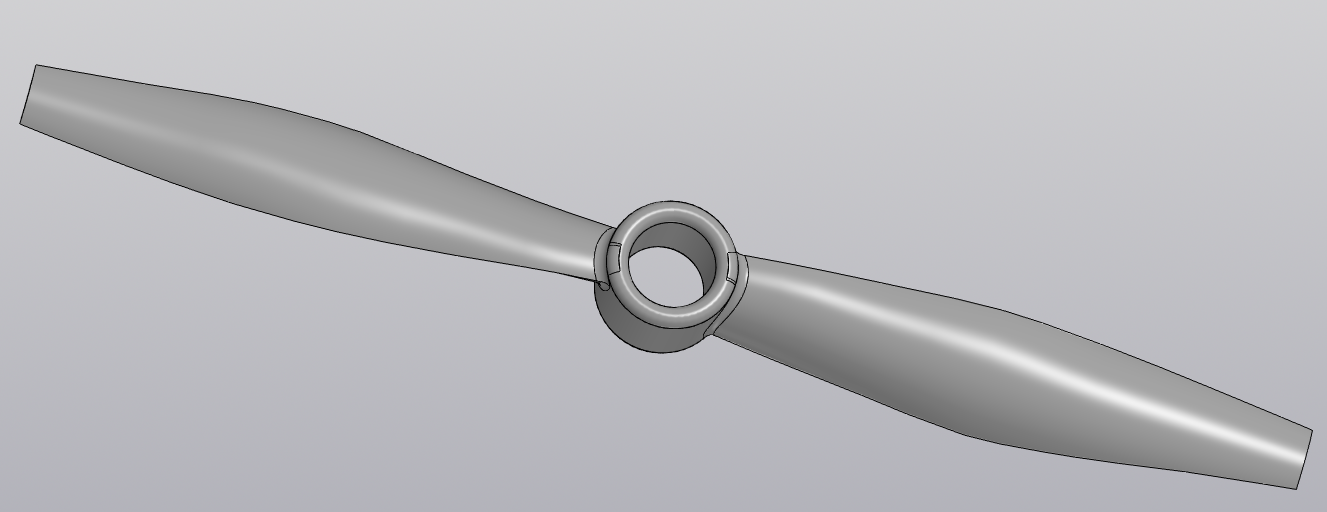


Рисунок 2.1 — Общий вид воздушного винта

На рисунке 2.3 представлен вид сверху с указанными параметрами:

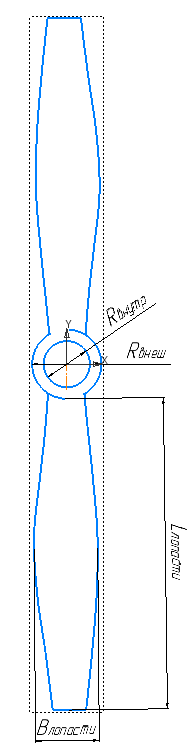


Рисунок 2.2 — Вид воздушного винта сверху

На рисунке 3.3 представлена модель стула с дополнительной функциональностью (круглое сидение, 3 круглые ножки вместо 4х):

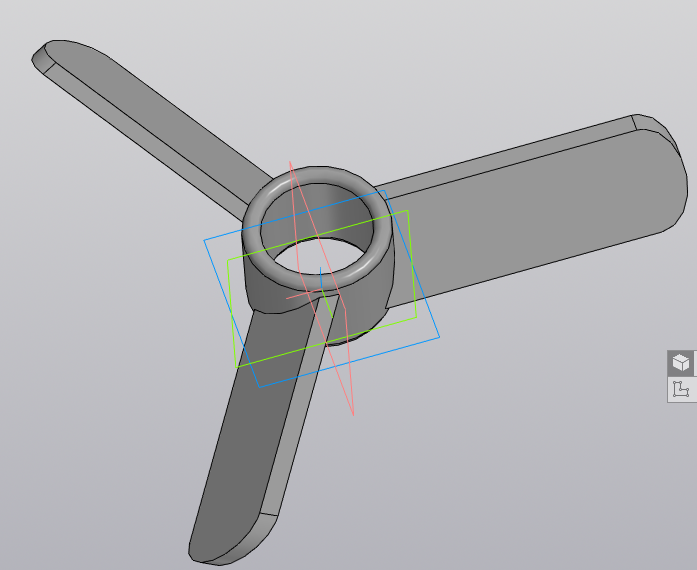


Рисунок 3.3 — Общий вид воздушного винта

с элептической формой лопастей

На рисунке 3.4 представлены вид спереди и вид сбоку с указанными параметрами:

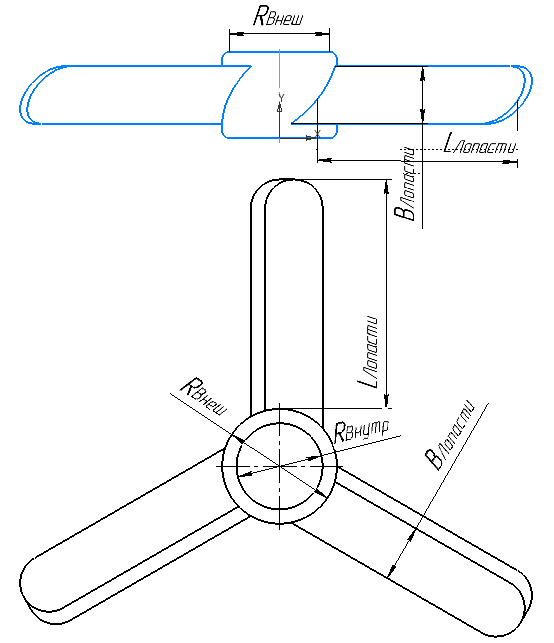


Рисунок 3.4 — Виды стула: сбоку, сверху

# **4 Выбор инструментов и средств реализации**

На основе требований к техническому заданию программа выполнена на языке программирования C# в среде Microsoft Visual Studio 2022 с использованием .NET Framework 4.7.2 [5], библиотека «Kompas6API5» [6] для основных операций в САПР КОМПАС-3D.

Инструментом тестирования и создания модульных тестов был выбран тестовый фреймворк NUnit [7] версии 3.13.2.

Технология разработки графического интерфейса: Windows Forms [8].

Были также использованы сторонние библиотеки:  
 StyleCop – для проверки правильности написания кода.

ReSharper или Spell Checker – проверка орфографии написанного кода.

Editor Guidelines – добавляет ограничительные линии на рабочее пространство, для соблюдения красивого оформления кода (чтобы не был слишком длинным).

Fine Code Coverage – показывает покрытие кода юнит тестами[9].

# **5 Назначение плагина**

Назначение разрабатываемого плагина обусловлено быстрым моделированием воздушного винта, с различными параметрами. Благодаря данному расширению, проектировщики и дизайнеры могут наглядно рассмотреть спроектированную модель, при необходимости перестроить под необходимые им параметры или просчитать характеристики различных форм лопастей, и принять решение какую использовать, в случае. Если ни одна форма не подошла, то в ручном режиме (уже в САПР) редактировать форму лопасти, что ускорит работу, т.к. основные компоненты уже будут в созданной детали.

# **6 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 6.1.

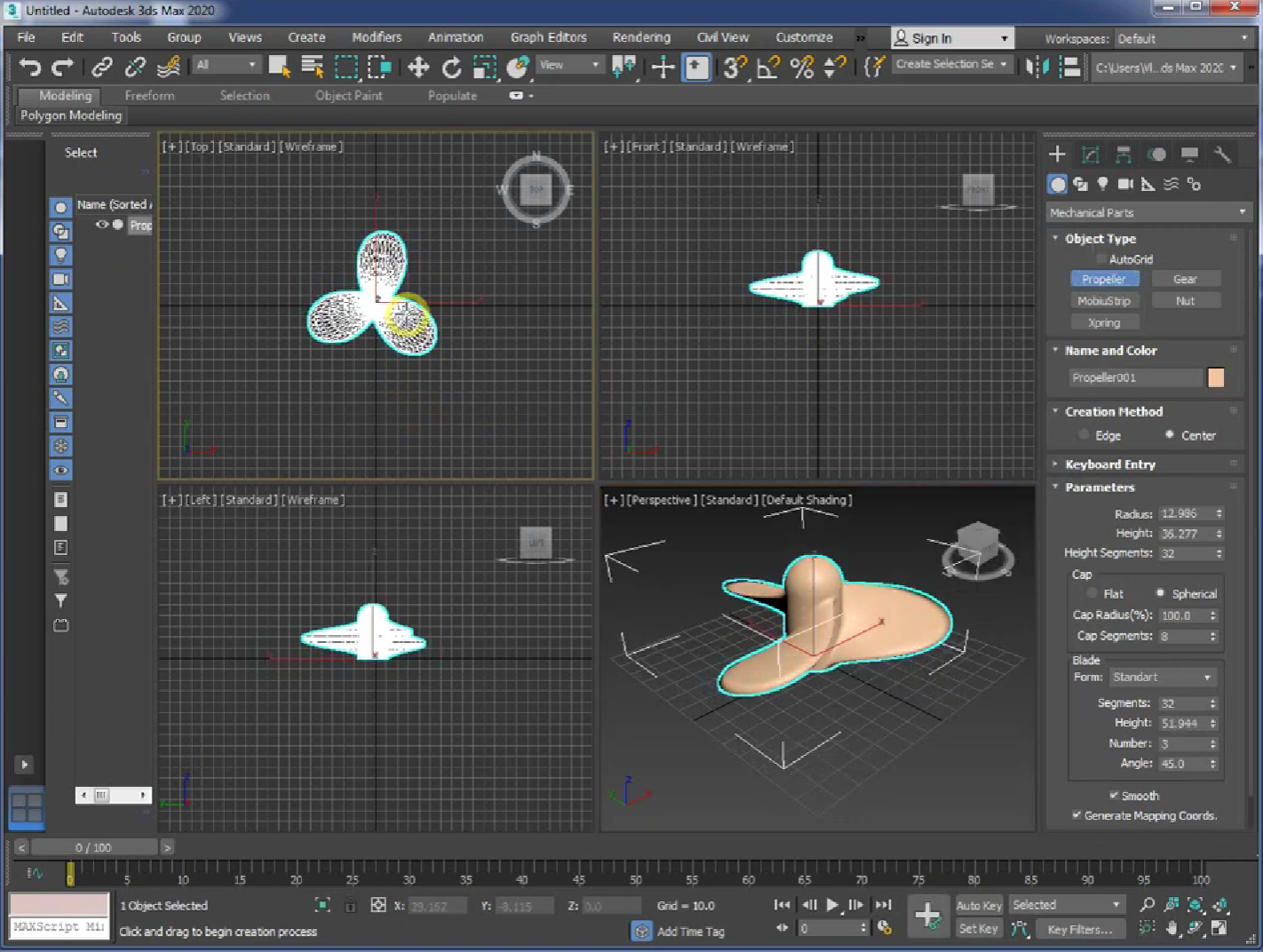


Рисунок 6.1 — Интерфейс плагина *Propeller*

# **7 Описание реализации**

Для графического описания абстрактной модели проекта, а также пользовательского взаимодействия (сценарии действия) использован стандарт UML.

UML язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это – открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML – моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML возможна генерация кода и наоборот.[11]

При использовании UML были построена 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. Изначальная диаграмма классов плагина представлена на рисунке 7.1.

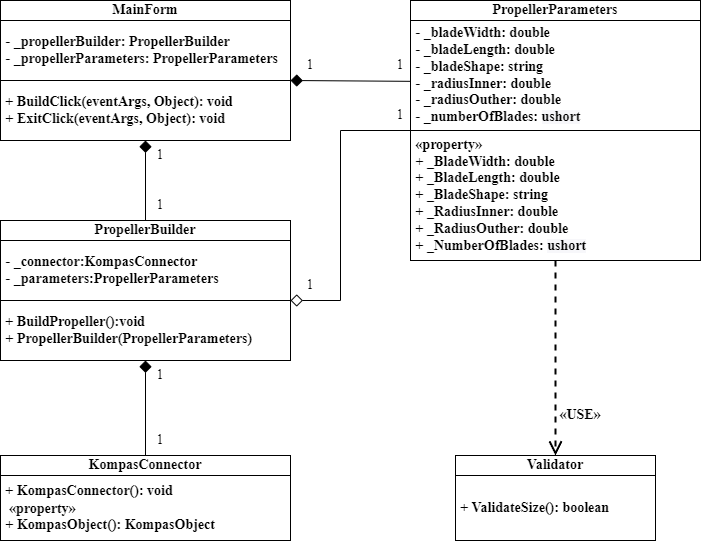


Рисунок 3.1 — Диаграмма классов плагина «Воздушный винт»

Т.к. в ходе работы над проектом, тема проекта была уточнена, соответсвенно изменился и нейминг компонентов (составляющих частей плагина) - новые названия составляющих указанны в скобках:

* MainForm является главным элементом управления для обработки действий в графическом интерфейсе;
* PropellerBuilder(AirScrewBuilder) – выполняет построение детали;
* PropellerParameters(AirScrewParameters) – 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, а так же метод проверки корректности пользовательского ввода;
* KompasConnector(Kompas3DWrepper) – класс связи с КОМПАС – 3D.

В итоговом проекте были дополнены классы и методы, которые отображены на итоговой диаграмме классов (рисунок 7.2).

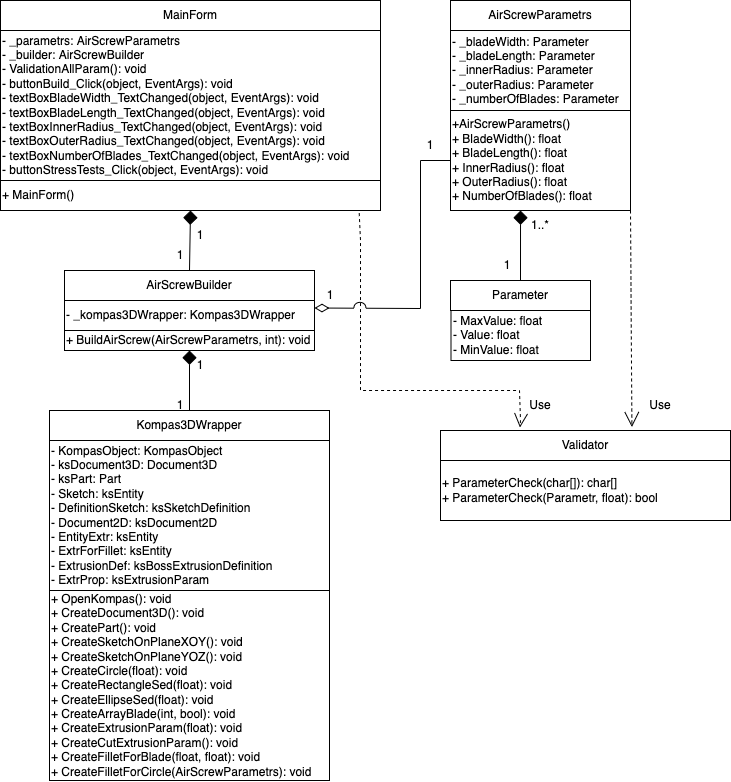


Рисунок 7.2 — Итоговая диаграмма классов плагина «Воздушный винт»

В процессе разработки изначальная задумка реализации классов была сохранена, были лишь дополнены классы методами, а также добавились перечисления для уменьшения дублирования кода.

# **8 Описание программы для пользователя**

Пользовательский интерфейс представляет собой форму для ввода параметров. На форме присутствует чертёж с параметрами для демонстрации параметров воздушного винта и поля для ввода. Пользователь вводит значения самостоятельно, опираясь на подсказки, отображенные около полей. Кнопка “ПОСТРОИТЬ” - находится в невктивном, по умолчанию, состоянии, и только при введение верхыпараметров становится активной. При начатии на активную кнопку, открывается, если не была ранее открыта, САПР и строится 3D-модель воздушного винта. На рисунке 8.1 представлен макет пользовательского интерфейса.

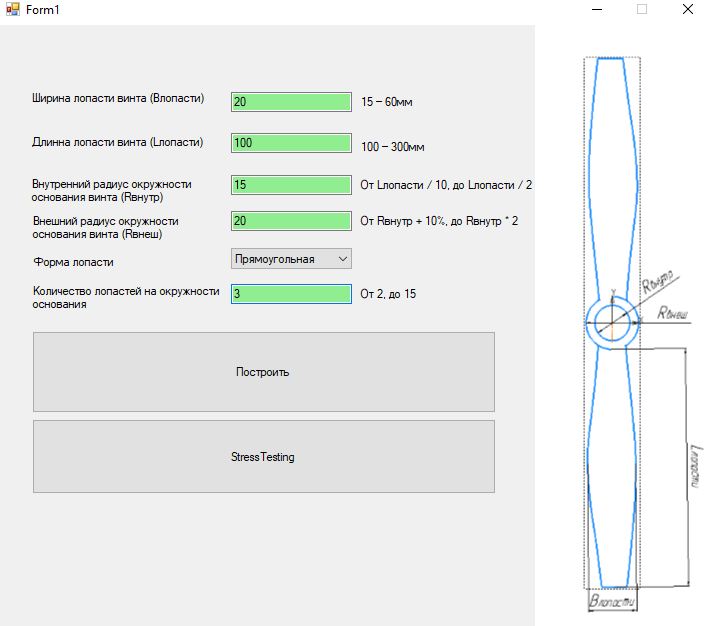


Рисунок 8.1 — Макет пользовательского интерфейса

Проверка правильности ввода значений проводится по ходу заполнения полей. Если введённый в поле символ является недопустимым - он стирается и кустор сбрасывается в начало строки, не стирая ранее введёное число.

Если же введены некорректные значения, и пользователь решил построить модель, несмотря на них, кнопка построения будет неактивна, пока не будут введены корректные значения (рисунок 8.2).

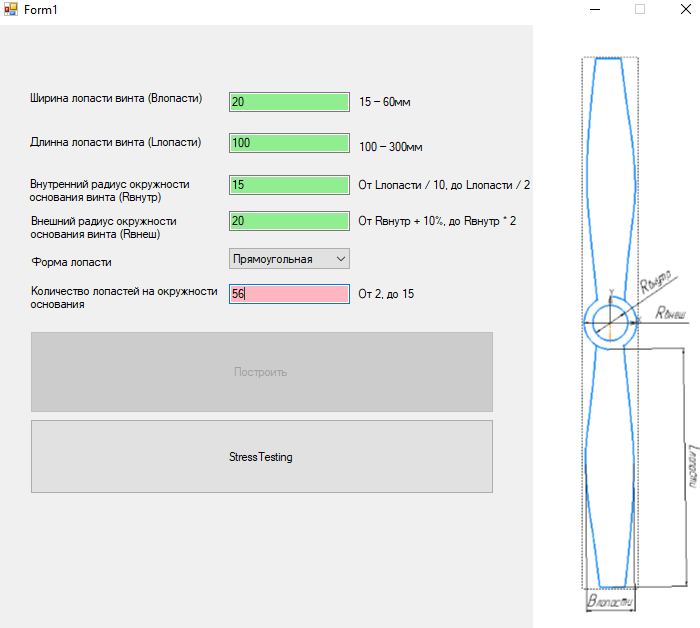


Рисунок 8.2 — Пример обработки ошибок при построении модели

Был также реализован дополнительный функционал воздушного винта с элептической формой лопастей (при выборе в comboBox “Эллептическая”). Будет строиться модель с скруглённой формой лопастей (Рисунок 8.3).

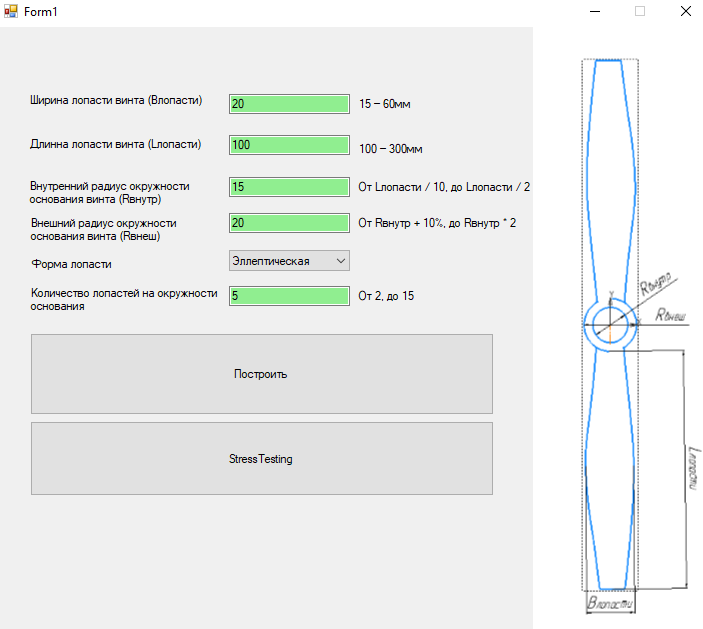


Рисунок 8.3 — Дополнительная функциональность

# **9 Тестирование плагина**

Тестирование позволяет убедиться в работоспособности программы, выявлять ошибки при изменении какого-либо функционала.

## **9.1 Функциональное тестирование**

При функциональном тестировании проверяется корректность работы плагина, а именно, соответствие полученного результата в виде трехмерной модели, с входными параметрами. [12]

Проведено тестирование максимальных и минимальных параметров модели.

Минимальные значения:

1. Ширина лопасти = 15 мм;
2. Длинна лопасти = 100 мм;
3. Внутренний радиус = 10 мм;
4. Внешний радиус = 11 мм;
5. Кол-во лопастей = 2 мм;

Максимальные значения:

1. Ширина лопасти = 15 мм;
2. Длинна лопасти = 100 мм;
3. Внутренний радиус = 10 мм;
4. Внешний радиус = 11 мм;
5. Кол-во лопастей = 2 мм;

На рисунке 9.1 представлена модель с минимально введенными параметрами.

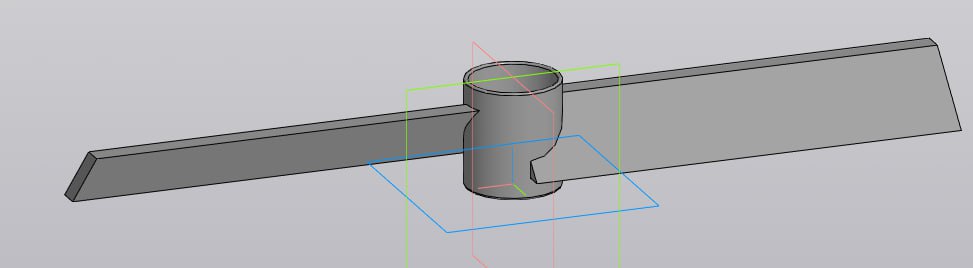


Рисунок 9.1 – Модель воздушного винта с минимальными введенными параметрами

На рисунке 9.2 представлена модель с максимальными введенными параметрами.

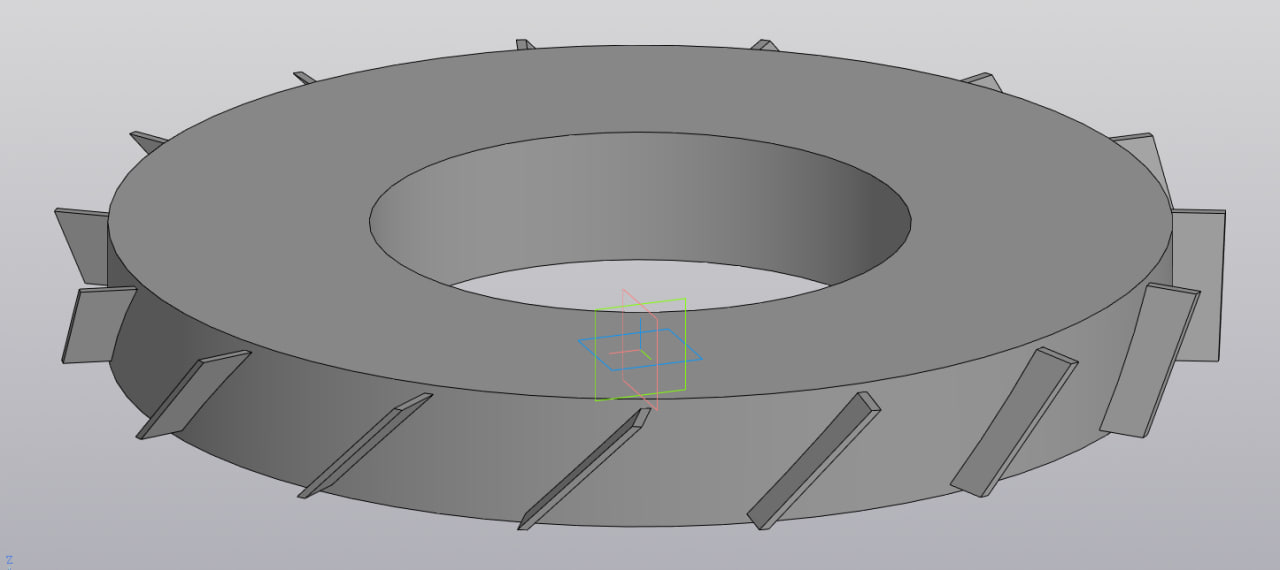


Рисунок 9.2 – Модель воздушного винта с максимальными введенными параметрами

На рисунке 9.3 представлена модель воздушного винта с элептической формой лопастей с максимальными введенными параметрами.

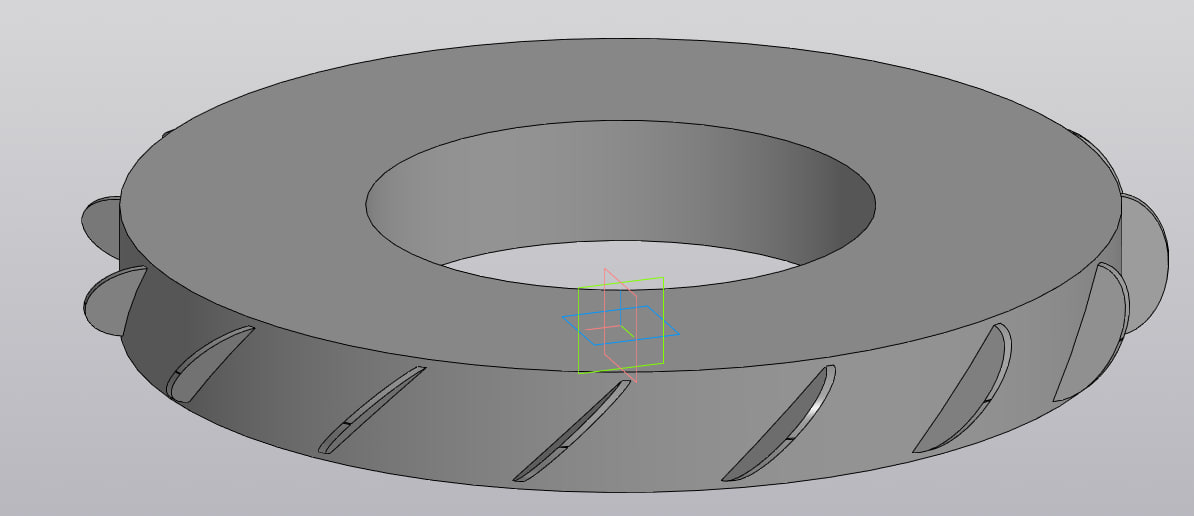


Рисунок 9.3 – Модель круглого стула с максимальными введенными параметрами

## **9.2 Модульное тестирование**

В целях проверки корректности работы методов и свойств классов при помощи тестового фреймворка NUnit версии 3.5 проведено модульное тестирование, проверялись открытые поля, свойства и методы. Были протестированы классы модели: ChairParameters, Validator. На рисунке 9.4 представлена информация о модульном тестировании программы. Покрытие модели тестами составило сто процентов, что показано на рисунке 9.5.

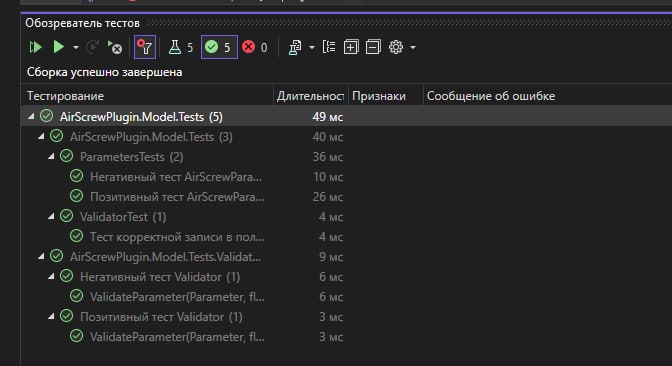


Рисунок 9.4 – Модульное тестирование плагина

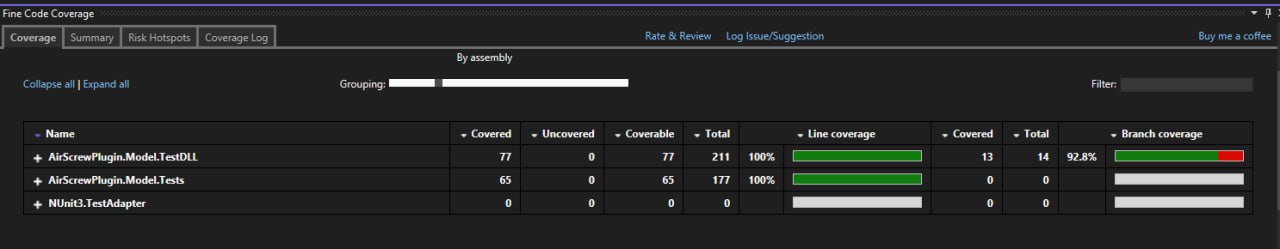


Рисунок 9.5 – Покрытие кода тестами

## **9.3 Нагрузочное тестирование**

В целях проверки производительности работы плагина, было проведено нагрузочное тестирование. Тестирование производилось на ноутбуке со следующей конфигурацией:

– CPU AMD Ryzen 5 3500U 2.10GHz;

– 8 ГБ ОЗУ (Доступно 6.9 Гб);

– GPU Radeon Vega 8 объемом памяти 2 ГБ.

– ОС Windows 10 Home х64.

Для нагрузочного тестирования был задан бесконечный цикл построения детали. Для измерения времени был использован класс Stopwatch. Тестирование заключалось в построении воздушного винта с минимальными параметрами. На рисунке 9.6 представлен график зависимости времени от количества построенных деталей.

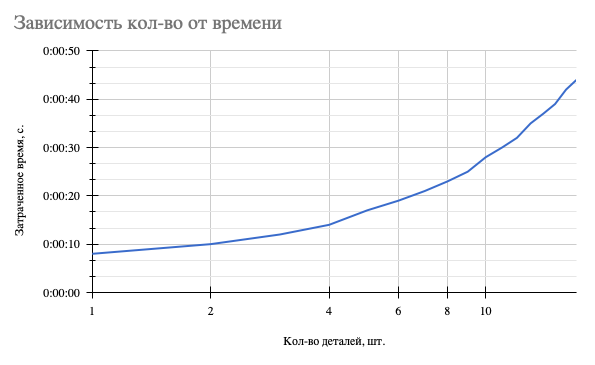


Рисунок 9.6 – График зависимости времени от построения модели

Из графика 9.6 видно, что время построения моделей изменяется линейно, но ближе к концу графика идет как экспонента. Так как график будет выглядеть как экспонента, была применена логарифмическая сетка для большей наглядности. Это связано с увеличением нагрузки на оперативную память и центральный процессор с каждым построением модели. За счет этого время построения модели увеличивается.

На рисунке 9.7 представлен график зависимости загруженности памяти от количества построенных деталей



Рисунок 9.7 – График зависимости загруженности памяти

Тестирование чуть менее двух минут, было построено семнадцать моделей воздушного винта со минимальными параметрами, работа плагина завершилась сообщением об ошибки памяти в Visual Studio и вылетом графического видеодрайвера AMD Software: Adrenalin edition. Исходя из графика, представленного на рисунке 9.6 можно увидеть, что построения детали в основном занимает менее десяти секунд, но после построения десятой детали, время построения некоторых деталей стало возрастать, данную тенденцию можно соотнести с самым большим изменением занимаемой памяти показанное на рисунке 9.7. Резкие уменьшения потребления памяти скорее всего обусловлены тем, что КОМПАС-3D подключает алгоритмы оптимизации при достижении максимальной занимаемой оперативной памяти компьютера.

Так же немаловажным фактором является использование файла подкачки, а точнее переход в виртуальную память. Виртуальная память — метод [управления памятью](https://ru.wikipedia.org/wiki/%D0%A3%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C%D1%8E) [компьютера](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80), позволяющий выполнять программы, требующие больше [оперативной памяти](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C), чем имеется в компьютере, путем автоматического перемещения частей программы между основной памятью и вторичным хранилищем (например, [твердотельным](https://ru.wikipedia.org/wiki/%D0%96%D1%91%D1%81%D1%82%D0%BA%D0%B8%D0%B9_%D0%B4%D0%B8%D1%81%D0%BA) накопителем)[13].

Также было решено провести второе нагрузочное тестирование в процессе которого будут создаваться модели воздушного винта с максимальными параметрами, в процессе построения, которых используются построения большего размера и в теории должны показать более нагруженный результат.

На рисунках 9.8 и 9.9 показан результат данного тестирования:



Рисунок 9.8 – График зависимости времени

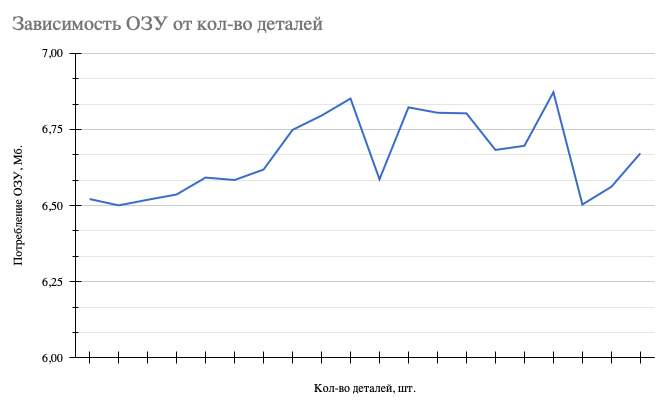


Рисунок 9.9 – График зависимости загруженности памяти

В данном тесте можно заметить такую же тенденцию линейного роста графика в зависимости по времени, однако ноутбук построил 20 деталей, что на 3 деталь больше, правда и заняло у него это около 70 секунд. Ситуация с памятью аналогичная, явно прослеживаются падения, связанные с файлом подкачки и выделения виртуальной памяти.

# **10 Заключение**

При выполнения лабораторных работ были изучены основные этапы проектирования программного продукта и его реализации, предметная область объекта проектирования, SDK «КОМПАС-3D». Было составлено техническое задание, разработан проект системы, составлены UML диаграммы классов, разработан макет пользовательского интерфейса.

В результате разработки произошло уточнение темы, что внесло небольшие правки в масштабы параметров и незначительное изменение названий компонентов плагина, был разработан и реализован плагин для САПР «КОМПАС-3D», выполняющий построение 3D-модели воздушного винта по заданным параметрам.

Над реализованным плагином были проведены функциональное, модульное и нагрузочное тестирование на платформе Windows 10.

# **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: 19.12.2023).
2. Обзор популярных систем автоматического проектирования (CAD) [Электронный ресурс]. – Режим доступа: https://www.pointcad.ru/novosti/obzor-sistem-avtomatizirovannogo-proektirovaniya, свободный (дата обращения: 19.12.2023).
3. КОМПАС-3D: О программе [Электронный ресурс]. – Режим доступа: https://kompas.ru/kompas-3d/about/, свободный (дата обращения: 19.12.2023).
4. Стул [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Стул, свободный (дата обращения: 19.12.2023).
5. Visual Studio [Электронный ресурс]. – URL: https://visualstudio.microsoft.com/ru/ (дата обращения: 19.12.2023).
6. КОМПАС-3D для разработчиков. [Электронный ресурс]. – URL: https://kompas.ru/solutions/developers/ (дата обращения: 19.12.2023).
7. NUnit [Электронный ресурс]. – URL: https://nunit.org/ (дата обращения: 19.12.2023).
8. Введение в Windows Forms [Электронный ресурс]. – URL: https://metanit.com/sharp/windowsforms/1.1.php (дата обращения: 19.12.2023).
9. Инструмент Fine Code Coverage [Электронный ресурс]. – URL: <https://marketplace.visualstudio.com/items?itemName=FortuneNgwenya.FineCodeCoverage> (дата обращения 19.12.2023).
10. OpenCutList. [Электронный ресурс]. — Режим доступа: https://github.com/lairdubois/lairdubois-opencutlist-sketchup-extension#opencutlist-by-lair-du-bois (дата обращения 19.12.2023).
11. Фаулер М. UML. Основы. 3-е издание / М.Фаулер. – 3-е изд., пер. с англ. – СПб.: Символ-Плюс, 2019. – 192 с.
12. Функциональное тестирование [Электронный ресурс]. – URL: https://daglab.ru/funkcionalnoe-testirovanie-programmnogo-obespechenija/ (дата обращения: 19.12.2023).
13. Виртуальная память — Википедия. [Электронный ресурс] — Режим доступа: https://ru.wikipedia.org/wiki/Виртуальная память (дата обращения 19.12.2023)