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)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил:

студент гр. 588-2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_В.С. Швоев

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.А. Калентьев

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

**Оглавление**

[1 Введение 3](#_Toc90061417)

[2 Постановка и анализ задачи 4](#_Toc90061418)

[2.1 Описание предмета проектирования 5](#_Toc90061419)

[2.2 Выбор инструментов и средств реализации 6](#_Toc90061420)

[2.3 Назначение плагина 6](#_Toc90061421)

[3 Обзор аналогов 7](#_Toc90061422)

[4 Описание реализации 8](#_Toc90061423)

[4.1 Диаграмма классов 8](#_Toc90061424)

[5 Описание программы для пользователя 11](#_Toc90061425)

[6 Тестирование программы 14](#_Toc90061426)

[6.1 Функциональное тестирование 14](#_Toc90061427)

[6.2 Модульное тестирование 17](#_Toc90061428)

[6.3 Нагрузочное тестирование 18](#_Toc90061429)

[Заключение 21](#_Toc90061430)

[Список использованных источников 22](#_Toc90061431)

# 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 «Забор» для системы автоматизированного проектирования Inventor 2022 с помощью интегрированной среды разработки Visual Studio 2019 Сommunity. [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.

# 2 Постановка и анализ задачи

В рамках лабораторных работ в соответствии с технически заданием требовалось разработать плагин, который на основе входных параметров, интегрируя с системой Inventor 2022, строит модель «Забор. [3] Также нужно было сделать поддержку работы плагина для САПР Kompas 3D (дополнительное задание).[4] Необходимо чтобы плагин позволял задавать параметры по умолчанию, а также изменять входные параметры забора, такие как:

* длина забора;
* общая высота забора;
* глубина погружения столба;
* высота верхней части забора;
* ширина столбика;
* расстояние между нижними перегородками;
* расстояние между верхними перегородками.

# 2.1 Описание предмета проектирования

Предметом проектирования является забор. Забор (также городьба, изгородь, огорожа, ограда, палисад, плетень, тын, частокол) — сооружение, которое охватывает территорию, как правило, на открытом воздухе, и обычно состоящее из столбов и перекрытий из различных материалов, и служащее для ограждения (защиты) и обрамления (обозначения границы) той или иной территории.[5]

На рисунке 2.1 представлен чертеж забора.



Рисунок 2.1 – Чертеж забора

Параметры забора:

* длина забора ***L*** (1 — 3м);
* общая высота забора ***H*** (1.5 — 2м);
* глубина погружения столба ***h1*** (1/3 — 1/2 от общей высоты забора);
* высота верхней части забора ***h2***(1/2 — 2/3 от общей высоты забора);
* ширина столбика ***W*** (10-70мм);
* расстояние между нижними перегородками ***g1*** (не больше половины длины забора и не меньше ширины одного столбика. Количество столбиков будет определяться автоматически исходя из данного параметра и длины забора);

расстояние между верхними перегородками ***g2*** (не больше половины длины забора и не меньше ширины одного столбика. Количество столбиков будет определяться автоматически исходя из данного параметра и длины забора).

# 2.2 Выбор инструментов и средств реализации

На основе требований к техническому заданию программа выполнена на языке программирования C# в среде Microsoft Visual Studio 2019 с использованием .NET Framework 4.7.2 [2], библиотека «Inventor» [6] для основных операций в САПР Inventor, библиотеки для Kompas 3D [7].

Инструментом тестирования и создания модульных тестов был выбран тестовый фреймворк NUnit [8] версии 3.13.2.

Для реализации пользовательского интерфейса использовалась система для построения настольные приложения WPF [9].

# 2.3 Назначение плагина

Назначение разрабатываемого плагина обусловлено быстрым моделированием заборов разных типов. Благодаря данному расширению, мастера по заборам могут наглядно рассмотреть спроектированную модель, при необходимости перестроить под необходимые им параметры.

# 3 Обзор аналогов

**Плагин Instant Fence and Railing для Sketchup**

SketchUp — программа для 3D дизайна и архитектурного проектирования. В основном используется для моделирования жилых домов, мебели, интерьера. Есть инструменты для проектирования лестниц, электропроводки, санитарно-технических коммуникаций и оборудования.[10]

Данный плагин содержит более трехсот предустановленных стилей, разделенных на библиотеки. Пользователи могут создавать собственные стили и сохранять изображения и значки. Применяя этот плагин, пользователи могут моделировать все заборы или перила в 3D, 2D линиях, расположенных с основной стороны или 2d скрытых линиях.

Плагин генерирует множество моделей заборов, балконных перил, перил, перил. Пикеты, перила и ограждения могут быть расположены по центру или с выбранной стороны.[11]

Интерфейс плагина показан на рисунке 3.1.



Рисунок 3.1 — Интерфейс плагина Instant Fence and Railing

# 4 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.[12]

При использовании UML были простроена диаграмма классов.

# 4.1 Диаграмма классов

Диаграмма классов – структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей между ними.[12]

На рисунке 4.1 представлена диаграмма классов.

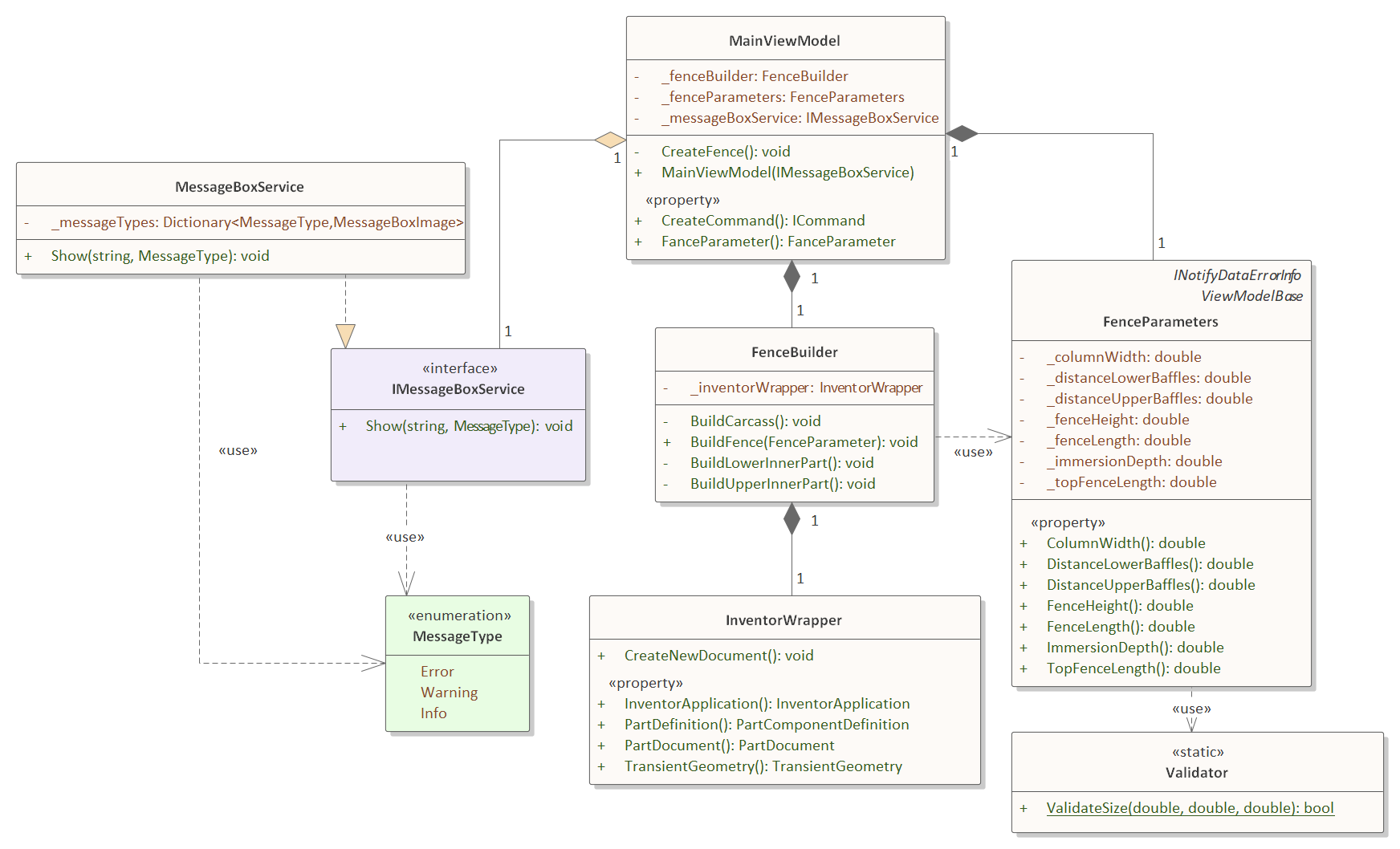


Рисунок 4.1 – Изначальная диаграмма классов

Главное окно связывается с MainViewModel для управления данными. MainViewModel содержит экземпляры следующие экземпляры классов:

* FenceParameter — содержит параметры забора, которые валидируются с помощью класса Validator;
* FenceBuilder — класс создания забора в Inventor, хранящий класс InventorApi, который содержит нужные экземпляры класса из всей Inventor API;
* IMessageBoxService — сервис для использования уведомляющих окон, реализация сервиса находится в классе MessageBoxService. Метод Show принимает в качестве одного из аргументов элемент перечисления MessageType, который отвечает за тип уведомляющего окна.

Будет использоваться дополнительная библиотека **MvvmLightLib** для более удобного использования паттерна MVVM. В данной библиотеке уже есть реализация таких интерфейсов, как:

* *INotifyPropertyChanged* — сообщает клиенту об изменении значения свойства;
* *ICommand* — определяет команду.

Также будет использоваться объект класса *ServiceProvider* из пространства имен *Microsoft.Extensions.DependencyInjection* для более простого и удобного внедрения сервисов.

В итоговом проекте созданы следующие классы и методы, которые отображены на итоговой диаграмме классов (рисунок 4.2).

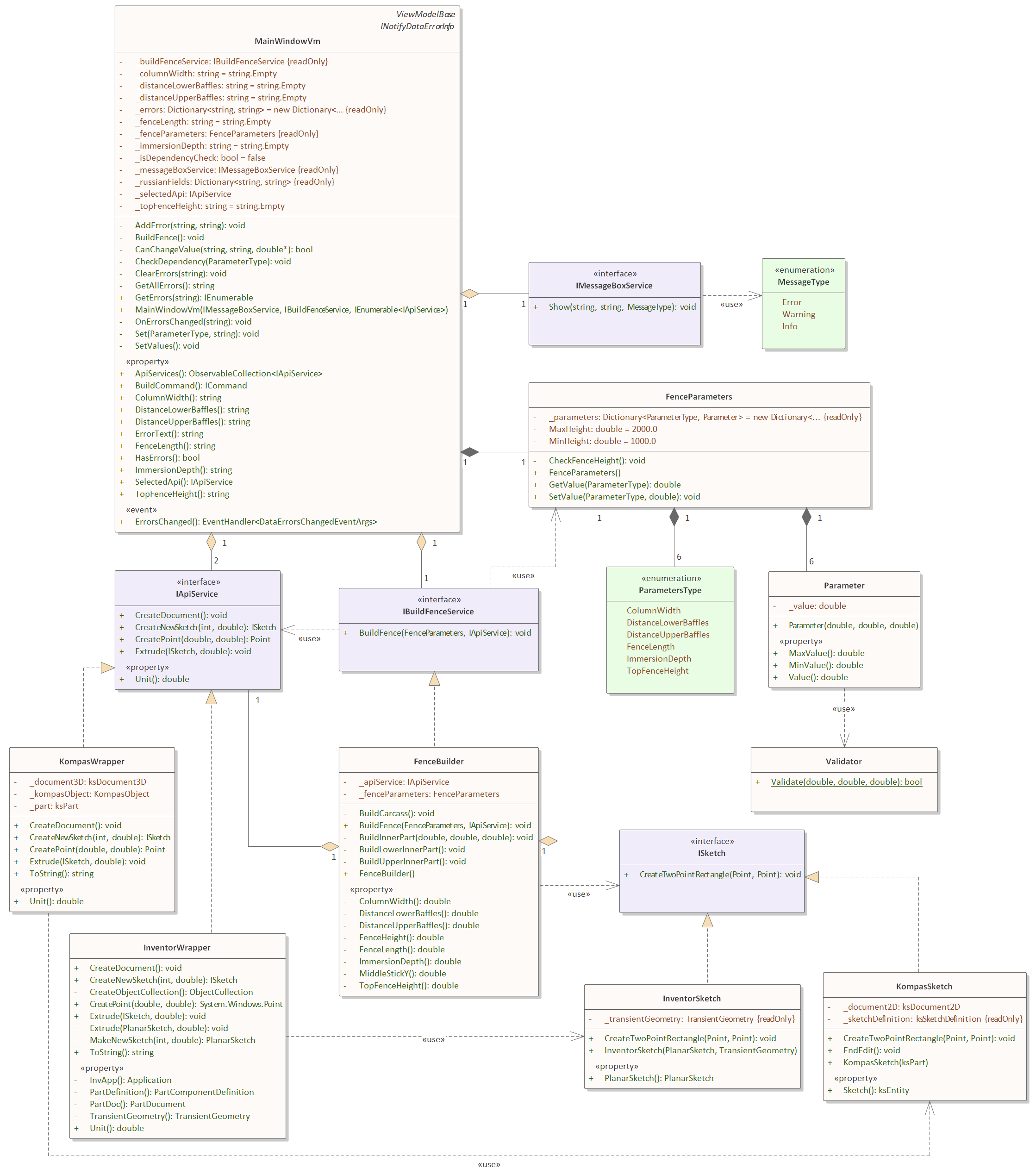


Рисунок 4.2 – Итоговая диаграмма классов

Был добавлен интерфейс (IApiService) для того, чтобы программа могла поддерживать несколько САПР для построения детали «Забор». Также были реализованы два класса от этого интерфейса — InventorWrapper и KompasWrapper.

Нужно было отделить от сущности эскизов разных САПР, поэтому был создан интерфейс ISketch для работы с эскизами. Также были реализованы два класса от этого интерфейса — InventorSketch и KompasScetch.

Для более простого изменения и чтения параметров был создан класс Parameter и перечисление ParameterType. Благодаря этому были убраны дубли в основной логике и в модульном тестировании.

# 5 Описание программы для пользователя

Макет пользовательского интерфейса представляет собой форму для ввода параметров забора. Построение модели осуществляется путем нажатия на кнопку «Построить». При попытке ввода недопустимых символов, они не будут вводиться в строку (если необходимо ввести цифры, то невозможно будет ввести другие символы).

Плагин состоит из диалогового окна, которое имеет 6 полей ввода параметров, 1 кнопку и 1 поле для выбора САПР, 1 изображение чертежа забора.

На рисунке 5.1 представлен пользовательский интерфейс.

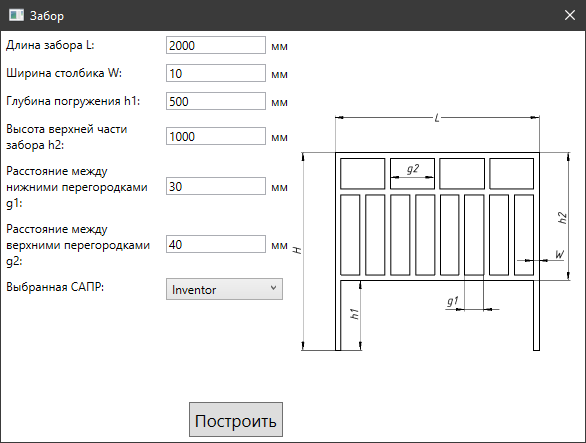


Рисунок 5.1 – Пользовательский интерфейс

Если ввести неверные параметры, после нажатия кнопки «Построить», высветится окно с просьбой ввести правильные параметры в поля ввода (рисунок 5.2).

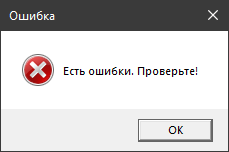


Рисунок 5.2 — Окно ошибки

После ввода необходимых параметров, построить деталь в САПР Inventor 2022 можно с помощью кнопки «Построить». Забор, построенный по заданным параметрам по умолчанию в САПР Inventor 2022, представлен на рисунке 5.3. Забор, построенный по заданным параметрам по умолчанию в САПР Kompas 3D, представлен на рисунке 5.4.

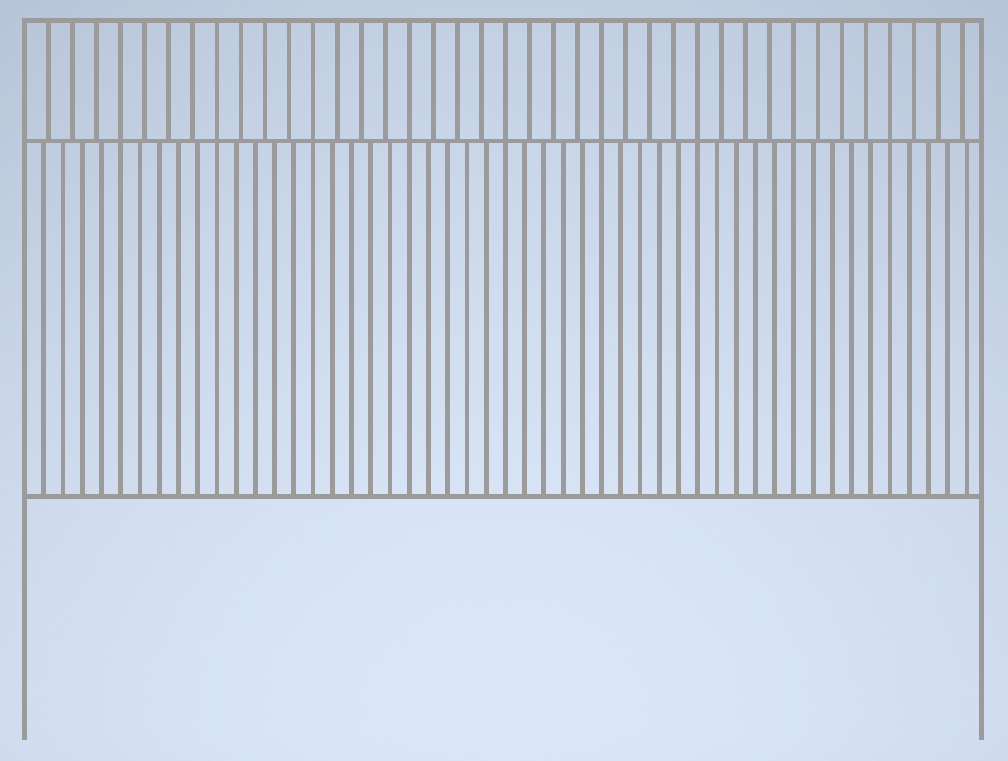


Рисунок 5.3 — Забор, построенный по заданным параметрам по умолчанию в САПР Inventor 2022



Рисунок 5.4 — Забор, построенный по заданным параметрам по умолчанию в САПР Kompas 3D

# 6 Тестирование программы

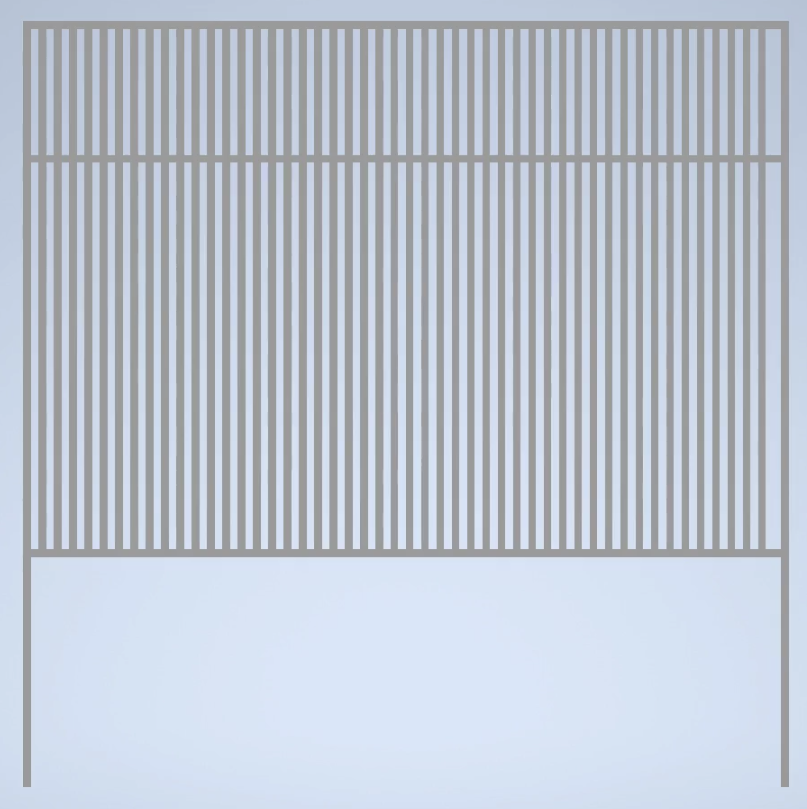
Тестирование позволяет убедиться в работоспособности программы, выявлять ошибки при изменении какого-либо функционала.

# 6.1 Функциональное тестирование

При функциональном тестировании проверялось корректность работы плагина «Забор», а именно, соответствие полученного результата в виде трехмерной модели, с входными параметрами. [13]

Проведено тестирование максимальных и минимальных параметров модели.

На рисунках 6.1 и 6.2 представлены проверки размеров модели с минимальным введенными параметрами в САПР Inventor 2022 и Kompas 3D (длина забора 1000 мм, ширина столбика 10 мм, глубина погружения 300 мм, высота верхней части забора 700 мм, расстояние между нижней перегородками 10 мм, расстояние между верхними перегородками 10 мм).

  
Рисунок 6.1 – Модель с минимальными введенными параметрами в Inventor 2022

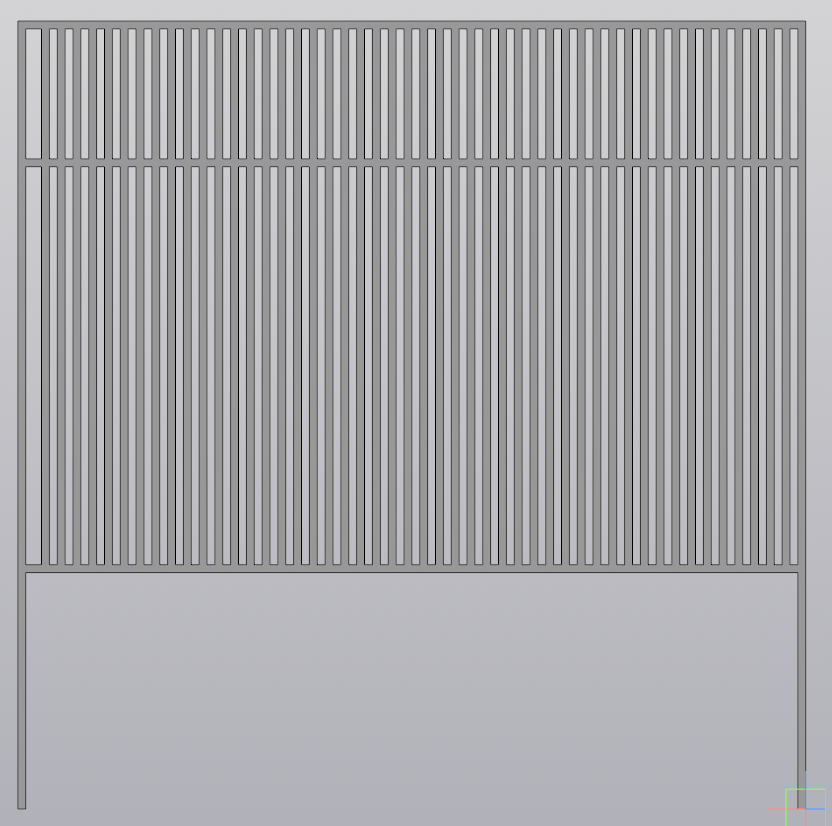


Рисунок 6.2 – Модель с минимальными введенными параметрами в Kompas 3D

Ниже на рисунках 6.3 и 6.4 представлены проверки размеров модели с максимальными введенными параметрами в САПР Inventor 2022 и Kompas 3D (длина забора 3000 мм, ширина столбика 70 мм, глубина погружения 500 мм, высота верхней части забора 1500 мм, расстояние между нижней перегородками 1500 мм, расстояние между верхними перегородками 1500 мм)

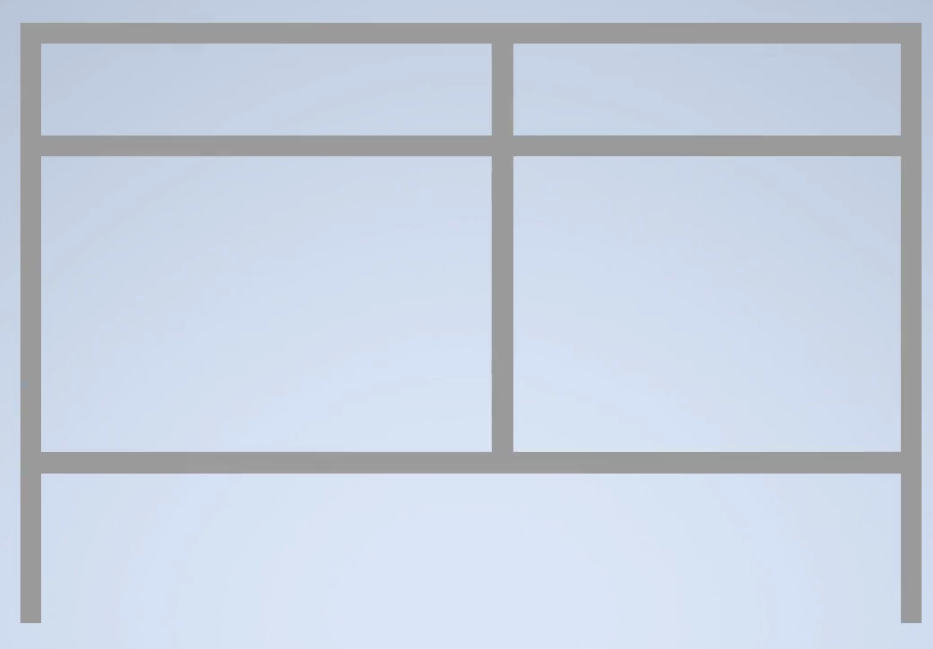


Рисунок 6.3 — Модель с максимально веденными параметрами в Inventor 2022

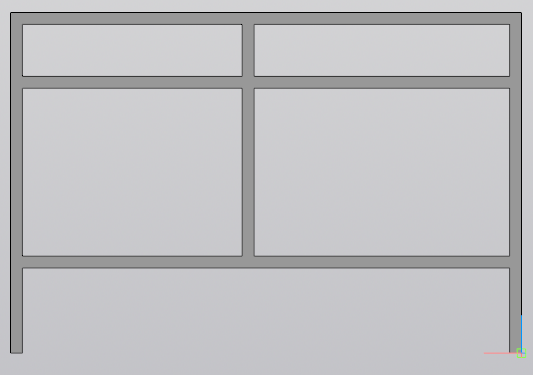
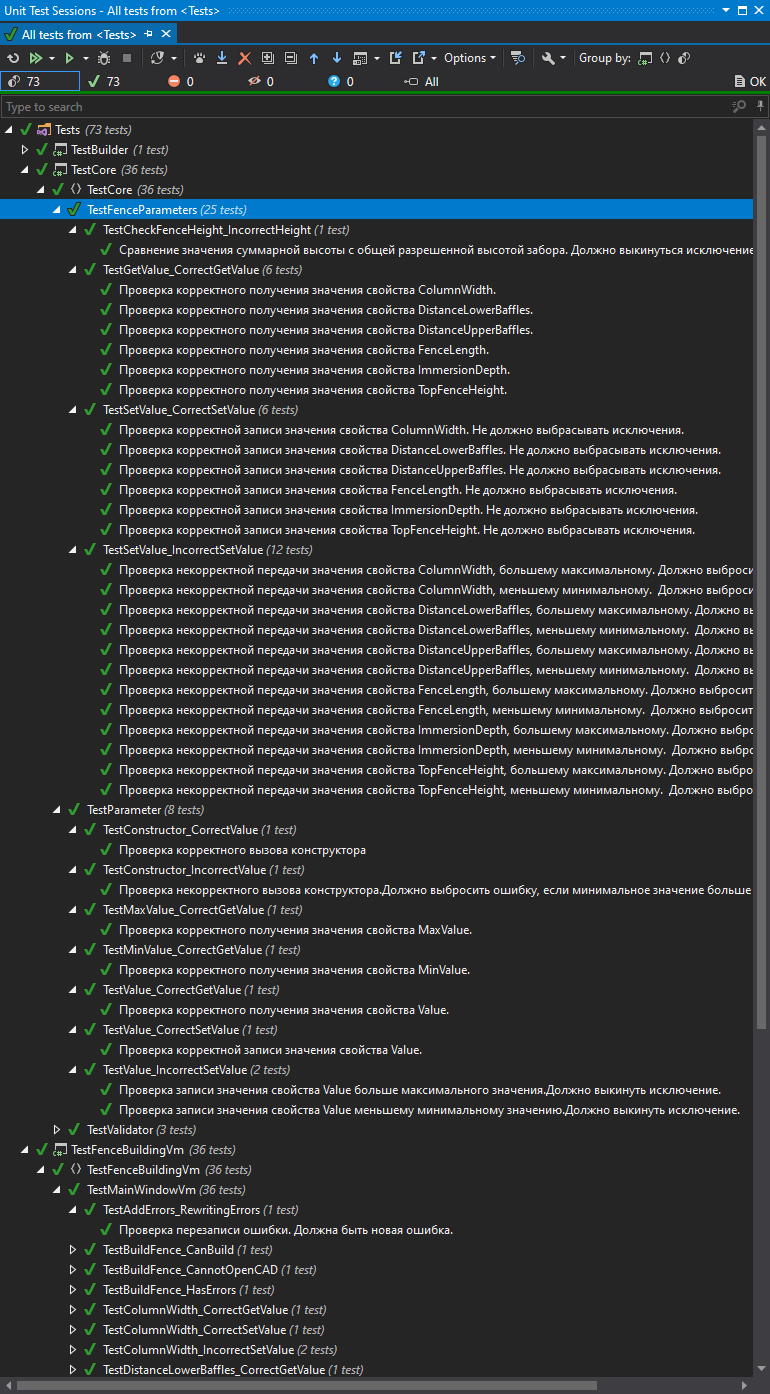


Рисунок 6.4 — Модель с максимально веденными параметрами в Kompas 3D

# 6.2 Модульное тестирование

В целях проверки корректности работы методов и свойств классов при помощи тестового фреймворка NUnit версии 3.13 проведено модульное тестирование [14], проверялись открытые поля и методы. На рисунке 6.5 представлено тестирование классов проектов: FenceBuildingVm, Core, Builder. Степень покрытия проектов — сто процентов. Было написано семьдесят три теста.

  
Рисунок 6.5 – Тестирование классов

# 6.3 Нагрузочное тестирование

В целях проверки производительности работы плагина, было проведено нагрузочное тестирование [15]. Тестирование производилось на ПК со следующей конфигурацией:

* ЦП Intel Core i5-8600K 3.6ГГц;
* 16 ГБ ОЗУ;
* графический процессор объемом памяти 6 ГБ.

На рисунке 6.6 для проведения нагрузочного тестирования был добавлен секундомер («Stopwatch»), который засекал время от начала построения, с каждым успешным построением фигуры производилась запись результатов в текстовый файлы «logInventor.txt» и «logKompas3D.txt».

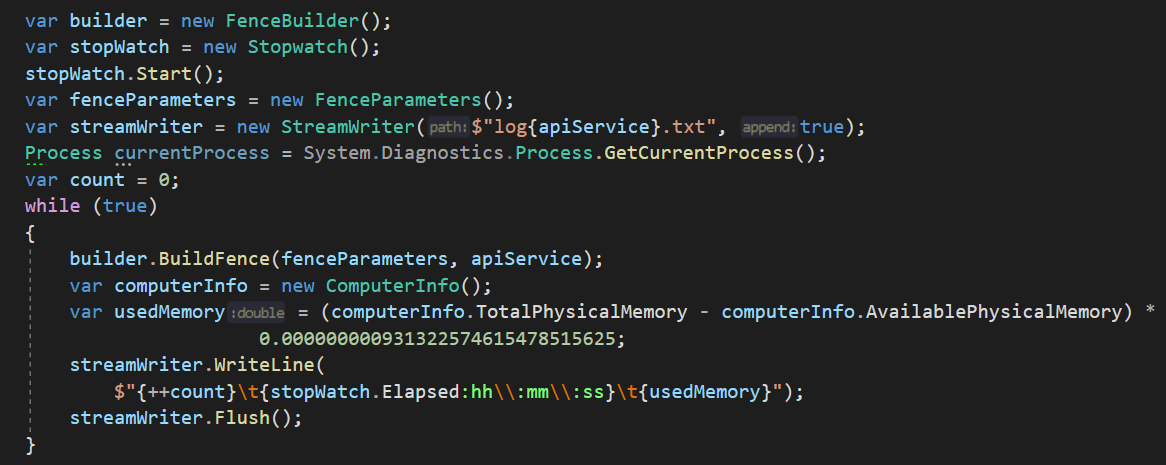


Рисунок 6.6 – Зацикливание перестроения фигуры

На графике, изображенном на рисунке 6.7 ось «Х» - количество построенных деталей, ось «Y» - количество потребляемой оперативной памяти. На графике, изображенном на рисунке 6.8 в текущей главе, ось «X» – время в секундах, ось «Y» – количество построенных деталей. На протяжении всех тестов (продолжительностью до сбоя Kompas 3D и Inventor) общая загруженность процессора была в пределах 22 процентов, потребление ОЗУ плагином прямолинейное от 14МБ до 20МБ для тестирования на САПР Inventor. Для САПР Kompas 3D общая загруженность процессора была в пределах 20 процентов, потребление ОЗУ плагином прямолинейное от 14мб до 20мб

На рисунке 6.7 представлено тестирование зацикленного перестроения фигуры со следующими параметрами:

* длина забора 2000 мм,
* ширина столбика 10 мм,
* глубина погружения 500 мм,
* высота верхней части забора 1000 мм,
* расстояние между нижней перегородками 30 мм,
* расстояние между верхними перегородками 40 мм

Рисунок 6.7 – График зависимости загруженности памяти от количества деталей

Рисунок 6.8 – График зависимости времени от количества построенных деталей с параметрами по умолчанию

Исходя из вышеуказанных графиков на рисунках 6.7 и 6.8, скорость построения в САПР Kompas 3D выше, чем на САПР Inventor. На рисунке 6.8 видно, что до запуска плагина, было занято около 8000 МБ оперативной памяти системой и сторонними процессами, которые к самому плагину отношения не имеют. САПР Kompas 3D потребляет больше оперативной памяти, чем Inventor.

# Заключение

В ходе выполнения лабораторных работ были изучены предметная область проектирования, предмет проектирования, аналоги предмета проектирования, API, функциональное и нагрузочное тестирование и на основании полученных данных были спроектированы UML диаграммы классов, разработан плагин для создания 3D моделей «Забор» в САПР Inventor 2022 и Kompas 3D, и проведено функциональное и нагрузочное тестирование плагина.

# Список использованных источников

1. Автоматизация вычислительных процедур в прикладных задачах инженерного проектирования [Электронный ресурс]. – URL: https://scienceforum.ru/2014/article/2014000201 (дата обращения: 10.12.2021).
2. Visual Studio [Электронный ресурс]. – URL: <https://visualstudio.microsoft.com/ru/> (дата обращения: 10.12.2021).
3. Autodesk Inventor— Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/Autodesk\_Inventor (дата обращения 22.10.2021).
4. КОМПАС-3D. Официальный сайт САПР КОМПАС [Электронный ресурс]. — Режим доступа: https://kompas.ru/ (дата обращения 22.10.2021).
5. Забор — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B1%D0%BE%D1%80 (дата обращения 22.10.2021).
6. Разработка приложений для Inventor - Autodesk. [Электронный ресурс]. – URL: https://www.autodesk.ru/autodesk-developer-network/software-platform-russian/develop-inventor (дата обращения: 10.12.2021).
7. КОМПАС-3D для разработчиков [Электронный ресурс]. – URL: https://kompas.ru/solutions/developers/ (дата обращения: 10.12.2021).
8. NUnit [Электронный ресурс]. – URL: <https://nunit.org/> (дата обращения: 10.12.2021).
9. Что такое Windows Presentation Foundation (WPF) [Электронный ресурс]. – URL: https://docs.microsoft.com/ru-ru/visualstudio/designers/getting-started-with-wpf?view=vs-2022 (дата обращения: 10.12.2021).
10. Sketchup — Википедия. [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/SketchUp (дата обращения 10.12.2021).
11. Плагин Instant Fence and Railing — Sketchup3DConstruction. [Электронный ресурс]. — Режим доступа: sketchup3dconstruction.com/skp/instant-fenceand-railing-plugin-for-sketchup.htm (дата обращения 10.12.2021).
12. UML. [Электронный ресурс]. – Режим доступа: http://www.uml.org/ (дата обращения 10.12.2021).
13. Функциональное тестирование [Электронный ресурс]. – URL: https://daglab.ru/funkcionalnoe-testirovanie-programmnogo-obespechenija/ (дата обращения: 10.12.2021).
14. Юнит-тестирование для чайников [Электронный ресурс]. – URL: <https://habr.com/ru/post/169381/> (дата обращения: 10.12.2021).
15. Нагрузочное тестирование: с чего начать и куда смотреть [Электронный ресурс]. – URL: <https://habr.com/ru/company/jugru/blog/329174/> (дата обращения: 10.12.2021).