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)

**Плагин для создания настольной лампы, для САПР КОМПАС-3D v18.1**

Проект системы

по дисциплине «Основы разработки САПР»

Выполнил

Студент гр. 587-3

\_\_\_\_\_\_\_\_\_\_\_ Е.А. Краснов

\_\_\_\_\_\_\_\_\_

Проверил:

Доцент кафедры КСУП, к.т.н.

\_\_\_\_\_\_\_\_\_\_\_ А.А. Калентьев

\_\_\_\_\_\_\_\_\_\_

Томск 2021

Оглавление

1 Назначение плагина 3

2 Группы пользователей и их функциональные

возможности плагина 5

3 Стек технологий разработки 5

4 Программные требования 5

5 Аппаратные требования 5

6 Пользовательский интерфейс 5

7 Требования к производительности 5

**1 Назначение плагин**

**«Компас»** — семейство систем автоматизированного проектирования, универсальная система автоматизированного проектирования, позволяющая в оперативном режиме выпускать чертежи изделий, схемы, спецификации, таблицы, инструкции, расчетно-пояснительные записки, технические условия, текстовые и прочие документы. Изначально система ориентирована на оформления документации в соответствии с ЕСКД, ЕСТД, СПДС и международными стандартами, но этим возможности системы не ограничиваются.

Система «Компас-3D» предназначена для создания трёхмерных ассоциативных моделей отдельных деталей (в том числе, деталей, формируемых из листового материала путём его гибки) и сборочных единиц, содержащих как оригинальные, так и стандартизованные конструктивные элементы. Параметрическая технология позволяет быстро получать модели типовых изделий на основе проектированного ранее прототипа. Многочисленные сервисные функции облегчают решение вспомогательных задач проектирования и обслуживания производства.

Система «Компас-3D» включает следующие компоненты: система трёхмерного твердотельного моделирования, универсальная система автоматизированного проектирования «Компас-График» и модуль формирования спецификаций. Ключевой особенностью «Компас-3D» является использование собственного математического ядра и параметрических технологий. [1]

**2 Описание API**

API (англ. Application Programming Interface) – описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой.

В КОМПАС-3D существуют API двух версий: API 5 и API 7. Обе версии реализуют различные функции системы и дополняют друг друга. Обе версии программных интерфейсов в равной мере поддерживаются и развиваются с учетом самих изменений в системе. В основном, для создания полноценных подключаемых модулей достаточно методов и свойств интерфейсов API 5.

Главным интерфейсом API системы КОМПАС-3D является KompasObject. Получить указатель на этот интерфейс можно при работе под управлением внешнего приложения (контроллера) – после вызова стандартной системной функции. Методы этого интерфейса реализуют наиболее общие функции работы с документами системы, системными настройками, файлами, а также дают возможность получить указатели на другие интерфейсы [2].

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

Таблица 1.1. – Методы интерфейса KompasObject.

|  |  |  |
| --- | --- | --- |
| Метод | Возвращаемое значение | Описание |
| Document3D() | Указатель на интерфейс документа трёхмерной модели ksDocument3D | Даёт возможность получить указатель на интерфейс трёхмерного документа (детали или сборки) |
| Visible |  | Свойство видимости приложения |
| Quit |  | Метод для завершения программы Kompas-3D |
| ActivateControllerAPI |  | Метод для активации контроллера API |
| ksDocument2D |  | Интерфейс событий графического документа, события интерфейса позволяют контролировать состояние документа. |

Таблица 1.2 – Методы интерфейса IPart.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| GetDefaultEntity  (short objType) | |  |  | | --- | --- | | objType | - тип объекта. | | |  | | --- | | Указатель на интерфейс [ksEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm) или [IEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm). |   Типы объектов (objType):   |  |  |  | | --- | --- | --- | | o3d\_planeXOY | 1 | - плоскость XOY | | o3d\_planeXOZ | 2 | - плоскость XOZ | | o3d\_planeYOZ | 3 | - плоскость YOZ | | o3d\_pointCS | 4 | - точка начала системы координат | | o3d\_axisOX | 71 | - ось OX | | o3d\_axisOY | 72 | - ось OY | | o3d\_axisOZ | 73 | - ось OZ | | Получить указатель на интерфейс объекта, создаваемого системой по умолчанию |
| GetPart(int type) | |  |  | | --- | --- | | type | - тип компонента. | | указатель на интерфейс компонента [ksPart](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksPart.htm) или [IPart](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksPart.htm). | Получить указатель на интерфейс компонента в соответствии с заданным типом |
| NewEntity(short objType) | |  |  | | --- | --- | | ob | jType- [тип объекта](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/Obj3dType_NewEntil_Part.htm). | | указатель на интерфейс [ksEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm) или [IEntity](mk:@MSITStore:D:\INSTAL\KOMPAS-3D%20V17.1\KOMPAS\SDK\SDK.chm::/ksEntity.htm). | Создать новый интерфейс объекта и получить указатель на него |

Таблица 1.3. Методы интерфейса ksDocument3D.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| Create (bool invisible, bool \_typeDoc) | invisible – признак режима редактирования документа  (TRUE – невидимый режим,  FALSE – видимый режим),  typeDoc – тип документа  (TRUE – деталь,  FALSE – сборка). | TRUE – в случае успешного завершения. | Дает возможность создать пустой документ (деталь или сборку) |

Продолжение таблицы 1.3

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| GetPart(int type) | type – тип компонента из перечисления Типы компонентов. |  | Получить указатель на интерфейс компонента в соответствии с заданным типом |

Таблица 1.4 – Методы интерфейса ksDocument2D

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| ksCircle (double xc, double yc, double rad, long style) | xc, yc - координаты центра окружности, rad - радиус окружности, style - стиль линии. | указатель на окружность - в случае удачного завершения, 0 - в случае неудачи. | Метод для создания окружности. |
| long ksLineSeg (double x1, double y1, double x2, double y2, long style) | x1, y1 - координаты первой точки отрезка, x2, y2 - координаты второй точки отрезка, style - стиль линии. | указатель на отрезок - в случае удачного завершения, 0 - в случае неудачи. | Метод для создания отрезка. |
| ksArcByPoint (double xc, double yc, double rad, double f1, double f2, short direction, long style); | xc, yc - координаты центра окружности, rad - радиус окружности, x1, y1 - координаты начальной точки дуги, x2, y2 - координаты конечной точки дуги, direction - направление отрисовки дуги: 1 - против часовой стрелки, -1 - по часовой стрелке, style - стиль линии. | указатель на отрезок - в случае удачного завершения, 0 - в случае неудачи. | Метод создает дугу по центру и конечным точкам |

Таблица 1.5 - Методы интерфейса [ksEntity](mk:@MSITStore:C:\Program%20Files\ASCON\KOMPAS-3D%20v18%20Study\SDK\SDK.chm::/ksEntity_props.htm).

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| ksBossExtrusionDefinition(BOOL forward, short type, double depth, double draftValue, BOOL draftOutward); | Forward- - направление выдавливания: TRUE - прямое направление, FALSE - обратное направление, type - тип выдавливания, depth - глубина выдавливания, draftValue - угол уклона, draftOutward - направление уклона: FALSE - уклон наружу, TRUE - уклон внутрь. | TRUE – в случае успешного завершения. | Метод выдавливает эскиз в одном направлении |

Продолжение таблицы 1.5

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Входные параметры | Возвращаемое значение | Описание |
| ksCutExtrusionDefinition (BOOL forward, short type, double depth, double draftValue, BOOL draftOutward); | Forward- - направление выдавливания: TRUE - прямое направление, FALSE - обратное направление, type - тип выдавливания, depth - глубина выдавливания, draftValue - угол уклона, draftOutward - направление уклона: FALSE - уклон наружу, TRUE - уклон внутрь. | TRUE – в случае успешного завершения. | Метод вырезания выдавливанием эскиз в одном направлении |

**3 Обзор аналогов**

Dialux - одна из самых функциональных компьютерных программ для выполнения светотехнических расчетов и инженерного проектирования внутреннего и внешнего освещения.

По изначально заданным условиям: количество светильников, их тип, расположение, - программа Dialux способна проводить разнообразные сложные светотехнические расчеты, при которых обязательно будут учтены все факторы, связанные с мебелью, различными предметами интерьера, геометрией помещения, цветом и текстурой всех поверхностей. Программа позволяет проводить расчеты для любых видов освещенности, КЕО, яркости, показателей блесткости, теней и дневного света. Утилита учитывает погодные условия, географическое расположение объекта, тени от окружающих объектов и зданий.

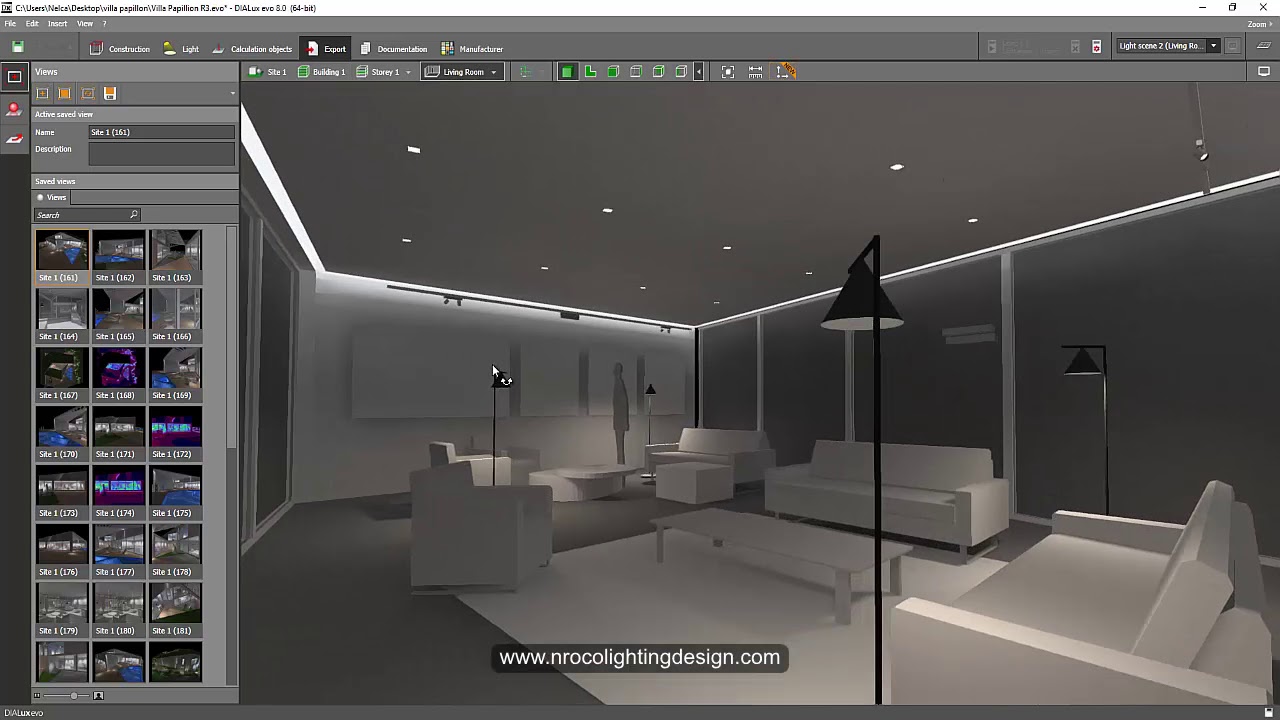
Для каждого участка можно смоделировать разнообразные световые эффекты с зеркальным отражением, прозрачностью, улучшенными текстурами по заданным параметрам. Модернизированное ядро программы делает просчеты сцен быстрыми, а цвета более естественными и насыщенными.

Программой полностью поддерживаются все современные национальные и международные стандарты и европейские единицы измерения. Можно экспортировать-импортировать объекты и данные в и из любых CAD-программ в форматах .dwg и .dxf. Наличие всплывающих подсказок и интуитивно понятное управление сильно облегчает работу с программой.

Программа DIALux значительно упрощает процесс расчета системы общего искусственного освещения помещений с трехмерной визуализацией проектных решений. Программа интерактивна: она позволяет пользователю перемещаться в освещенном, в соответствии с расчетом, интерьере.[3]

Интерфейс программы DIALux представлен на рисунке 3.1.

Рисунок 3.1 – Программа DIALux



**4 Описание предмета проектирования**

Настольные лампы служат для освещения определенной поверхности рабочего стола. Чаще всего их устанавливают для школьников, а также в рабочих кабинетах для создания качественного рабочего освещения в вечернее время, так как естественного света уже не хватает. Общего освещения комнаты также недостаточно. Кроме этого, настольные приборы освещения необходимы для вязания, вышивания и шитья.

Плагин, предназначен для создания светильников заранее определенного дизайна (рисунок 4.1 – 4.3). Плагин должен:

1. Обеспечить изменение размера основания , .
2. Обеспечивать изменение размера у стойки , .
3. Обеспечивать изменение размера у площадки на которой закреплен патрон , .
4. Обеспечивать создание выреза под кнопку включения , , .
5. Обеспечивать создание выреза под электрический провод , , .
6. Обеспечивать создание отверстия под электрический провод в центре основания , .
7. Обеспечивать создание 2 отверстий под саморезы с расстоянием , и диаметром .

Рисунок 4.1 ­ Изображение настольной лампы

Рисунок 4.2 ­ Изображение настольной лампы

Рисунок 4.3 ­ Изображение настольной лампы

**5 Диаграмма классов**

Диаграмма классов описывает типы объектов системы и различного рода статические отношения, которые существуют между ними. На диаграммах классов отображаются также свойства классов, операции классов и ограничения, которые накладываются на связи между объектами. Диаграмма классов приведена на рисунке 5.1.

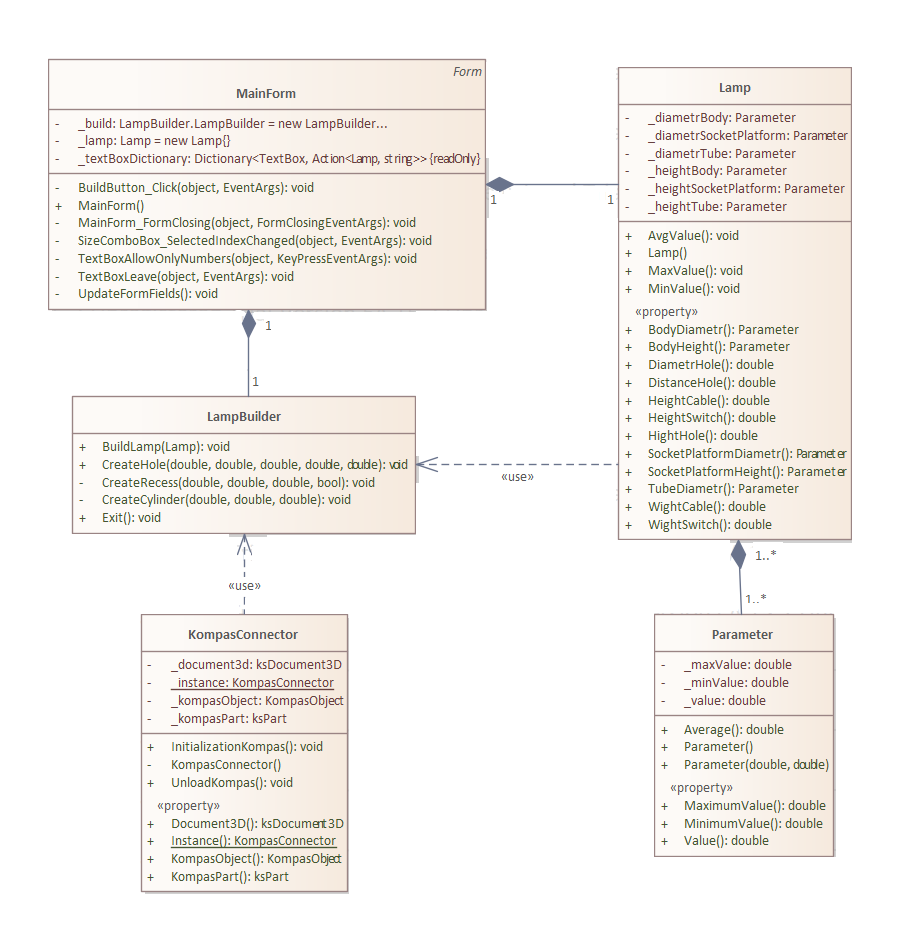
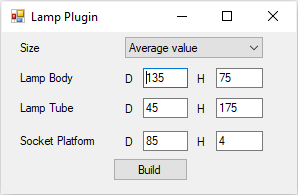


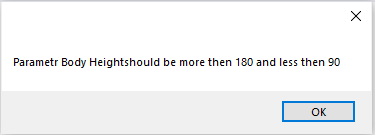
Рисунок 5.1 – Диаграмма классов

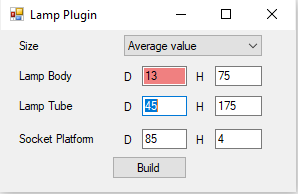
Для реализации был выбран следующий набор классов:

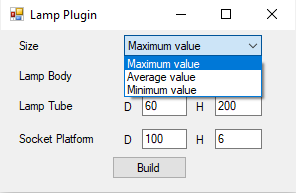
1. MainForm – класс диалогового окна, который обеспечивает взаимодействие между пользователем и программой;
2. Lamp − класс, хранящий в себе все параметры проектируемой 3D-модели;
3. Parameter – класс хранящий значение параметра, максимальное и минимальное ограничение.
4. KompasConnector – класс для работы с API КОМПАС 3D.
5. LampBuilder – класс, осуществляющий вызов методов API, необходимых для постройки 3D-модели.

**6 Макет пользовательского интерфейса**









Тут должно быть описание UI

**Список литературы**

4 http://electricalschool.info/main/lighting/1703-programma-dialux-dlja-raschjota-i.html#:~:text=Dialux%20%2D%20%D0%BE%D0%B4%D0%BD%D0%B0%20%D0%B8%D0%B7%20%D1%81%D0%B0%D0%BC%D1%8B%D1%85%20%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D1%85,%D1%8D%D1%82%D0%BE%D0%BC%20%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D1%8F%D0%B5%D1%82%D1%81%D1%8F%20%D0%B8%20%D0%BE%D0%B1%D0%BD%D0%BE%D0%B2%D0%BB%D1%8F%D0%B5%D1%82%D1%81%D1%8F%20%D0%B1%D0%B5%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D0%BD%D0%BE.

Тут должны быть ссылки