# Проект системы

# Описание САПР

## Описание программы

AutoCAD – это программное обеспечение автоматизированного проектирования (САПР), с помощью которого архитекторы, инженеры и строители создают точные 2D– и 3D–чертежи. [1]

AutoCAD поддерживает запись (экспорт) файлов, формата DGN, SAT, STL, IGES, FBX, DWG и DXF. А также чтение (импорт) файлов, формата 3DS, DGN, JT, SAT, PDF, STEP и некоторых других. Начиная с версии 2012, AutoCAD позволяет преобразовывать файлы, полученные из трёхмерных САПР (таких как Inventor, SolidWorks, CATIA, NX и т. п.) в формат DWG.

## Описание API

**API** — это аббревиатура от *Application Programming Interface*: способ взаимодействия программиста с программным продуктом. Например, AutoCAD API — это способ, которым программисты могут работать с AutoCAD, и он определяет, какие функции программист может использовать в AutoCAD. Например, запрограммировать AutoCAD для добавления линии к чертежу, потому что AutoCAD предоставляет функциональные возможности для этого через свой API.На рисунке 1 изображена схема взаимодействия.

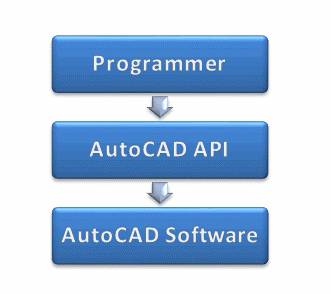


Рисунок 1 Схема взаимодействия

Иными словами: коммерческие компании-разработчики программного обеспечения, такие как Autodesk, часто распространяют набор *библиотек,* которые используются в своей собственной программе для взаимодействия с конкретным программным продуктом, например AutoCAD, и расширения его функциональности.Этот набор библиотек известен как API программного продукта.

**ObjectARX**

Среда программирования ObjectARX® используется для адаптации и расширения функциональных возможностей AutoCAD и продуктов на его основе. Она обеспечивает непосредственный доступ к структурам базы данных AutoCAD, графической системе и определениям встроенных команд. С помощью объектно-ориентированных интерфейсов программирования на языке C++ разработчики могут создавать приложения для AutoCAD.

Библиотеки среды программирования ObjectARX предоставляют универсальные наборы инструментов, с помощью которых разработчики AutoCAD API могут воспользоваться преимуществами открытой архитектуры программного обеспечения AutoCAD. Они также обеспечивают прямой доступ к структурам базы данных AutoCAD, графической системе и собственному определению команд. Технология ObjectARX помогает разрабатывать быстрые, эффективные и компактные приложения САПР. Он позволяет опытным пользователям настраивать программное обеспечение AutoCAD и освобождает проектировщиков САПР от повторяющихся задач. Файлы меньшего размера, более быстрые операции рисования и плавное взаимодействие делают приложение, созданное в среде программирования ObjectARX, лучшим выбором для программного решения для проектирования.[1]

**.NET**

В состав ObjectARX SDK входит также управляемый API, который часто называют AutoCAD .NET API. Для адаптации и расширения функциональных возможностей AutoCAD и продуктов на его основе может применяться любой язык программирования, поддерживающий .NET. [2]

AutoCAD .NET API собран из различных DLL-файлов, которые содержат широкий ряд классов, структур, методов и событий, обеспечивающих доступ к объектам файла чертежа в приложении AutoCAD.

Каждый DLL-файл определяет различные пространства имен, которые используются для организации размещения компонентов библиотек.

Четыре основные DLL-файла AutoCAD .NET API:

1. AcDbMgd.dll. Используется для работы с объектами файла чертежа;
2. AcMgd.dll. Используется для работы с самим приложением AutoCAD;
3. AcCui.dll. Используется для работы с файлами пользовательских настроек;
4. AcCoreMgd.dll. Содержит часть функционала из файла AcDbMgd.dll и часть из файла AcMgd.dll.

Основные методы AutoCAD .NET API, используемые при создании плагина представлены в таблицах 1.1 – 1.4.

Таблица 1.1 – Основные методы интерфейса DocumentManager

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| MdiActiveDocument() | Document | Метод для создания и получения документа чертежа |
| MdiActiveDocument.Editor() | Editor | Метод для получения редактора текущего чертежа |

Таблица 1.2 – Основные методы интерфейса TransactionManager.

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| StartManager() | Transaction | Метод, реализующий работу с примитивами |
| Transaction.Commit() | void | Метод для завершения работы с примитивами |

Таблица 1.3 – Основные методы класса BlockTableRecord

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| AppendEntity(Object object) | void | Метод, добавляющий в текущее пространство примитив типа Object |

Таблица 1.4 – Основные методы класса Solid3d

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| CreateWedge(double length, double width, double height) | void | Метод для создания объемного объекта по заданной длине, ширине и высоте |
| Extrude(int region, double height, double angle) | void | Метод, выполняющий выдавливание указанной области region, на высоту height при заданном наклоне angle |

## Обзор аналогов

## 1.3.1 Плагин SHVAC–RD–3D для AutoCAD

Данный плагин, от компании Soft Draft, предназначен для моделирования угольника, выступа и трубу. [3]

На рисунке 1.2 представлен пользовательский интерфейс плагина, для построения выступа.



Рисунок 1.2 – Пользовательский интерфейс плагина SHVAC–RD–3D

## 1.3.2 Плагин Bolts 3D imperial для AutoCAD

Плагин создает болты, гайки и шайбы в виде трехмерных тел для удобного размещения в трехмерных моделях конструкционной стали.[3]

Встроенный поворот позволяет размещать болты в направлении вниз, сбоку или вверх.

Размерные данные для болтов хранятся во внешних файлах, в которые возможно добавить дополнительную информацию о размерах болтов или гаек, которые являются уникальными для конкретного применения.

На рисунке 1.3 представлен пользовательский интерфейс плагина.



Рисунок 1.3 – Пользовательский интерфейс плагина Bolts 3D Imperial

## Описание предмета проектирования

Предметом проектирования является модель графина

Параметры графина:

* Габариты графина: длина L (от 50 до 200 мм), ширина W (от 50 до 200 мм), высота H (от 50 до 500 мм);
* Диаметр основания D: от 50 до 100 мм;
* Диаметр горла графина d: от 25 мм до диаметра основания;
* Форма графина: квадратный, шарообразный, стандартный;
* Наличие / отсутствие пробки;
* Форма пробки и ее высота h (от 10 мм до 70 мм);
* Наличие / отсутствие ручки;
* Длина ручки l: от 25 мм до 2/3 высоты графина.

На рисунке 4 показаны геометрические параметры графина

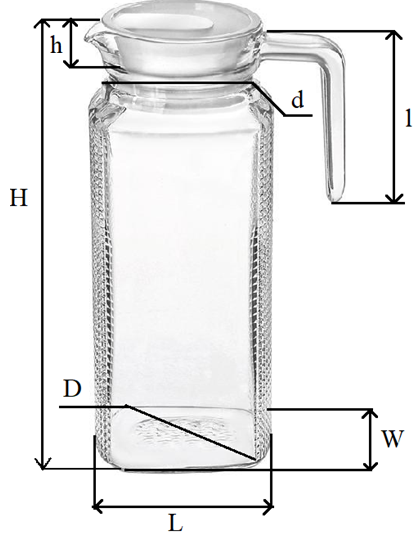
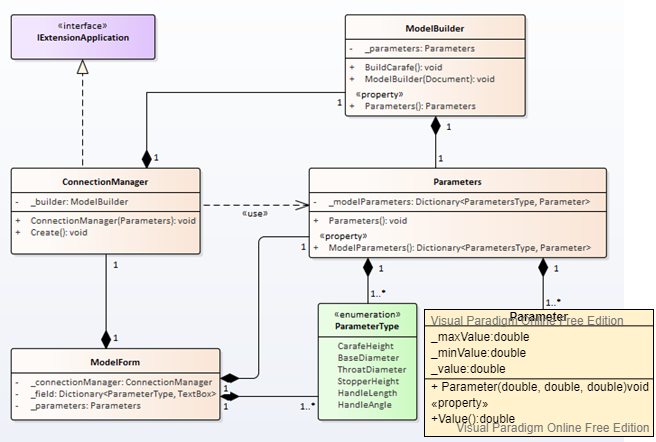


Рисунок 4 – Геометрические параметры графина

## 3 Диаграмма классов

UML диаграммы классов представляют собой графическую интерпретацию классов системы, их атрибутов, методов и взаимосвязей между ними.

На рисунке 5 показана UML диаграмма классов

Рисунок 5 – Диаграмма классов плагина

Класс «ModelForm» является формой пользовательского интерфейса. Реализует методы, используемые для взаимодействия с пользователем.

ModelBuilder – класс, который хранит параметры модели, введенные пользователем.

ConnectionManager – класс, реализующий методы для связи программы с САПР и хранящий объект класса построителя 3D модели (BuilderModel),

Класс Parameter хранит информацию об одном параметре проектируемой модели.

Класс Parameters хранит словарь с параметрами модели.

ParameterType – перечисление, содержащее название параметров графина.

# Список использованных источников

1. Официальный сайт AutoDesk. Продукт AutoCAD [Электронный ресурс] – Режимдоступа:<https://www.autodesk.com/developer-network/platform-technologies/autocad/objectarx> (дата посещения: 16.03.2021);
2. Официальный сайт AutoDesk. [Электронный ресурс] – Режим доступа: <https://www.autodesk.com/autodesk-university/ru/forge-content/au_class-urn%3Aadsk.content%3Acontent%3Aad8a2bb7-4eb5-4e36-a5a2-42448b9d54ed> (дата посещения: 16.03.2021);
3. Официальный сайт SoftDraft. [Электронный ресурс] – Режим доступа: <https://www.softdraft.com/> (дата посещения: 16.03.2021).