

**RODRIGO DINIS**

**DESENVOLVIMENTO DE UM SOFTWARE PÓS-  
PROCESSADOR PARA MÁQUINAS CNC DE LAPIDAÇÃO DE  
GEMAS**

**FLORIANÓPOLIS, 2016**

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA DE SANTA CATARINA  
CAMPUS DE FLORIANÓPOLIS  
DEPARTAMENTO ACADÊMICO DE METAL-MECÂNICA  
MESTRADO PROFISSIONAL EM MECATRÔNICA**

**RODRIGO DINIS**

**DESENVOLVIMENTO DE UM SOFTWARE PÓS-  
PROCESSADOR PARA MÁQUINAS CNC DE LAPIDAÇÃO DE  
GEMAS**

**Dissertação submetida ao Programa de  
Pós-Graduação em Mecatrônica do  
Instituto Federal de Educação, Ciência  
e Tecnologia de Santa Catarina como  
parte dos requisitos para a obtenção do  
título de Mestre em Mecatrônica.**

**Orientador: Milton Pereira, Dr. Eng.  
Co-orientador: Aurélio da Costa Sabino  
Netto, Dr. Eng.**

**FLORIANÓPOLIS, 2016**

005.1  
D585d

Dinis, Rodrigo

Desenvolvimento de um software pós-processador para máquinas CNC de lapidação de gemas [DIS] / Rodrigo Dinis; orientação de Milton Pereira, co-orientação de Aurélio da Costa Sabino Netto – Florianópolis, 2016.

1 v.: il.

Dissertação de Mestrado (Mecatrônica) – Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina.

Inclui referências.

1. Software - Desenvolvimento. 2. Lapidação. 3. Automação. 4. Máquinas CNC. 5. Gemas – Mineralogia. I. Pereira, Milton. II. Sabino Netto, Aurélio da Costa. III. Título.

Sistema de Bibliotecas Integradas do IFSC  
Biblioteca Dr. Hercílio Luz – Campus Florianópolis

# DESENVOLVIMENTO DE UM SOFTWARE PÓS- PROCESSADOR PARA MÁQUINAS CNC DE LAPIDAÇÃO DE GEMAS

**RODRIGO DINIS**

Esta dissertação foi julgada adequada para obtenção do título de Mestre em Mecatrônica e aprovada na sua forma final pela banca examinadora do Programa de Pós-Graduação em Mecatrônica do Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina.

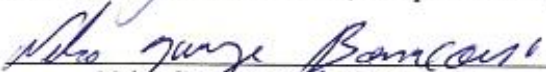
Florianópolis, 17 de fevereiro de 2016.

Banca Examinadora:



Milton Pereira, Dr. Eng.

Instituto Federal de Santa Catarina, Campus Florianópolis



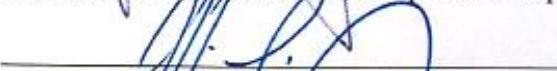
Nelson Gaudin Bonacorso, Dr. Eng.

Instituto Federal de Santa Catarina, Campus Florianópolis



Márcio Edgar Stivanello, Dr. Eng.

Instituto Federal de Santa Catarina, Campus Florianópolis



Jefferson de Oliveira Gomes, Dr. Eng.

Instituto Tecnológico de Aeronáutica, São José dos Campos

## DEDICATÓRIA

Dedico este trabalho à minha querida esposa, que teve muita paciência e colaboração comigo durante o desenvolvimento deste trabalho. Não há também como esquecer das minhas filhas Amanda e Júlia, que comigo estiveram ao lado ou longe, mas sempre em meu coração, durante o tempo de desenvolvimento deste trabalho.

Também aos meus pais, José e Rosilda, *in memoriam*, por tudo aquilo que recebi sem nunca pedirem nada em troca.

A minha muito amada vovó Aúrea, *in memoriam*, cuja vida foi um exemplo de desvelo à família e ao próximo.

## **PENSAMENTO**

Ser aluno é estar vivo, a vida tem muito a ensinar e temos que estar preparados para aprender. Sócrates, filósofo grego (469 a.C.-399 a.C.) em sua imensa sabedoria sabia que nada sabia, portanto estava sempre apto a novos conhecimentos. Galileo Galilei (15 Fevereiro 1564 – 8 Janeiro 1642), disse que nunca encontrara um homem tão ignorante que nada pudesse aprender com ele.

O que podermos apreender destes pensadores é que com simplicidade, humildade e boa vontade para com a ciência e a humanidade, estamos sempre aprendendo.

## **AGRADECIMENTOS**

Ao professor Felício Gesser, pelas incontáveis ajudas que me deu durante a pesquisa, fiz uso de seus ensinamentos e aproveitei muito as referências bibliográficas que ele indicou. Muito obrigado.

A minha esposa, pelo apoio, pelas revisões, e pelo carinho. Carinho este que também recebo das minhas amadas filhas, a Amanda e a Julia. Com vocês a vida é realmente muito melhor. Obrigado.

Ao meu orientador Dr. Eng. Milton Pereira pelo apoio, colaboração e principalmente pela simplicidade e motivação. Obrigado Professor Milton, sem tu este trabalho não seria possível.

Aos professores do Centro Universitário de Brasília, onde graduei-me, em especial ao Prof. Dr. Paulo Foina, o qual me deu suporte para iniciar esta jornada que foi o meu mestrado.

## RESUMO

No processo de lapidação manual há uma grande quantidade de diagramas de projeto de lapidação de gemas, cada um representando um modelo diferente de lapidação. A lapidação tem sido feita com o uso destes diagramas em máquinas não automatizadas, cujo tempo de produção é baixo e são muito passíveis de erros humanos. Existem softwares CAD bem difundidos na área de lapidação de gemas que possibilitam a automação da criação de projetos dando bastante flexibilidade de criação que trazem aos lapidários a geração de um modelo a ser seguido durante o processo de produção das gemas. Para melhorar a produtividade da lapidação a tendência é partir de modelos CAD existentes e integrar rapidamente com máquinas automáticas de lapidação. Tendo como fundamento cadeias de manufatura baseados em CNC, foi desenvolvido um software pós-processador, capaz de integrar o projeto feito em CAD e as máquinas automáticas CNC. Esta dissertação apresenta a concepção, o projeto e os resultados obtidos na construção do software, o qual trabalhará com diferentes formatos de lapidação de gemas, comandos e máquinas CNC. Ele gerará programas CNC de lapidação de forma flexível. Tem-se observado que a produtividade na lapidação manual é algo em torno de 15 peças por semana para cada lapidário. Verifica-se também que há problemas de padronização nas cadeias de manufatura. Primeiro nos softwares CAD, que não possuem formato de arquivo compatível, e em segundo, a linguagem CNC, que apesar de haver um padrão ISO, cada fabricante tem suas particularidades. O uso do software pós-processador permite a geração padronizada do código CNC, partindo de extensões de software para atender diferentes sistemas CAD e comandos CNC. Aliando-se a máquinas de lapidação automáticas, ele possibilita um aumento da produtividade, associado à melhoria e estabilidade da qualidade da lapidação.

**Palavras-chave:** Gemas, automação da lapidação, recursos minerais.

## ABSTRACT

In the manual gemstone cutting process there are several gemstone cutting designs based on diagrams, each one representing a different cutting model. The gemstone cutting have been made using these diagrams on non-automatized machines, whose production time is low and are quite susceptible to human errors. There are widespread CAD softwares in the lapidary field, which allows the design creation automation and gives great flexibility to bring to the lapidary a model to be followed during the gemstone production process. To improve the the gemstone cutting productivity the tendency comes from the use of CAD designs and also fastly integrate it to automatic cutting machines. Having as fundament the manufacturing chains based on CNC, was developed a post-processor software capable to integrate the CAD design and automatic CNC machines. This dissertation presents the concept, the design, and the results obtained in the software construction, which will work with different cutting designs and CNC machines to generate CNC programs in a flexible way. It has been observed that the production in the manual gemstone cutting results is something about 15 pieces a week per lapidary. Standartization problems has been also observed in the manufacturing chain. First in the CAD applications that do not have a file standard compatible and second, the CNC language, that despite the existent ISO standard, each manufacturer has their own particularities. The use of the post-processor software will allow a quick and standardized CNC code, from the extensions of software to attend different CAD systems and CNC controllers. Alided to automatic gemstone cutting machine it will allow: an increase in the productivity, together with improvement and stability of the gemstone cutting quality.

**Key-words:** Gemstones, Gem cutting automation, mineral resources.



## LISTA DE ILUSTRAÇÕES

Figura 1: Mapa gemológico brasileiro. (IBGM, 2005) .....	15
Figura 2 - Lapidação Cabochão (SCHUMANN, 2006) .....	20
Figura 3 - Lapidação Facetada .....	21
Figura 4 - Polidor de minerais (NADUR apud MONTAGNA, 2009) ...	22
Figura 5 - Lapidação Indiana. (NADUR apud BRUTON. 1978).....	23
Figura 6 – Lapidário usando máquina com graduador .....	23
Figura 7 - Graduador (Foto tirada pelo autor). .....	24
Figura 8 - Máquina de lapidar " <i>Faceting-head</i> " .....	25
Figura 9 - Máquina CNC para lapidação de diamante. ....	26
Figura 10 - Máquina automática com cabeçote e discos. ....	27
Figura 11 - Depois de escaneado, o software calcula o corte otimizado	28
Figura 12: Lapidação Brilhante Redondo (GEURTS et al., 2011).....	29
Figura 13: Pedra bruta após desbaste (NEVADA, 2015).....	33
Figura 14: Gema Após a Pré-Formação (NEVADA, 2015).....	33
Figura 15: Gema Após o Encanetamento e Transferência .....	34
Figura 16: Atividade de Facetamento (NEVADA, 2015). .....	35
Figura 17: Resumo da Atividade de Facetamento.....	36
Figura 18: Diagrama de Lapidação design Brilhante Redondo (STRICKLAND,2002).....	38
Figura 19: Cadeia de produção CAx mais comum.....	40
Figura 20: Identificação dos Eixos CNC Pela Regra da Mão Direita (MUNDO CNC,2016).....	41
Figura 21: Processo Básico de RP.....	43
Figura 22: Modelo de Processo Fabricação Convencional x RP (Elaborado pelo Autor).....	45
Figura 23: Processo Proposto para Lapidação CNC .....	46
Figura 24: CAD de Modelagem Superficial (NARAYAN et. al., 2008) .....	47
Figura 25: CAD de Modelagem Sólida.....	48
Figura 26: Metodologia geral de trabalho de sistemas CAM.....	50
Figura 27: Módulos do Pós-Processador (Elaborado pelo Autor).....	54
Figura 28: Exemplo de STL versus formato CAD (STL, 2015). ....	55
Figura 29: Arquivo .ASC gerado pelo GemCad® .....	57
Figura 30: Metodologia Proposta do Projeto Pós-Processador .....	61
Figura 31: Padrão de Projeto Abstract Factory .....	72
Figura 32: Padrão Model-View-Controller .....	73
Figura 33: Geração do Modelo de Solução (SUN, 2003).....	76
Figura 34: Protótipo da tela principal do software Pós-Processador.....	78

Figura 35: Protótipo de edição de atividade de manufatura.....	79
Figura 36: Protótipo de tela da edição de Comandos CNC (Elaborado pelo autor) .....	80
Figura 37: Mapeamento de atividade de manufatura x Comando CNC (Elaborado pelo autor) .....	81
Figura 38: Protótipo de tela de inclusão de extensão de CAD,.....	82
Figura 39: Protótipo de tela de inclusão de extensão de CAD,.....	83
Figura 40: Mapeamento de atividade de manufatura x CAD.....	84
Figura 41: Protótipo da interface de especificação da atividade de manufatura (Elaborado pelo autor) .....	85
Figura 42: Protótipo da interface de geração de código NC .....	86
Figura 43: Vista frontal do modelo de máquina de lapidar Faceting Head CNC (Elaborado pela Equipe do Projeto).....	88
Figura 44: Mesa da Lapidadora (Elaborado pela Equipe do Projeto) ....	89
Figura 45: Assistente de Inclusão de Extensões CAD (Elaborado pelo autor).....	91
Figura 46: Extensão do Software GemCad® incluída (Elaborado pelo autor).....	92
Figura 47: Mapeamento de Atividades de Manufatura para elementos CAD (Elaborado pelo autor) .....	94
Figura 48: Inclusão de Comando CNC (Elaborado pelo autor) .....	95
Figura 49: Novo comando CNC incluído no Pós-Processador (Elaborado pelo autor) .....	96
Figura 50: Mapeamento de Atividades de Manufatura com Comandos CNC (Elaborado pelo autor) .....	96
Figura 51: Criação e manutenção de atividades de manufatura (Elaborado pelo autor) .....	97
Figura 52: Especificação das atividades de manufatura (Elaborado pelo autor).....	98
Figura 53: Interface do usuário de geração de código CNC (Elaborado pelo autor).....	101

## LISTA DE TABELAS

Tabela 1: Limites GIA para graduação da simetria de diamantes lapidação RB. ....	30
Tabela 2: Tolerâncias admissíveis para máquina de lapidação (Adaptado de GEURTS et al., 2011) .....	31
Tabela 3: Requisitos Funcionais do Software Pós-Processador (Elaborado pelo autor).....	64
Tabela 4: Requisitos Não-Funcionais do Software Pós-Processador (Elaborado pelo autor).....	66

## LISTA DE ABREVIATURAS E SIGLAS

2D	-----	Duas dimensões, normalmente X e Y
3D	-----	Três dimensões, normalmente X, Y e Z
API	-----	<i>Application Programming Interface</i>
ASCII	-----	<i>American Standard Code for Information Interchange</i>
AWT	-----	<i>Abstract Window Toolkit</i>
BASIC	-----	<i>Beginners All-purpose Symbolic Instruction Code</i>
CAD	-----	<i>Computer Assisted Design</i>
CAE	-----	<i>Computer Aided Engineering</i>
CAI	-----	<i>Computer Aided Inspection</i>
CAM	-----	<i>Computer Assisted Manufacturing</i>
CAPP	-----	<i>Computer Aided Process Planning</i>
CNC	-----	Comando Numérico Computadorizado
DAO	-----	<i>Data Access Object software design pattern</i>
GIA	-----	<i>Gemological Institute of America</i>
GOF	-----	<i>Gang of Four</i>
GRASP	-----	<i>General Responsibility Assignment Software Patterns</i>
GUI	-----	<i>Graphical User Interface</i>
IBGM	-----	Instituto Brasileiro de Gemas e Metais Preciosos
IEEE	-----	<i>Institute of Electrical and Electronics Engineers</i>
ISO	-----	<i>International Organization for Standardization</i>
JNI	-----	<i>Java Native Interface</i>
JOGL	-----	<i>Java Binding for OpenGL</i>
MVC	-----	<i>Model View Controller software design pattern</i>
NC	-----	<i>Numeric Control</i>
OpenGL	-----	Biblioteca Gráfica de Código Aberto
RP	-----	<i>Rapid Prototyping</i>
SRS	-----	<i>Software Requirement Specification</i>
STL	-----	<i>Stereo Lithography Format</i>
UML	-----	<i>Unified Modelling Language</i>
UP	-----	<i>Unified Process software development methodology</i>

## SUMÁRIO

<b>Capítulo 1</b> .....	14
<b>INTRODUÇÃO</b> .....	14
1.1. Justificativa e Relevância da Pesquisa .....	15
1.2. Objetivos .....	17
1.3. Estrutura do Trabalho .....	17
1.4. Resultados Esperados.....	18
<b>Capítulo 2</b> .....	19
<b>REVISÃO BIBLIOGRÁFICA</b> .....	19
2.1. Processo de Lapidção de Gemas .....	19
2.2. Cadeia de Manufatura .....	38
2.3. CAD/CAM.....	46
<b>Capítulo 3</b> .....	59
<b>METODOLOGIA PARA DESENVOLVIMENTO</b> .....	59
3.1. Metodologia UP .....	59
<b>Capítulo 4</b> .....	62
<b>PROJETO DO SOFTWARE</b> .....	62
4.1. Modelo de Requisitos .....	62
4.2. Modelo de Projeto.....	68
4.3. Modelo de Arquitetura.....	69
4.4. Modelo de Solução .....	75
<b>Capítulo 5</b> .....	87
<b>AVALIAÇÃO DAS FUNCIONALIDADES</b> .....	87
5.1. Modelo 3D de Máquina de Lapidar CNC .....	87
5.2. Funcionalidades do software.....	90
<b>Capítulo 6</b> .....	103
<b>CONCLUSÃO</b> .....	103
6.1 Conclusão.....	103
6.2 Perspectivas Futuras.....	104
<b>REFERÊNCIAS</b> .....	106
<b>ANEXOS</b> .....	111
<b>APÊNDICES</b> .....	113

# Capítulo 1

## INTRODUÇÃO

---

O Brasil é mundialmente conhecido pela sua riqueza em pedras preciosas. Segundo a Companhia de Pesquisa de Recursos Minerais (CPRM - Serviço Geológico do Brasil), das nove províncias gemológicas existentes no mundo, das nove regiões geográficas excepcionalmente ricas em gemas, o país é líder não apenas na quantidade produzida, mas também na diversidade. Somente o estado de Minas Gerais contribui com cerca de 25% da produção mundial (Favacho, 2001).

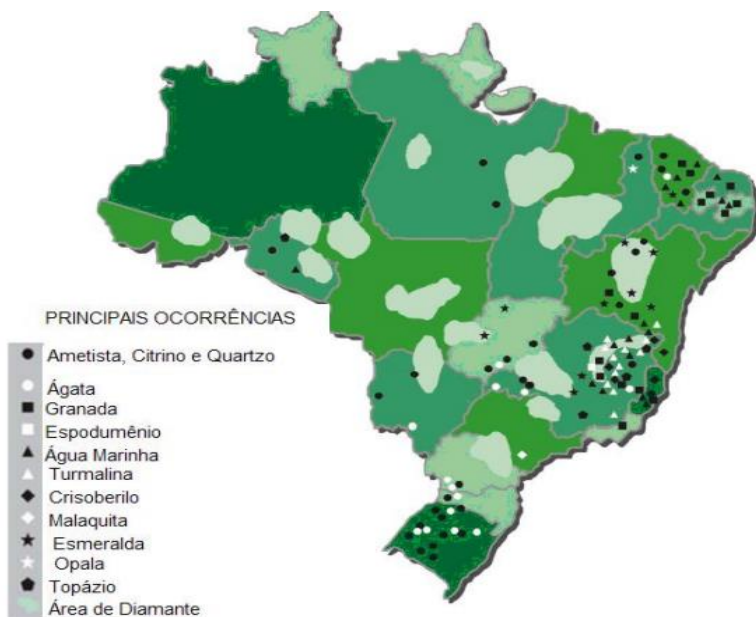
Em contrapartida, não existe no mercado nacional uma grande oferta de máquinas de lapidação que transformem uma gema bruta em uma gema lapidada com a qualidade exigida pelo mercado internacional. No Brasil, mais usualmente, o trabalho de lapidação de maneira geral, é feito de forma artesanal. Isto não oferece os padrões de qualidade competitivos, encontrados nas chamadas pedras calibradas. O lapidário é o único responsável pela qualidade da gema, muitas vezes não sendo possível manter a qualidade em todos os trabalhos devido à irregularidade inerente do ser humano. Tal qualidade se tornaria garantida caso o trabalho fosse realizado por uma máquina. Estes são fatores que têm levado à exportação de pedras brutas para lapidação ao exterior para que lá se faça o processamento adequado da matéria prima.

Na figura 1, a partir de levantamentos do Instituto Brasileiro de Gemas e Metais Preciosos (IBGM), são apresentados os estados produtores e as respectivas pedras que produzem.

Segundo RIBEIRO (2008), “O Brasil é um país abundante em recursos naturais, principalmente de minerais. Dessa forma, é de grande importância que se tomem medidas específicas, adequadas para o uso eficiente desses recursos, de maneira que o país se torne competitivo diante do mercado internacional. Os minerais, quando exportados isoladamente, em sua forma bruta, pouco contribuem para o crescimento da economia nacional. Mas, quando trabalhados, agregando-se valor ao produto, geram riqueza, empregos e divisas”.

Tendo em vista a vasta riqueza reportada e a exploração precária que tem se dado no país, este trabalho segue na direção de desenvolver ferramentas tecnológicas para avançar na direção da eficiência na

manufatura para melhor aproveitamento do material e aumento de produtividade. Há a possibilidade de reduzir perdas relacionadas a falhas de produção, com a possibilidade de simulação dos processos em computador. Outro ponto é a agilidade no processamento entre o software CAD e a obtenção do produto final manufaturado, sem a necessidade de escrever código, possibilitando que o próprio projetista de gemas efetue a geração do programa CNC.



**Figura 1: Mapa gemológico brasileiro. (IBGM, 2005)**

O uso de máquinas-ferramentas CNC abre o caminho para um significativo aumento de produtividade no segmento de lapidação. De acordo com o ministério das indústrias do Paquistão (SMEDA,2013), a produtividade de lapidação manual é de 15 peças/homem por semana, ou seja, 3 peças ao dia para um lapidário. Uma máquina de lapidar automática poderia suplantar este nível de produtividade.

## **1.1. Justificativa e Relevância da Pesquisa**

Uma dificuldade encontrada na área de desenvolvimento de máquinas CNC está relacionada com os problemas de padronização dos

comandos numéricos. No início do desenvolvimento de máquinas comandadas numericamente, a falta de padronização era sentida nas empresas, com várias máquinas fabricadas por diferentes fornecedores, sendo que cada fornecedor tinha uma linguagem própria. Com isso, havia a necessidade de uma equipe técnica especializada para cada tipo de comando, elevando os custos de fabricação e manutenção (GONÇALVES, 2013). O software pós-processador possibilitará uma solução de padronização na área de CNC focado na área de lapidação de gemas. Isto será alcançado através da criação de extensões de entrada, processamento e saída.

As extensões de entrada permitirão ao usuário configurar no software pós-processador todos os formatos de entrada que compõe sua rotina de trabalho, são os formatos de software CAD suportados. Extensões de processamento são as descrições de processo, nelas o usuário irá definir como será o processo de transformação de modelos CAD em código CNC. E, por último, as extensões de saída serão aquelas que o usuário definirá os comandos CNC que executarão a programação gerada.

Outra questão a se considerar é a integração da manufatura em processos de lapidação de gemas. Na lapidação de gemas foi desenvolvido um sistema CAD, o GemCad®, desenvolvido por Robert Strickland, que é um software para o sistema operacional Windows para auxiliar no desenho de modelos de lapidação. Ele simula uma máquina de facetamento e mostra o desenho reproduzido em três dimensões (NADUR, 2009). Este software, como outros CADs para design de gemas, executa somente parte da cadeia de manufatura CAx. O pós-processador proposto irá fazer a integração da manufatura de produção de gemas, trazendo artefatos do sistema CAD até o comando CNC, formando uma cadeia completa de manufatura e fechando a lacuna que há neste segmento da indústria.

Finalmente, pode-se destacar um outro fator que impulsiona o desenvolvimento do pós-processador para gerar automaticamente o programa CNC, que é o aumento da produtividade, aliada à possibilidade de redução de perdas através da simulação das trajetórias, checagem e prevenção de possíveis colisões nas máquinas ou peças durante o processo de lapidação.



## **1.2. Objetivos**

### **1.2.1. Objetivo Principal**

Desenvolver um software pós-processador capaz de interpretar arquivos de software CAD (*Computer Aided Design*) da área de lapidação de gemas para gerar programa CNC específico da máquina que irá executar a lapidação.

### **1.2.2. Objetivos Específicos**

Como desdobramento do objetivo principal, os seguintes itens serão trabalhados no projeto:

- Projeto e implementação de arquitetura de software extensível;
- A arquitetura extensível será desenvolvida de maneira a cumprir com os requisitos de múltiplos formatos de arquivos de entrada (CAD) e saída (máquinas CNC);
- Desenvolvimento de uma linguagem para descrição do processo que não seja dedicada a um comando CNC.

## **1.3. Estrutura do Trabalho**

Este trabalho está organizado em 6 capítulos. No primeiro capítulo é feita a introdução ao tema, justificando a construção do software pós-processador para lapidação automática de gemas no panorama atual do mercado brasileiro nesta área.

No segundo capítulo apresenta os elementos da área de tecnologia de lapidação de gemas, os quais usa-se para levantar os requisitos da construção do pós-processador sob a forma de revisão bibliográfica.

O terceiro capítulo discute a metodologia selecionada para o desenvolvimento de software, descreve as fases pelas quais serão passadas e os artefatos de que serão produzidos em cada uma delas.

O quarto capítulo trata do projeto de software elaborado para cumprir com os requisitos levantados na fase de Iniciação da metodologia UP.

O quinto capítulo avalia as funcionalidades dos artefatos construídos na fase de desenvolvimento baseados na fase de elaboração da metodologia selecionada.

O sexto capítulo apresenta as conclusões do trabalho e traz perspectivas de trabalhos futuros, visando a melhoria e continuidade do

trabalho desenvolvido.

## **1.4. Resultados Esperados**

Espera-se obter um programa CNC com comandos para automatizar a manufatura da lapidação de gemas gerado pelo software pós-processador. Este programa deverá estar de acordo com o arquivo CAD de entrada e com a especificação de máquina e controle CNC feitas no software. A arquitetura do software deverá permitir a extensão automática a partir de opções feitas pelo usuário de diferentes sistemas CAD, geometrias de máquina e padrões de linguagem de comando CNC.

Durante o ciclo de desenvolvimento do software haverá o levantamento dos requisitos. Entre os requisitos, haverá alguns que terão seu desenvolvimento obrigatório e outros que serão apenas desejados. Aqueles que terão desenvolvimento obrigatório deverão ser desenvolvidos na íntegra no software e serão avaliados pelo grau de adequação do artefato gerado para o requisito solicitado.

Haverá também o desenvolvimento de um modelo de máquina virtual de lapidação em software CAD, o qual será usado como base geométrica para a execução do código CNC baseado em um comando CNC.

Sendo assim, ao final do projeto será possível obter um código CNC gerado a partir de um software CAD específico para a área de lapidação de gemas. Este código será baseado na geometria da máquina de lapidar desenvolvida, que permitirá a rápida adequação de novos modelos de lapidação (*designs*).

Isto irá criar um panorama de resultados úteis na geração de uma cadeia de manufatura para lapidação de gemas com o uso de máquinas automáticas de comando numérico computadorizado.

# Capítulo 2

## REVISÃO BIBLIOGRÁFICA

---

Neste capítulo serão explorados alguns fundamentos para o desenvolvimento do software pós-processador. Eles darão subsídio para o levantamento dos requisitos e a respectiva implementação dos mesmos. Tudo isto resultará nas funcionalidades requeridas para a criação de um software que terá as características necessárias para suprir necessidades de empresas e profissionais da área de lapidação de gemas.

Primeiramente, será estudado a cadeia de manufatura da produção de produtos em geral, mostrando especificamente a cadeia automatizada com uso de tecnologia CNC. A seguir, vai se explorar a adequação que pode ser feita na cadeia de manufatura CNC para usá-la na lapidação de gemas, passando pela observação da prototipagem rápida e mostrando similaridades que esta cadeia tem com relação a proposta de uma cadeia para lapidação de gemas.

O próximo elemento que será estudado é o uso das ferramentas CAD e CAM para automação de parte da cadeia da manufatura. O software pós-processador trabalha situado no mesmo nível dos sistemas CAM e baseia-se em modelos CAD. É feito também um levantamento das tecnologias e padrões de arquivos de entrada que serão inicialmente desenvolvidas com o software pós-processador. Estes arquivos são relativos a sistemas CAD, que serão lidos do software, e padrões de código CNC que serão gerados por ele.

Para finalizar, será feito um estudo do processo completo de lapidação de gemas, incluindo um levantamento do estado da arte das tecnologias usadas neste segmento e os parâmetros que levam às tolerâncias para construção de máquinas de lapidação modernas.

### **2.1. Processo de Lapidação de Gemas**

Zanatta et. al. (2010) colocam que o beneficiamento das gemas a partir do seu estado bruto é realizado pelo processo de lapidação, com o objetivo de aperfeiçoar suas qualidades ópticas, aumentar seu brilho, definir suas formas, realçar suas cores e belezas naturais. Também citam que é possível encontrar que um processo de lapidação executado com perfeição pode dar maior valor ao material bruto ao se comparado a um com lapidação feita de forma incorreta. Uma lapidação considerada incorreta possui facetas desalinhadas, proporções fora do padrão, e

polimento irregular.

Nesta seção serão explorados os seguintes tópicos: as gemas e algumas técnicas de lapidação, o processo de lapidação na técnica facetada, máquinas de automação do processo de lapidação e as tolerâncias admissíveis na produção de máquinas de lapidação de gemas.

### 2.1.1. Gemas Lapidadas

Gema ou pedra preciosa é um mineral, material petrificado ou uma rocha que possui atributos que, ao passar pelo processo de lapidação, tornam-se produtos mercadológicos usados para adorno em pessoas. De acordo com Schumann (2006), não existe uma definição aceita por todos para este termo, porém há um denominador comum, todas as gemas têm algo especial, alguma beleza em torno delas. Elas são principalmente minerais (ex.: diamante), minerais agregados (ex.: jade), ou, mais raramente, rochas (ex.: lápis-lazúli). Algumas são de origem orgânica (como âmbar, coral e pérola) e outras de origem sintética.

Existem diversas técnicas de lapidação na atualidade. Entre as técnicas mais difundidas estão a lapidação cabochão e a lapidação facetada. A lapidação cabochão é a técnica mais antiga de lapidação, e consiste de dar formas arredondadas para gemas, um alisamento e polimento para dar brilho.



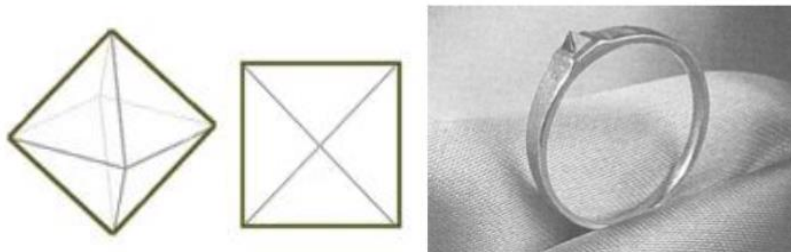
**Figura 2 - Lapidação Cabochão (SCHUMANN, 2006)**

A lapidação cabochão mostrada na figura 2 eleva a sua beleza através da exposição de um fenômeno da lapidação conhecido como asterismo, cujo efeito é obtido através da técnica que aproveita formações existentes no material. Outros destes fenômenos são: olho de gato ou *chatoyance*, jogo de cores, entre outros.

A técnica de lapidação facetada é usada normalmente em materiais translúcidos, onde se cria facetas em ângulos específicos de um

determinado material que irão aumentar o brilho através da reflexão da luz. Segundo Nadur (2009) a técnica de lapidação facetada teve início na Índia por volta de 400 a.C. Esta técnica que não teve grandes evoluções até o século XIII quando os mercadores venezianos que passaram por lá e a trouxeram para Europa.

Na figura 3 pode-se ver as lapidações facetadas mais primitivas.



**Figura 3 - Lapidação Facetada  
(NADUR apud ALLABOUTGEMSTONES, 2009)**

O projeto do pós-processador terá como foco a lapidação facetada, pois esta lapidação atinge um mercado maior do que a lapidação cabochão e é também mais desenvolvida tecnologicamente na atualidade. Exemplos do desenvolvimento são os diagramas de facetamento produzidos em softwares CAD e as máquinas modernas para se lapidar facetas que serão apresentadas na próxima seção.

### 2.1.2. Máquinas de Lapidação de Gemas

A pesquisa realizada nesta seção traz consigo um relato histórico do desenvolvimento da arte de lapidar pedras preciosas durante os tempos pelo homem. Esta arte transforma a pedra bruta retirada da natureza, onde foi gerada pela associação de elementos químicos orgânicos (para materiais como o âmbar) ou inorgânicos (pedras coradas, diamante, etc.), em um produto de extrema beleza que encanta pessoas durante milênios.

Os processos de lapidação evoluíram de uma forma totalmente artesanal para máquinas de alta tecnologia como pode ser constatado nas técnicas descritas a seguir:

#### a) Polimento de minerais

No princípio havia o uso de materiais abrasivos, com uma espécie de lixa onde somente dava-se um brilho nas pedras. Na figura 4 temos uma ilustração datada da idade média que dá uma ideia do processo.



**Figura 4 - Polidor de minerais (NADUR aput MONTAGNA, 2009)**

- b) Tecnologia “*Bow-driven*” ou lapidação Indiana com disco abrasivo e arco

Os indianos melhoraram um pouco, com o uso de instrumentos rudimentares, a lapidação feita somente com lixas, com a introdução do disco. O disco deles funcionava com energia motriz humana, ou seja, era totalmente manual. Usa-se as mãos e um arco para movimentar o disco neste sistema rudimentar, que ainda hoje é utilizado.

Países como a Índia e o Sri Lanka, lapidam da mesma forma desde 1800, ideal apenas para gemas coradas, na qual os lapidadores seguram a pedra próxima à roda, comumente mal posicionadas, e com o outro braço movimentam a roda através de um arco, chamada de “*Bow-driven*”. Esta lapidação é chamada “*Native-cut*”. (HURLBUT E KAMMERLING, 1991). Este processo é ilustrado na figura 5.



**Figura 5 - Lapidação Indiana. (NADUR apud BRUTON. 1978)**

c) Tecnologia “Jamb-peg”. Por volta de 1960, o simples instrumento usado para estabelecer o ângulo das facetas, chamado “*Jamb-peg*”, ou seja, com o *dopstick* encaixado em um dos furos, era posicionado com o ângulo desejado e o material desgastado pela roda horizontal umedecida com água para sua refrigeração.



**Figura 6 – Lapidário usando máquina com graduador  
(Foto tirada pelo autor).**



**Figura 7 - Graduador (Foto tirada pelo autor).**

c) Tecnologia “*Faceting-Head*”, ou máquina mecânica com cabeçote de índices e discos abrasivos.

Estas máquinas são dotadas de motores elétricos, como as máquinas “*Jamb-peg*”. Entretanto, possuem um cabeçote de lapidação que possui controle de altura, de ângulo, e um índice de rotação (foi dividido a circunferência em partes chamadas índice). Isto significa que foi uma evolução, especificamente direcionada ao cabeçote. Mesmo assim, todo esse controle ainda é mecânico. Estas máquinas são as mais usadas em países desenvolvidos na atualidade, onde há disponibilidade delas a preços bem acessíveis e com boa qualidade. No Brasil há empresas que comercializam suas máquinas “*Faceting-Head*”.

Nos anos 2000, as máquinas de lapidação no mercado eram as chamadas “*Faceting-head*”, de origem dos Estados Unidos da América (NADUR, 2009).

Constituídas por um braço lateral onde é acoplado o “*dopstick*”, agora metálico, com posicionamento de angulações. O braço possui uma rotação em duas direções, vertical e horizontal (NADUR apud HURLBUT E KAMMERLING, 2009). A rotação vertical é o ajuste do ângulo de lapidação e a horizontal é o índice de lapidação ambos ajustados no cabeçote da máquina.

Já existem máquinas deste tipo que são dotadas de alguma automação, por exemplo, a máquina da empresa norte-americana UltraTec (2013), modelo V5, que é dotada de um cabeçote com o ângulo



automatizado. É possível, através de um display de cristal líquido e botões, fazer o ajuste e visualização do ângulo de lapidação.

Um modelo de lapidadora “*Faceting-head*” é mostrado na figura 8.



**Figura 8 - Máquina de lapidar "*Faceting-head*"  
(ULTRATEC INC. 2013)**

d) Tecnologia CNC – Controle Numérico Computadorizado e Laser

A técnica CNC, composta por códigos de posicionamento em 3 dimensões, é usada em várias aplicações na indústria, e foi incluída recentemente na joalheria e na lapidação. Como elemento cortante, utiliza-se brocas ou fresas diamantadas, chamadas de *Router*, laser, plasma, ou jato d’água em alta pressão.

Já são conhecidos alguns fabricantes de lapidadoras usando o processo CNC, alguns usando discos abrasivos e outros com outras ferramentas de corte. O uso do disco abrasivo torna o desenvolvimento CNC bastante particular, com mudanças significativas na cinemática das máquinas quando adaptadas para lapidação.

No Brasil existem dois registros de máquinas CNC na área de lapidação. A primeira foi a tentativa de construir uma máquina computadorizada, em 2005, pelo Centro Universitário UNIVATES, em Lajeado, Rio Grande do Sul, em parceria com uma empresa privada, RW Importação e Exportação Ltda. e financiamento da FINEP (Financiadora de Estudos e Projetos). A etapa inicial deste projeto previa a introdução de tecnologia para lapidação em grande escala, dos denominados cabochões, calibrados e padronizados, utilizando pedras preciosas de baixo valor (ametista, ágata e citrino). A introdução desta tecnologia traria

como consequência a redução do custo de lapidação e o aumento da competitividade neste mercado (NADUR apud KLIPPEL, 2007).

O fato é que empresas que detém a tecnologia CNC para corte de pedras preciosas ainda são poucas e o custo ainda é muito elevado, normalmente são usadas apenas para pedras caras como o diamante. Como a máquina mostrada na figura 9.



**Figura 9 - Máquina CNC para lapidação de diamante.  
(SOLANKI MACHANIC WORKS INC, 2013)**

- e) Tecnologia “*Faceting-Head*” automática com cabeçote de índices e discos abrasivos

Esta máquina é a evolução da “*Faceting-head*”, que possuía somente o motor que automatizava a rotação do disco com energia elétrica. A máquina automática já faz a automação quase que completa do processo de produção de gemas lapidadas. A partir de uma pedra formada em um “dopstick”, inserida no cabeçote de lapidação, selecionado o diagrama de lapidação, parametrizado as dimensões da pedra formada entregue à máquina, tem-se a lapidação da gema feita de forma automática. Isto tudo com muita precisão devido a automação dos processos repetitivos, os quais normalmente ocorrem falhas humanas. Tal tecnologia tem um custo mais acessível, entretanto é pouco explorada em território brasileiro.

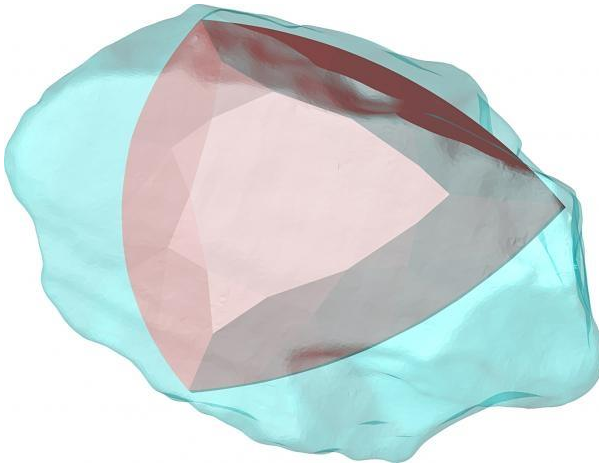


**Figura 10 - Máquina automática com cabeçote e discos.  
(KLM TECHNOLOGY, INC. 2013)**

Na Internet, foi localizada uma máquina CNC de lapidação “Faceting-Head” automática, da américa-coreana KLM. Há dois modelos, uma chamada JANG 1 (cuja foto é mostrada na figura 10) e a JANG 2. A primeira faceta uma gema por vez, utilizando-se de dois discos paralelos e sobrepostos (um para facetar, outro para polir), ambos refrigerados a água; e outro para lapidar doze pedras idênticas simultaneamente, que são colocadas lado a lado, linearmente em um “pente” de aço. O software CAD/CAM utilizado, foi desenvolvido pela própria empresa e possibilita inúmeros formatos de desenhos de gemas (NADUR apud ASPAHAN, 2008).

Outra máquina CNC com tecnologia avançada é a que foi desenvolvida pelo instituto Fraunhofer para Matemática Industrial (ITWM) em Kaiserslautern, Alemanha. Este instituto desenvolveu uma máquina de lapidação CNC de 17 eixos que fornece uma precisão de 10 micrometros. Ela tem sido divulgada como muito econômica, pois permite a economia de mais de 30% da pedra bruta. Isto ocorre pelo fato de que a atividade de desbaste, detalhada mais adiante, ser feita através de software e hardware. Existe um hardware que escaneia a pedra bruta e um software que otimiza o uso maximizado do material (FRAUNHOFER ITWM, 2014) veja a figura 11 a seguir que mostra uma imagem 3D onde

foi aplicado o algoritmo de otimização do uso da pedra bruta.



**Figura 11 - Depois de escaneado, o software calcula o corte otimizado (FRAUNHOFER ITWM, 2014)**

Nota-se a partir deste estudo que na atualidade a tecnologia CNC é a que mais tem sido empregada para automatizar o processo de lapidação, sendo que a automação está retirando as máquinas semi-automáticas do mercado e inserindo maior rapidez e qualidade nas fábricas.

Estas são as tecnologias das máquinas que evoluíram com o tempo, partindo do trabalho manual artesanal e chegando nos dias de hoje com alto nível de automação fornecido pelas máquinas CNC.

### 2.1.3. Parâmetros Físicos para Construção de Máquinas de Lapidação

Para se chegar às tolerâncias aceitáveis na fabricação de gemas e assim poder obter parâmetros de precisão dos eixos de uma máquina CNC dedicada à fabricação de pedras preciosas, um estudo foi realizado onde foi verificado que para classificar as gemas e ter parâmetros de qualidade é necessária uma referência pré-estabelecida de parâmetros, já que isto é mais simples do que criar um novo procedimento de classificação e parametrização. Devido ao fato de não existir um padrão estabelecido para pedras preciosas de valor mais baixo, mostra-se como uma boa opção

usar padrões estabelecidos para o diamante. O diamante, por ser um produto bastante importante para várias indústrias, disponibiliza estudos aprofundados com relação à sua qualidade e pode ser usado como referência de qualidade geométrica de pedras acabadas.

Nos Estados Unidos da América há um instituto dedicado a padronizações relacionadas a materiais gemológicos, é o Instituto Gemológico da América do Norte (GIA – *Gemological Institute of America*). Este instituto tem feito um esforço na classificação de diamantes e gerou o Sistema de Graduação de Lapidação de Diamantes da GIA. Este sistema é usado para graduar diamantes cortados no modelo Brilhante Redondo, que é o design mais conhecido e popular de corte de diamantes. Ele pode ser visto na figura 12.

A GIA tem feito estes estudos para ajudar membros do mercado joalheiro a entender melhor os atributos da aparência de diamantes, e como estes atributos têm sido avaliados dentro do sistema de graduação deles.



**Figura 12: Lapidação Brilhante Redondo (GEURTS et al., 2011)**

No sistema da GIA são levados em conta os seguintes atributos da peça: simetria; polimento; tamanho da Culeta; e espessura da Cintura.

Os atributos de polimento mostram uma classificação qualitativa e ocorre em testes visuais dos diamantes analisados através da aparência da vista superior, que é onde ocorre a reflexão da luz nas peças lapidadas. O polimento, é avaliado visualmente com o uso de ampliação de 10X. Trata-se de um procedimento ainda bastante qualitativo e mais detalhes podem ser encontrados no artigo de Blodgett (2009).

Os atributos relacionados com a simetria, tamanho da Culeta e

espessura da Cintura são, ao contrário daqueles relacionados com o polimento, quantitativos. Eles são importantes para a parametrização de máquinas de lapidação, e, portanto, para este trabalho.

Há aspectos da simetria que são relacionados com proporções, como apresentado por Geurds et. al. (2009), que dividiu estes parâmetros quantificáveis em 10 para fazer a seleção de diamantes, e estes dados são interessantes para serem usados na construção de máquinas.

Estas características de simetria podem ser medidas com confiabilidade suficiente por um scanner óptico sem contato para determinar o grau de simetria em lapidações Brilhantes Redondas (RB) para diamantes (GEURDS, et al., 2009).

A Tabela 1 mostra a classificação GIA para a graduação de simetria de diamantes lapidados com o design Brilhante Redondo e suas faixas limite para cada grau de simetria:

**Tabela 1: Limites GIA para graduação da simetria de diamantes lapidação RB. (GEURTS et al., 2011)**

Parâmetro	Excelente	Muito Bom	Bom
Desvio de Circunferência (%)	0-0.9	1.0-1.8	1.9-3.6
Mesa Fora de Centro (%)	0-0.6	0.7-1.2	1.3-2.4
Culeta Fora de Centro (%)	0-0.6	0.7-1.2	1.3-2.4
Alinhamento Mesa/Culeta (%)	0-0.9	1.0-1.8	1.9-3.6
Variação da altura da Coroa (%)	0-1.2	1.3-2.4	2.5-4.8
Variação do ângulo da Coroa (°)	0-1.2	1.3-2.4	2.5-4.8
Variação da profundidade do Pavilhão (%)	0-1.2	1.3-2.4	2.5-4.8
Variação do ângulo do Pavilhão (°)	0-0.9	1.0-1.8	1.9-3.6
Variação da espessura da Cintura (%)	0-1.2	1.3-2.4	2.5-4.8
Variação do tamanho da Mesa (%)	0-1.2	1.3-2.4	2.5-4.8

Com os dados das dimensões dos materiais a serem usinados é possível calcular as faixas de tolerâncias aceitáveis para a máquina de

lapidação. Os dados dimensionais são importantes, pois de acordo com o tamanho de pedras a ser usinado a escala de precisão varia proporcionalmente, sendo que as pedras menores exigem maior grau de precisão do que as pedras maiores. As faixas de tolerância do GIA são mostradas na tabela a seguir.

**Tabela 2: Tolerâncias admissíveis para máquina de lapidação  
(Adaptado de GEURTS et al., 2011)**

Parâmetro	Tamanho (mm ou °)	Tolerância Max (%)	Varição ( $\mu\text{m}$ ou °)
Desvio de Circunferência (%)	4	1.8	72
Mesa Fora de Centro (%)	-	1.2	48
Culeta Fora de Centro (%)	-	1.2	48
Alinhamento Mesa/Culeta (%)	-	1.8	72
Varição da altura da Coroa (%)	0,162	2.4	96
Varição do ângulo da Coroa (°)	34,5	2.4	0,024
Varição da profundidade do Pavilhão (%)	1,724	2.4	96
Varição do ângulo do Pavilhão (°)	40,75	1.8	0,018
Varição da espessura da Cintura (%)	0,1	2.4	4
Varição do tamanho da Mesa (%)	2,2	2.4	96

Mesmo se tratando de referências dimensionais para diamantes, é possível extrair dados importantes da tabela independentemente do material. Nota-se que uma máquina para atingir a precisão necessária para lapidação deverá trabalhar na casa dos micrometros. Pode-se verificar isto a partir do parâmetro "Varição da espessura da Cintura", que tem o valor de 4  $\mu\text{m}$ . Este dado vai integrar uma restrição dos eixos lineares da máquina de lapidação, que devem ter resolução em escala micrométrica para atender este requisito.

Outro dado importante é com relação aos ângulos, o item "Varição do ângulo do Pavilhão", que apresenta um valor de 0,018 graus, que vai restringir a resolução dos eixos angulares da lapidadora.

A conclusão a respeito das restrições apresentadas é que uma máquina de lapidação CNC, seguindo os elevados padrões de mercado, exigiria um grande investimento financeiro para ser construído devido ao

alto grau de precisão requerido em seus eixos. Por isto, diante do cenário de recursos disponíveis para o projeto, escolheu-se para este projeto construir o software pós-processador e eventualmente fazer e simulações computadorizadas com modelos 3D ao invés de construir um protótipo de máquina de lapidação CNC.

#### 2.1.4. Processo de Lapidação com Máquinas Semi-Automáticas

Segundo Nadur apud Sinkankas (2009), fazendo-se uma adaptação do processo de lapidação de gemas, determina-se as seguintes etapas na lapidação de gemas:

- 1) Desenvolvimento ou escolha de um modelo de lapidação;
- 2) Desbaste;
- 3) Serragem;
- 4) Pré-formação;
- 5) Encanetamento;
- 6) Facetamento;
- 7) Polimento;
- 8) Descolagem;
- 9) Limpeza.

A única adaptação feita do processo de lapidação apresentado por Nadur (2009) foi a inclusão da atividade de desenvolvimento ou escolha de um modelo de lapidação. Esta atividade foi incluída levando-se em conta a introdução do computador no processo. Lapidários experientes já têm usado softwares de desenho CAD, como o GemCad®, para auxiliar no desenvolvimento da lapidação. Ele desenvolve seu desenho ou escolhe um desenho disponível na Internet, e tendo como ponto de partida este desenho, o mesmo faz uso de máquinas de lapidar semi-automáticas para executar a lapidação de acordo com instruções fornecidas pelo desenho escolhido. O desenho e as instruções serão analisadas em detalhes na seção 2.1.5.

A atividade de desbaste é feita golpeando-se a pedra com um martelo para retirar as falhas encontradas, o que produzirá uma pedra com as arestas quebradas com somente o material interessante para lapidação. É feita então uma análise das propriedades físicas da pedra para encontrar a melhor posição da mesa, ou seja, a posição em que a luz incidirá sobre a pedra quando lapidada. A figura 13 ilustra como fica uma gema após a atividade de desbaste. Note que não há impurezas na pedra apresentada, pois todas devem ser retiradas neste processo.

A serragem é feita após o desbaste. Durante o desbaste foi feito um



projeto de como a pedra seria lapidada e isto dará o direcionamento para a serragem, que tem por objetivo fazer com que a pedra fique rapidamente próxima ao formato desejado, com os contornos de acordo com o que foi escolhido no desenho de lapidação. A serragem é feita com uma serra diamantada com núcleo de cobre ou bronze e refrigerada a água ou óleo.



**Figura 13: Pedra bruta após desbaste (NEVADA, 2015)**

Na pré-formação são usadas rodas ou rebolos de lapidação de granulometria grossa para fazer uma remoção rápida de material, melhorando a forma produzida pela serra em direção ao projeto elaborado na atividade de desbaste. A refrigeração da atividade é feita com água. O rebolo tem sua granulometria variando de acordo com a dureza da gema a ser lapidada. Quanto mais dura for a pedra, maior será a granulometria do rebolo.

Após a pré-formação a gema ficará como ilustrado na figura 14 (nota-se que o desenho de lapidação escolhido é uma gota para pingente) e estará pronta para a próxima atividade que é o encanetamento.



**Figura 14: Gema Após a Pré-Formação (NEVADA, 2015)**

Para executar a atividade de encanetamento é necessário colar a pedra em um elemento chamado de vareta ou “*dopstick*”, feito de de latão ou alumínio. A cola usada pode ser Epoxy ou mistura contendo goma-laca que é chamada de lacre, sendo que usualmente é feita primeiramente a lapidação da coroa e depois o pavilhão da gema. Isto significa que durante o processo haverá uma descolagem da pedra, já que primeiro será lapidada a parte superior (coroa) e depois a pedra é virada para ser lapidada a parte inferior (pavilhão) (Nadur, 2009).

A figura 15 à esquerda, mostra a pedra após ter sido encanetada, e à direita apresenta a atividade de transferência que deve ocorrer após uma das partes da pedra ter sido lapidada. A colagem com lacre exige que este seja fundido com auxílio da chama de um pequeno maçarico.



**Figura 15: Gema Após o Encanetamento e Transferência (NEVADA, 2015)**

O facetamento é a próxima etapa após o encanetamento. Isto é realizado na máquina de lapidação como visto na figura 16. O facetamento em máquinas “*Faceting Head*” segue o desenho de lapidação mostrado na seção 2.1.5. Neste diagrama são encontrados os passos do facetamento com as variações do ângulo do cabeçote e do índice de lapidação que devem ser ajustados durante esta atividade.



**Figura 16: Atividade de Facetamento (NEVADA, 2015).**

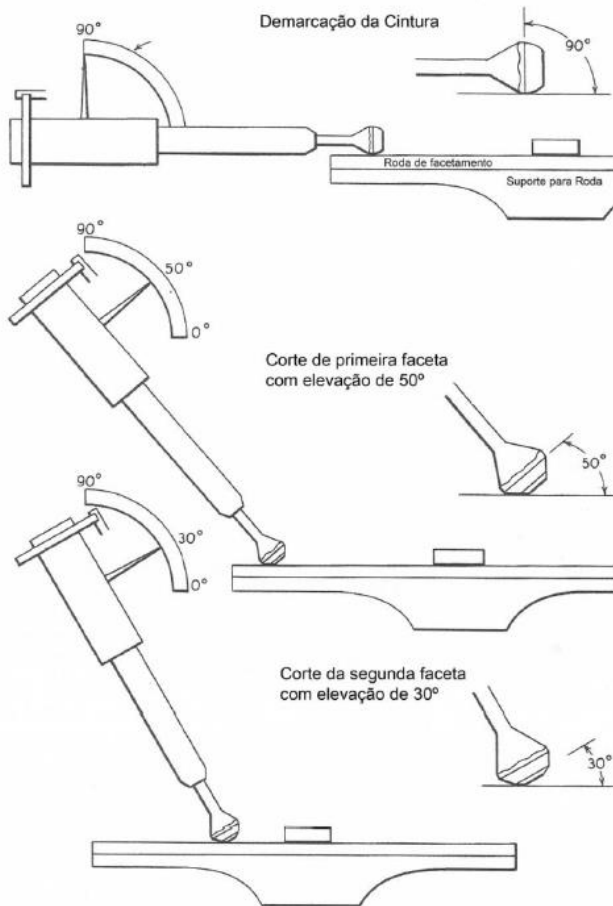
Segundo Nadur (2009), o facetamento é feito em discos de alumínio, aço ou cobre para quartzos, que são minerais de dureza relativamente baixa. Para materiais mais duros que o quartzo, como berilo, coríndon e turmalina, deve-se usar discos diamantados por pulverização, eletrodeposição ou fundição do metal ao redor de partículas de diamante. Este disco também é chamado disco diamantado. A figura 17 mostra um resumo da etapa de facetamento.

A etapa de polimento usa discos similares aos discos de facetamento e a mesma máquina é usada para esta atividade. A diferença básica é que o disco de facetamento será substituído na máquina por um disco específico de polimento contendo os pós de polimento (trípoli, alumina, óxido de cério, entre outros).

O disco de polimento usa pós com granulometria mais fina, o qual fará o acabamento final, deixando a pedra em um estado de polimento em que não haverá riscos visíveis nas suas superfícies. Isto permitirá que a luz entre através das facetas criadas na coroa e possam ser refletidas no pavilhão, criando o efeito de brilho desejado pelo lapidário.

Após o polimento é feita a transferência da pedra para se poder trabalhar a sua parte inferior. É neste momento que é feito o pavilhão, como indicado na figura 15 à direita.

A próxima etapa é a descolagem, cujos passos dependem do material usado para colar a pedra na vareta. Ao usar o lacre, o mais usual é resfriar a pedra em um congelador por alguns minutos. A retração de materiais com índices de dilatação diferentes faz com que a pedra se solte facilmente e a descolagem estará concluída.



**Figura 17: Resumo da Atividade de Facetamento (NADUR apud SINKANKAS, 2009)**

Para finalizar, é preciso limpar a pedra, retirando todo o material deixado, especialmente o pó usado no polimento. Para isto, assim que a pedra é descolada, é simplesmente usado álcool.

### 2.1.5. Desenho de Lapidação

Na figura 18 é mostrado um diagrama com um desenho (do inglês *design*) de lapidação. Este *design* da figura é conhecido como lapidação brilhante, o qual pode ser gerado em ferramentas CAD

específicas para lapidação. De acordo com Strickland (2002), que é o autor do software GemCad®, um dos mais populares do setor, no topo do diagrama pode-se ver quatro perspectivas de visualização do design. Ao lado das perspectivas tem-se uma tabela com uma parametrização obrigatória para o design específico de lapidação. Na parte inferior do diagrama há outra tabela, que indica as instruções de facetamento. Estas instruções especificam os ângulos do cabeçote de lapidação e a configuração de índice de lapidação em duas fases. A primeira fase mostra o corte do pavilhão (parte superior da gema, em inglês *pavilion*), e a segunda fase mostra o facetamento da coroa (parte inferior da gema, em inglês *crown*). Note também que há camadas de corte, com cada uma em um ângulo estabelecido. No pavilhão, por exemplo há 45, 90 e 43. O ângulo de 90 graus é chamado de cintura da gema.

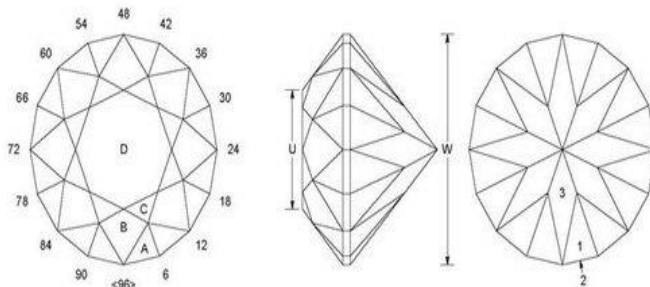
Nas quatro perspectivas mostradas na figura 18 há algumas dimensões rotuladas. Estas dimensões normalmente mostram:

- W – Largura da gema (em inglês *width*);
- L – Comprimento da gema (em inglês *length*);
- C – Altura da coroa;
- P – Profundidade do pavilhão;
- T – Tamanho da mesa;
- U – Comprimento da mesa;

Na tabela de parametrização obrigatória mostrada no diagrama tem-se os seguintes dados:

- Número de facetas, total de 73 para o design SRB;
- Ângulos para R.I. = 1,540 – índice de refração;
- L/W – Razão entre comprimento e largura (*length/width*);
- P/W – Razão entre pavilhão e largura (*pavilion/width*);
- C/W – Razão entre coroa e largura (*crown/width*);
- T/W – Razão entre mesa e largura (*table/width*);
- U/W – Razão entre comprimento da mesa e largura;
- Vol./W<sup>3</sup> – razão entre volume da gema e a largura ao cubo.

Estas razões deverão ser executadas para se produzir uma gema classificada como calibrada, a qual teria melhor aceitação no mercado. Estas relações necessitam de requisitos de projeto mecânico bastante rígidos nas especificações dos componentes da máquina para que não ocorram folgas, o que faria com que a máquina não tivesse precisão nos movimentos de seus eixos.



### Standard Round Brilliant (SRB)

Angles for R.I. = 1,540  
 57 + 16 girdles = 73 facets  
 8-fold, mirror-image symmetry  
 96 index  
 $L/W = 1,000$   $T/W = 0,516$   $U/W = 0,516$   
 $P/W = 0,466$   $C/W = 0,218$   
 $Vol./W^3 = 0,255$

#### PAVILION

1	45.00°	03-09-15-21-27-33-39-45- 51-57-63-69-75-81-87-93	Pavilion break facets
2	90.00°	03-09-15-21-27-33-39-45- 51-57-63-69-75-81-87-93	Girdle facets
3	43.00°	96-12-24-36-48-60-72-84	Pavilion main facets

#### CROWN

A	47.00°	03-09-15-21-27-33-39-45- 51-57-63-69-75-81-87-93	Crown break facets
B	42.00°	96-12-24-36-48-60-72-84	Crown main facets
C	27.00°	06-18-30-42-54-66-78-90	Star facets
D	0.00°	Table	Table facet

**Figura 18: Diagrama de Lapidação design Brillhante Redondo (STRICKLAND,2002)**

Nota-se que o RI no diagrama apresentado na figura 18 é 1,540, que é o índice de refração usado especificamente para quartzo. O ângulo usado para o pavilhão foi 43° para faceta principal, o que é uma aproximação do ângulo crítico. A partir destes parâmetros é possível obter o máximo de brilho

## 2.2. Cadeia de Manufatura

A partir da definição de cadeia de manufatura haverá um

direcionamento, com base no estudo de tecnologias existentes, para a geração de uma cadeia específica para a lapidação de gemas. A criação de um processo automático para lapidação oriundo da manufatura moderna torna-se uma necessidade para que, seguindo estes conceitos, tenha-se fundamentos para criação dos requisitos funcionais do software pós-processador. Será feita a exploração da cadeia de manufatura, para depois mostrar um paralelo com os processos existentes, especialmente o processo da prototipagem rápida, e ao final desta seção é sugerida a geração de um processo específico para lapidação de gemas baseado no conhecimento dos processos existentes.

### 2.2.1. Definição

Souza e Ulbrich (2013), definiram cadeia de manufatura como a integração de atividades e processos em uma cadeia de fabricação de produtos específicos; projeto, materiais, processo, fabricação, qualidade, gestão e marketing. Isto, dito resumidamente, é o ciclo de vida de um produto durante sua manufatura.

No trabalho destes autores, a cadeia de manufatura pode ser dividida em dois níveis:

1. Sistemas de gestão e planejamento de produção, onde encontram-se os processos relacionados à engenharia de produção;
2. Sistemas de processo de fabricação, que são relacionados às atividades de processo de fabricação do produto.

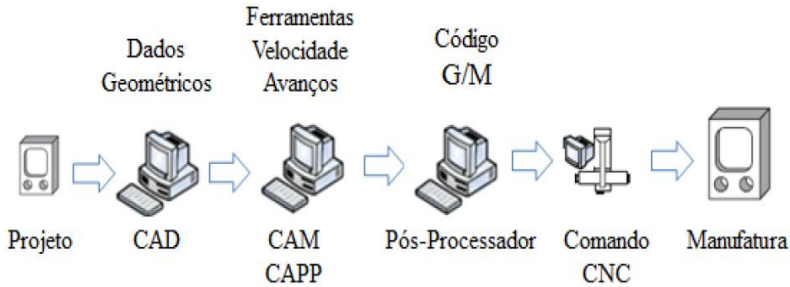
Os dois sistemas citados acima possuem ferramentas computacionais que auxiliam na execução de seus processos. Este trabalho tem o contexto relacionado às ferramentas computacionais ligadas a sistemas de processo de fabricação.

Os sistemas de processo de fabricação, dentro da cadeia de manufatura, podem ser auxiliados por sistemas computacionais chamados CAx, dentre estes sistemas pode-se destacar:

- CAD (*Computer Aided Design*) - Software aplicativo dedicado ao desenho de produtos;
- CAM (*Computer Aided Manufacturing*) - Software aplicativo para auxiliar na fabricação ou manufatura do produto;
- CAE (*Computer Aided Engineering*) - Software aplicativo para auxiliar simular a engenharia do produto;
- CAI (*Computer Aided Inspection*) - Software aplicativo para auxiliar na validação, na inspeção de formas geométricas; e
- CAPP (*Computer Aided Process Planning*) - Software aplicativo para

auxiliar no planejamento de processos de fabricação.

Gonçalves (2013), destaca a cadeia mais comum CAx utilizada na fabricação. Esta cadeia pode ser vista na figura 19.



**Figura 19: Cadeia de produção CAx mais comum (GONÇALVES, 2013)**

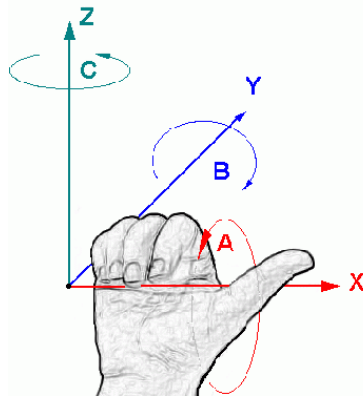
Na figura 19 mostra-se que os requisitos de projeto são transferidos para um CAD, onde a geometria do produto é desenvolvida para atender tais requisitos e armazenada em um arquivo de computador. A seguir, o arquivo é importado em um aplicativo CAPP ou um aplicativo CAM. Neste nível são definidas as velocidades de avanço, as ferramentas e estratégias de usinagem para fabricação do produto. Normalmente os dados resultantes deste processo são armazenados em um formato proprietário de um arquivo CAM. No próximo estágio, um pós-processador transformará os dados do arquivo CAM para um programa CNC compatível com o comando CNC, que finalmente passará as informações para a máquina CNC que realizará a manufatura (GONÇALVES, 2013).

### 2.2.2. Operação de desbaste 2 ½ eixos

O número de eixos de uma máquina-ferramenta é definido pelo número de movimentos independentes que podem ser comandados sobre as juntas da máquina (GESSER apud BOHEZ, 2002).

Convencionou-se que os eixos estão dispostos como mostrado na figura 20, a seguir.





**Figura 20: Identificação dos Eixos CNC Pela Regra da Mão Direita (MUNDO CNC,2016)**

Em um sistema tridimensional genérico, os eixos X, Y e Z podem mover-se simultaneamente sendo chamados de sistema de 3 eixos simultâneos. No sistema 2 ½ eixos, as geometrias são criadas com movimentações que geram posicionamentos nos eixos X e Y, sendo então deslocado o eixo Z, que é responsável pela ação de corte em operações como furação, rebaixamento, alargamento, rosqueamento e outras.

Para Souza e Ulbrich (2013), é comum operações de fresamento que iniciam com um bloco prismático de material em estado bruto serem executadas em 2 ½ eixos, pois a ferramenta posiciona-se em X e Y sobre a peça e depois é realizada uma determinada profundidade em Z para remoção de material.

Esta operação de desbaste usada para atividade de fresamento é similar ao que ocorre na lapidação de gemas usando as máquinas semi-automáticas atuais. Assim sendo, a máquina de lapidar e o controle CNC funcionarão semelhantemente a estas operações de 2 ½ eixos e também com semelhanças aos sistemas de prototipagem rápida baseados na produção de camadas sucessivas. Na prototipagem rápida os 2 ½ eixos são explorados de forma diferente. Uma plataforma move-se em Z para definir uma camada e o sistema XY faz o desenho 2D a ser depositado. Na lapidação automática proposta neste trabalho, as facetas das pedras são posicionadas através de dois eixos de rotação, a plataforma move-se em Z para a camada ser lapidada, trata-se, portanto, de um processo semelhante em termos de maquinaria com diferenças no software de controle.

O software de controle da prototipagem rápida chama-se *Slicer*, fatiador em português, e tem como entrada a peça oriunda do sistema CAD e sua saída é código CNC, normalmente padrão ISO 6983 com a trajetória da ferramenta formando camadas de material a ser depositado.

O software pós-processador deste projeto trabalhará com cada faceta individualmente, criando um código CNC a partir de sistemas CAD, mas ao invés de gerar trajetórias para criação de protótipos, irá criar posições individuais das facetas em gemas sendo lapidadas em máquina CNC.

A seguir será abordado mais profundamente o processo de prototipagem rápida. Isto advém da necessidade de geração de mais conhecimentos do ciclo de trabalho da prototipagem que será subsídio para criação dos requisitos necessários para implementação do software pós-processador.

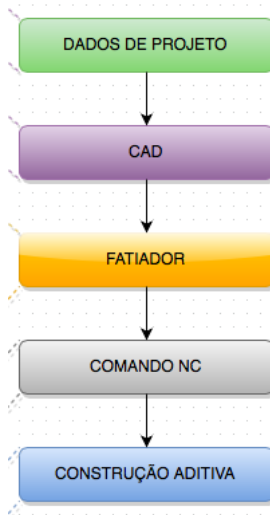
### 2.2.3. Processo de Prototipagem Rápida e a Lapidação CNC

O processo de prototipagem rápida, descrito por Souza e Ulbrich (2013), pode ser dividido em 5 passos básicos, são eles:

1. Criação do modelo;
2. Conversão do modelo para o formato STL, que consiste em malha de triângulos;
3. Fatiamento do modelo em camadas;
4. Construção do modelo físico por adição de camadas, baseado nas camadas fatiadas anteriormente;
5. O acabamento e a limpeza são posteriores à prototipagem, e melhoram a aparência e durabilidade do protótipo.

A figura 21 ilustra os passos em um processo de prototipagem rápida.

O primeiro passo é a coleta e documentação dos requisitos do projeto, enquanto o segundo passo consiste na criação do modelo, que é feita usando um pacote de software CAD. Existe uma quantidade grande destes softwares disponíveis no mercado, como, por exemplo, o SolidWorks®, Catia® ou Pro/ENGINEER®. Continuando o segundo passo, é necessário a conversão para o formato STL, o que já pode ser feito dentro dos pacotes CAD. O SolidWorks®, por exemplo, é capaz de realizar esta conversão ao salvar o arquivo. O STL é um padrão da indústria, portanto os pacotes são obrigados a dar suporte a este formato.



**Figura 21: Processo Básico de RP**  
(Adaptado de SOUZA E ULBRICH, 2013)

Este formato representa uma superfície tridimensional através de uma montagem de triângulos planos. O arquivo possui as coordenadas tridimensionais (x, y e z) dos vértices dos triângulos que compõe o objeto. Por causa da técnica empregada para representar os objetos tridimensionais, o STL sofre algumas limitações. Ele não é capaz de representar superfícies curvas com exatidão. Ao aumentar-se o número de triângulos, fazendo-os menores e em maior número, consegue-se um aumento da aproximação netas curvas, porém um custo adicional relativo a isto é gerado: o tamanho dos arquivos aumenta proporcionalmente, o que torna seu processamento mais lento.

Depois de convertido será realizado o fatiamento do arquivo STL, que é a preparação do modelo para ser produzido. O processo em camadas exige que o modelo seja dividido em fatias de acordo com o tamanho selecionado pelo usuário. Este processo é feito por um software chamado de fatiador, mais conhecido pelo nome em inglês *Slicer*, onde cada fabricante de máquina de RP normalmente fornece seu próprio software para realizar esta tarefa.

O quarto passo é a leitura do código NC que sai do fatiador e conversão em comandos de hardware. Isto é feito por um software de comando CNC. Dentre os comandos CNC pode-se destacar o Linux CNC, um dos mais populares por ser um software livre de código aberto.

O quinto e último passo é a construção camada a camada do protótipo. Esta construção usa uma das técnicas descritas na seção anterior e usa as máquinas RP para contruir uma camada de cada vez usando materiais como papel, polímeros, ou metal em pó.

No final destes passos é feita a limpeza e o acabamento do protótipo. Em alguns casos são feitos polimentos, pinturas, aplicação de resinas especiais, ou até mesmo cura em estufas.

A partir da cadeia de produção CAx apresentada por Gonçalves (2013), mostrada na figura 19, e o processo básico de RP de Souza e Ulbrich (2013), ilustrado na figura 21, foi elaborado um modelo abstrato de manufatura.

Este modelo surgiu para fins de criação de requisitos do software pós-processador e para dividir o processo em unidades autônomas mais fáceis de operacionalizar. Ele é mostrado na figura 22, e traz uma arquitetura em camadas de software independentes (operações funcionais do software). Sendo que cada processo de manufatura poderá implementar as camadas com diferentes ferramentas e diferentes processos. As 5 camadas abstratas da figura são explicadas a seguir:

A camada de requisitos é onde define-se os parâmetros de construção em geral.

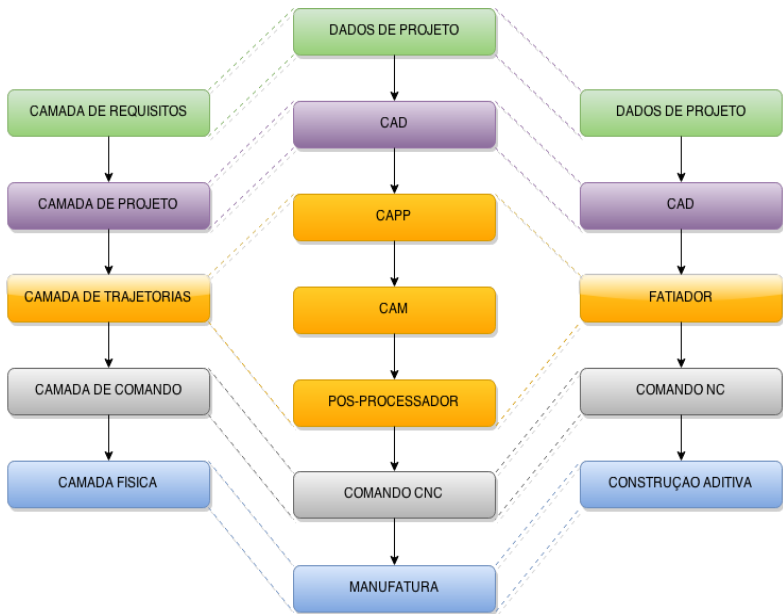
A camada de projeto consiste no desenvolvimento de um modelo computacional do produto. Pode-se usar, por exemplo, equipamentos de digitalização para obter um modelo ou modelamento em ferramentas CAD.

A camada de trajetória é onde se faz os cálculos ou definições de trajetórias baseados no modelo desenvolvido na camada de projeto. Nela, pode-se gerar simulações computadorizadas para evitar colisões ou perda de material.

A camada de comando é feita para haver uma separação entre o software e o hardware de fabricação. Ela normalmente tem como entrada programa CNC e converte para cinemática no hardware executor.

A camada física é o hardware que executa os comandos fornecidos pela camada de comando.

A abstração mostrada na figura 22 é importante no processo de desenvolvimento do software pós-processador. Ela é feita para que se possa construir um software com maior flexibilidade e abrangência de diversas áreas do processo de manufatura, bem como para o entendimento de alto nível da estruturação da manufatura moderna. Outro item importante é que a localização, dentro da cadeia de manufatura, do software deste projeto.



Modelo Abstrato de Manufatura

Cadeia de Produção CAX Comum

Processo Básico de RP

## Figura 22: Modelo de Processo Fabricação Convencional x RP (Elaborado pelo Autor)

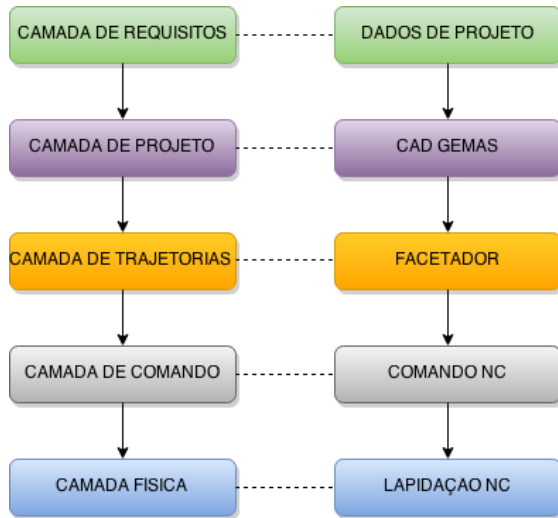
Notou-se também que ele é mais abrangente do que somente um pós-processador na cadeia de produção CAX. O software pós-processador para lapidação está atuando na camada definida como camada de trajetórias na figura 5, mas desempenha mais tarefas do que aquelas definidas para esta camada. O software seria mais próximo aos fatiadores definido no processo básico de RP. Isto pelas suas capacidades de interpretar arquivos CAD, bem como, pelos cálculos de trajetória baseados nestes arquivos. Porém, as semelhanças param por aí, pois os processos envolvidos são diferentes, os fatiadores, como visto, trabalham criando camadas, e o pós-processador vai criar facetas, porém ambos são baseados em arquivos CAD.

### 2.2.4. Processo de Manufatura Rápida para Lapidação de Gemas

A conclusão do estudo destes processos de manufatura

computadorizado, conhecido como manufatura rápida, é compor uma solução baseada nos processos existentes de manufatura baseada em máquinas flexíveis de comando numérico. Criar a partir disto uma solução para lapidação que tenha como modelo as tecnologias envolvidas no processo da prototipagem rápida.

O processo resultante foi chamado Processo de Lapidação CNC, ele está ilustrado na figura 23. A união deste conhecimento irá proporcionar requisitos para o software pós-processador, sendo o produto uma união entre os sistemas CAM da área de usinagem e os softwares fatiadores, conhecidos como *Slicers* da área de prototipagem rápida, originando um novo conceito em software pós-processador para a área de lapidação que poderia ser chamado de Facetador CNC.



**Figura 23: Processo Proposto para Lapidação CNC (Elaborado pelo Autor)**

### 2.3. CAD/CAM

Inicialmente, os comandos numéricos não dispunham de ferramentas que facilitassem no processo de usinagem. Os programas CNC eram feitos manualmente, porém esta abordagem tinha problemas inerentes, como, por exemplo, de programas muito grandes e difíceis de manter devido à complexidade das peças. Para auxiliar neste processo, foram desenvolvidas aplicações CAD (*Computer Aided Design*) e CAM (*Computer Aided Manufacturing*). Estas ferramentas permitiram

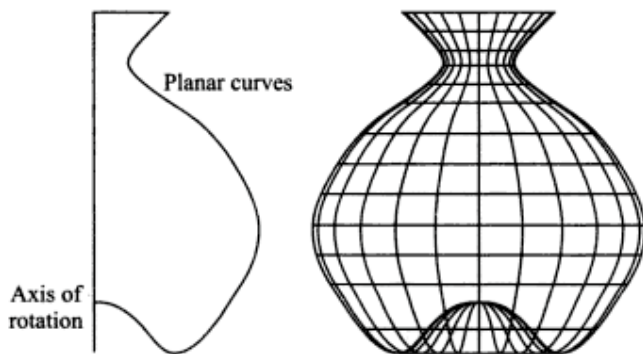
programar peças simples ou complexas mais rapidamente e com maior precisão (SOUZA E ULBRICH,2013).

Lima (2004) relata que, apesar de existirem sistemas CAD/CAM integrados em um só aplicativo, é comum se encontrar o ambiente CAD separado de CAM, embora seja necessária a integração entre os aplicativos. Neste trabalho dá-se importância significativa à computação gráfica e o seu uso na manufatura através da cadeia CAx, portanto CAD e CAM serão separados e discutidos individualmente com maior profundidade nas próximas seções.

### 2.3.1. CAD

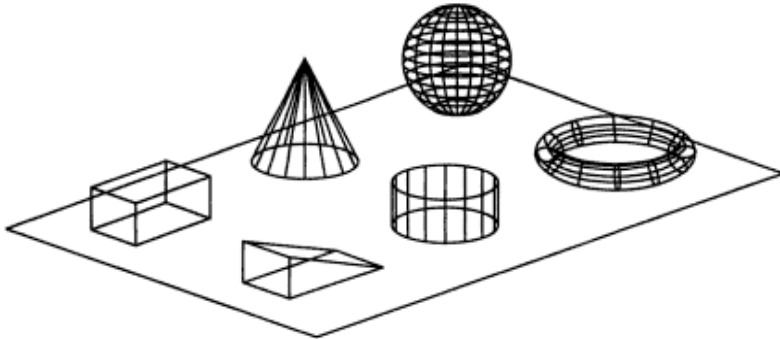
Sistemas CAD são aplicativos de computador que usam computação gráfica para virtualizar modelos do mundo real. Neles um usuário poderá criar modelos tridimensionais seguindo requisitos de construção que possibilitarão a visualização e mudanças no modelo, o que permitirá a validação destes modelos sem que eles existam na realidade, somente no computador. Estes modelos podem ser em duas ou três dimensões (2D ou 3D), dependendo do sistema CAD em uso. Sendo que, quando a modelagem é feita em 3D, ela é chamada modelagem virtual sendo ela representada por coordenadas cartesianas no espaço (x, y e z).

Para Souza e Ulbrich (2013), na atualidade são desenvolvidos sistemas CAD para aplicação em diversas áreas, tais como arquitetura, engenharia civil, engenharia mecânica, engenharia elétrica, *design* e desenvolvimento de produtos. Os sistemas CAD podem ser divididos em três tipos: modeladores de superfícies, de sólidos e híbridos. As figuras 24 e 25 a seguir mostram exemplos de modelagens em sistemas CAD.



**Figura 24: CAD de Modelagem Superficial (NARAYAN et. al., 2008)**

Os sistemas CAD com técnica de modelagem de superfície não possuem espessura, e sua geometria é representada como uma casca. A técnica trabalha as relações superficiais dos modelos e maior precisão na definição de curvas. Com isto, pode-se concluir que a modelagem de superfície é empregada na modelagem de formas geométricas complexas,



**Figura 25: CAD de Modelagem Sólida**  
(NARAYAN et. al., 2008)

ou seja, para modelar objetos que possuem elevado número de curvas e garante a continuidade de tangência e curvatura nos pontos de um objeto modelado.

Nos sistemas modeladores de sólidos os objetos representam um volume fechado e possuem informações volumétricas, superficiais e físicas. Este sistema tem dificuldade para lidar com formas complexas com curvaturas. Porém, é adequado para engenharia e mais simples de operar com relação aos de modelagem por superfície.

Os sistemas híbridos são os mais modernos e mais flexíveis, também são os mais caros devido à complexidade, sendo empregados em plataformas chamadas de grande porte.

### 2.3.2. CAM

O CAM (*Computer Aided Manufacturing*) é uma ferramenta computacional para, a partir de uma geometria definida em sistemas CAD e de dados sobre a máquina e comando NC, executar a geração de programas CNC. Estes programas poderão ser usados na usinagem de peças por diversos processos. Dentre eles pode-se citar o fresamento, o torneamento, a eletroerosão, o corte em chapas e outros.

Conforme descrito por Carvalho (2013), o desenvolvimento de sistemas CAM data do início de 1955 e começou com o desenvolvimento de uma linguagem que seria independente de comando CNC, a linguagem



APT (*Automatic Programming Tool*), que tinha como objetivo a elaboração de programas de comando numérico. Esta linguagem teria códigos específicos para definição de formas geométricas e de parâmetros associados à usinagem, o que resulta em uma especificação de percursos de máquina em um nível intermediário entre o software CAD e o controle da máquina CNC. Em resumo, esta linguagem forneceu uma camada de abstração, um nível mais genérico, que possibilitaria a descrição do processo de usinagem em alto nível. A linguagem APT necessita de um pós-processador para gerar o programa CNC de acordo com as características individuais de cada máquina e seu respectivo software controlador.

A estrutura descrita, que apresenta um processador genérico e um específico, ainda é usado na implementação de sistemas CAM modernos, com a vantagem que atualmente dispõe-se de interfaces gráficas muito mais amigáveis que agilizam o trabalho do operador do sistema.

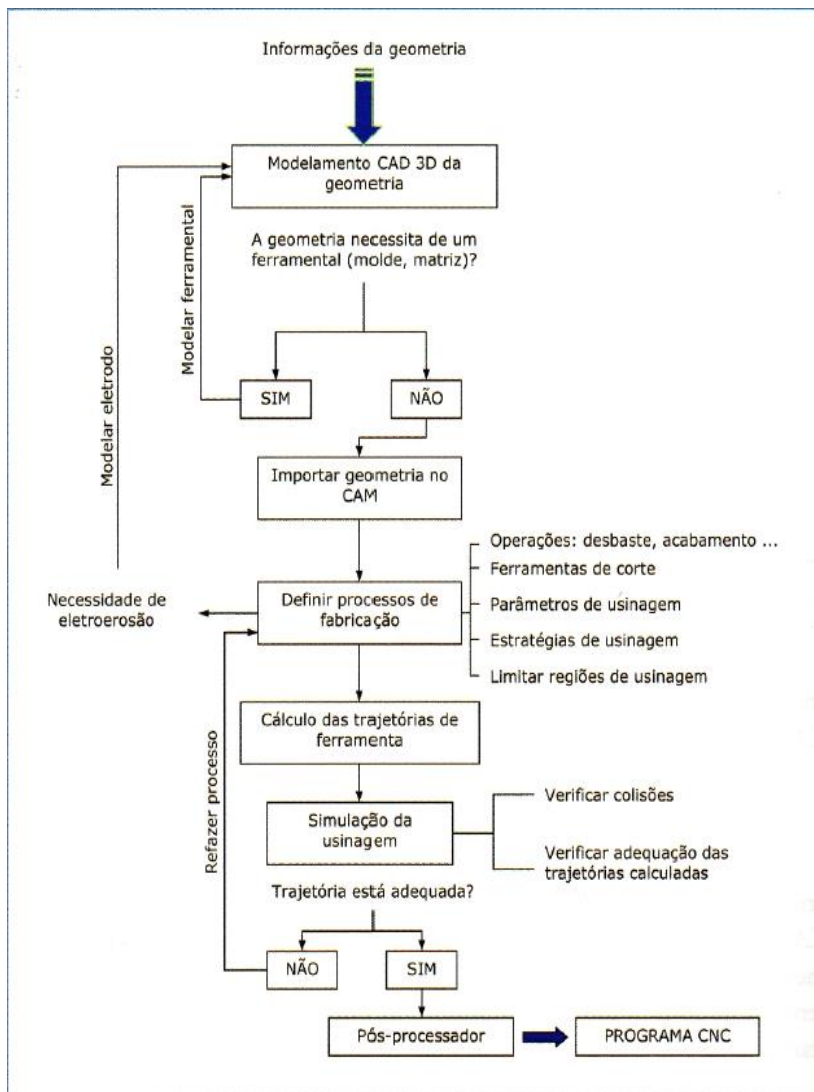
As principais funções de um sistema CAM, de acordo com Souza e Ulbrich (2013) são:

- a) Calcular as trajetórias da ferramenta para usinagem;
- b) Simular a usinagem em computador através de modelos; e
- c) Gerar programas CNC para máquinas específicas.

O procedimento de trabalho de sistemas CAM para realizar as funções descritas é mostrado na figura 26.

O foco principal deste trabalho, os pós-processadores, está incluído nos processos que compõe o sistema CAM. Sendo que o pós-processador desenvolvido, além do processo convencional em pós-processadores de conversão de trajetórias para código CNC, deverá também conter mais processos de sistemas CAM.

Podemos citar os seguintes processos a serem implementados no software: importação da geometria, uma descrição das trajetórias da ferramenta (ao invés de cálculo, como descrito por Souza e Ulbrich (2013)), a entrada de dados relativos aos processos de fabricação e dados que eventualmente não puderam ser obtidos da geometria a ser usinada.



**Figura 26: Metodologia geral de trabalho de sistemas CAM (SOUZA E ULBRICH, 2013)**

### 2.3.3. Pós-Processadores

Para Adivarekar e Liou (2012), os pós-processadores são interfaces entre os sistemas CAM e as máquinas CNC. Para executar suas funções ele precisa converter dados de corte (*CL Data*) da saída de sistemas CAM para programas CNC. Ele é um tradutor que lê, interpreta dados de manufatura fornecidos pelo sistema CAM, e converte estes dados para um código CNC determinado, dependendo da configuração de máquina e controlador CNC.

Uma definição mais ampla foi feita por Suh et. al. (2008), que define da seguinte maneira: "Atualmente para maquinaria com G-Code, um pós-processador é requerido para converter o desenho do espaço de produto para o espaço de maquinaria".

Os elementos chave para desenvolvimento de um pós-processador para usinagem CNC são:

1. Processamento baseado em dados de localização de corte (*Cutting Location Data*, ou *CL Data*);
2. Modelo cinemático da máquina CNC; e
3. Compensação de erros.

No princípio, os softwares pós-processadores eram separados do sistema CAM, sendo, portanto, independentes. No entanto, nos lançamentos mais modernos, os pós-processadores vêm integrados à plataforma CAM. Nestas plataformas existem softwares próprios integrados para criação de pós-processadores que facilitam a criação (SOUZA E ULBRICH, 2013).

### 2.3.4. Desenvolvimento de Aplicações CAD/CAM

Para o desenvolvimento de aplicações de computação gráfica, como CAD e CAM, existem diversas API (*Application Programming Interfaces*) disponíveis. Entre elas, destacam-se as seguintes: OpenGL e Direct3D. Estas bibliotecas funcionam mais frequentemente em linguagens de mais baixo nível como C ou C++. Sendo assim, elas fornecem as fundações para que se desenvolvam APIs gráficas em linguagens de alto nível, como por exemplo: Java, C#, BASIC, Delphi, etc.

A biblioteca gráfica OpenGL é uma API com padrão aberto criada pela empresa *Silicon Graphics*® que fornece um grande número de funções para renderização de gráficos em 2D e/ou 3D, sendo que oferece suporte a várias plataformas de sistema operacional. Incluindo o Microsoft Windows e o Linux.

Por outro lado, a biblioteca Direct3D, um concorrente do

OpenGL, é uma API proprietária da empresa Microsoft que possui funcionalidades semelhantes ao OpenGL, entretanto só pode ser executada em ambiente Windows.

No escopo deste trabalho descarta-se o uso e o estudo aprofundado da API Direct3D, pois ela é menos flexível com relação às plataformas de sistemas operacionais e direciona para o Windows. Nesta plataforma os custos de licença e a falta de transparência em relação ao código fonte do sistema operacional inviabilizam qualquer avanço futuro de projeto.

De acordo com Chen e Wegman (2006), OpenGL é a biblioteca gráfica mais amplamente usada, sendo suportada nas plataformas *desktop* e *workstation* mais populares. Na atualidade a OpenGL está também disponível em dispositivos móveis possuindo um conjunto de comandos gráficos ou funções que são embutidas em linguagens como C, C++, Java entre outras.

Java foi citada anteriormente como linguagem de alto nível que suporta bibliotecas gráficas, especificamente o OpenGL. Sendo uma linguagem de alto nível, ela possui APIs que possuem maior produtividade devido à agregação de valor que é dado pelo encapsulamento das camadas de software inferiores, de baixo nível. Este encapsulamento das operações de baixo nível cria uma nova interface mais prática e fácil de utilizar. A linguagem Java se torna assim, devido às suas características de execução independente de plataforma e ao suporte à biblioteca OpenGL, apropriada para o desenvolvimento do software pós-processador deste trabalho.

As APIs de computação gráfica da plataforma Java que fornecem interfaces 2D e 3D são: JOGL, Java3D e JavaFX 3D. JOGL é uma implementação Java que vincula a linguagem com a biblioteca OpenGL. Ela fornece gráficos 3D suportados por hardware (placas aceleradoras de vídeo normalmente suportam OpenGL para dar maior velocidade de processamento gráfico) em aplicações escritas em Java e permitem a integração de OpenGL com APIs gráficas Java AWT e Swing (CHEN e WENGMAN, 2006).

Java3D é uma API desenvolvida pela empresa *Sun Microsystems* para renderizar gráficos interativos 3D usando a linguagem de programação Java. Esta API usa OpenGL ou Direct3D para realizar a renderização, enquanto a descrição de cenas 3D, lógica de aplicação e interatividade é feita em alto nível com código Java (SELMAN, 2002).

A API JavaFX 3D é a mais atual dentre as dicitadas. Ela fornece uma biblioteca gráfica tridimensional para a plataforma JavaFX. Segundo Yoon (2013), os casos de uso em que JavaFX 3D é aplicável são os

seguintes:

- Visualização de inventário e processos;
- Visualização científica e de engenharia;
- Gráficos comerciais em 3D;
- CAD e CAE;
- Processamento de imagens na área da medicina.

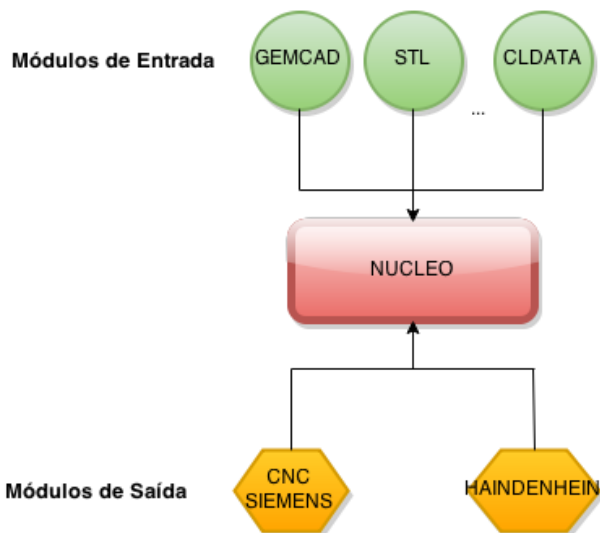
De acordo com Chen e Wengman (2006), a biblioteca JOGL trabalha em mais baixo nível do que Java3D. Ela é apenas um encapsulador (*wrapper*) para a biblioteca OpenGL através de Java Native Interface (JNI). Ela também é de mais baixo nível que JavaFX 3D. Portanto, o uso de JOGL neste projeto, devido a requisitos de tempo de desenvolvimento, teria seu uso descartado.

JavaFX 3D e Java3D possuem o mesmo nível de API e os mesmos conceitos. Ao invés de funções OpenGL de baixo nível, elas trabalham com construções gráficas de alto nível chamadas *scene graphs* para criação e manipulação de objetos dentro de um cenário 3D. Ambas também são adequadas para os casos de uso em engenharia, entretanto a documentação do Java3D é mais vasta, devido ao fato de que o JavaFX 3D é uma biblioteca relativamente nova. Isto torna o Java3D o melhor candidato para desenvolver eventuais requisitos e casos de uso que necessitem visualização 3D no software pós-processador desenvolvido neste trabalho.

### 2.3.5. Formatos de Arquivos CAD e Código CNC

Nesta seção são apresentados alguns formatos de arquivos relevantes para o software pós-processador, sendo que ele deverá ter uma arquitetura capaz de suportar diferentes formatos de arquivos de entrada. Estes arquivos são oriundos de diferentes softwares CAD. Além disso, o software deverá suportar a geração, em sua saída, de código NC para diferentes comandos.

Sendo assim, o sistema a ser desenvolvido pode ser dividido em três módulos: módulo de extensões de entrada, núcleo do sistema, e módulo de extensões de saída, como ilustrado na figura 27. São descritos a seguir os formatos de arquivo com maior potencial para serem implementados na fase de implementação do projeto como módulos de entrada ou de saída e instalados junto ao núcleo da aplicação.

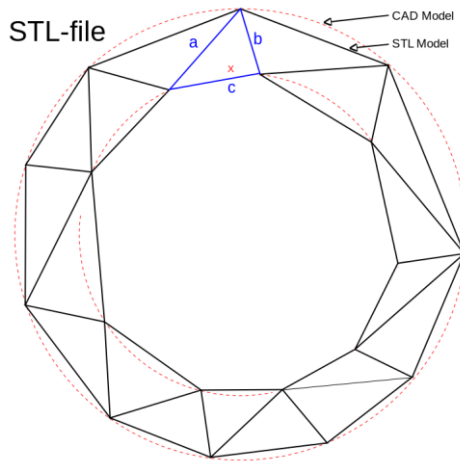


**Figura 27: Módulos do Pós-Processador (Elaborado pelo Autor)**

### 2.3.5.1. Formato STL

Desenho auxiliado por computador (CAD) e máquinas de RP têm se tornado muito importante na indústria e em projetos acadêmicos. Uma disjunção chave existe entre as renderizações de modelos em três dimensões e as capacidades das máquinas de RP. É preciso quebrar os modelos em uma forma mais simples para que possam ser executados em máquinas que executarão a manufatura, criando assim os protótipos físicos. O padrão de *facto* que soluciona esta aplicação chama-se StereoLithography ou, simplesmente, STL (CHAMBERLAIN et. al., 2001).

Ainda de acordo com Chamberlain et. al. (2001), o formato de arquivo STL foi criado pela empresa 3D Systems para como parte do primeiro projeto comercial da tecnologia de prototipagem rápida de stereolitografia datado do ano de 1987. Este formato representa as superfícies dos modelos CAD como triângulos, os quais funcionam muito bem para superfícies retangulares, mas também podem ser usados para superfícies curvas. Nas superfícies curvas, aumenta-se o número de triângulos necessários para representá-las, e no final irá haver somente uma aproximação da superfície curva. Esta aproximação, mostrada na figura 28, deve ser analisada para ver se atende os requisitos de uma dada aplicação, para então validar o uso desta tecnologia.



**Figura 28: Exemplo de STL versus formato CAD (STL, 2015).**

Pacotes CAD incluem opções para conversão do modelo em formato STL. Este processo é chamado de tesselação (tradução livre de *tessellation*). A saída da tesselação é um arquivo STL (*Standard Tessellation Language*), que pode estar nos formatos ASCII ou binário. Em geral, arquivos binários são de 1/4 a 1/6 do tamanho de arquivos ASCII (STL, 2015).

O formato ASCII pode ser lido em editores de texto convencionais fornecidos nos sistemas operacionais. Entretanto, o formato binário é mais compacto, porém não pode ser lido facilmente pelos usuários.

Para trazer para o escopo deste trabalho, pode-se dizer que o modelo STL é uma interface entre a camada de projeto e a camada de trajetórias. Exatamente nesta camada está o software pós-processador, fazendo com que o seu estudo seja interessante.

A estrutura de um arquivo STL ASCII inicia com a linha de cabeçalho definida a seguir:

```
solid name
```

onde *name* é um a cadeia de caracteres opcional usada para nomear o modelo. Após o cabeçalho o arquivo continua descrevendo os triângulos que compõe o modelo, cada um representado da seguinte maneira:

```
facet normal ni nj nk
  outer loop
```

```

        vertex v1x, v1y, v1z
        vertex v2x, v2y, v2z
        vertex v3x, v3y, v3z
    endloop
endfacet

```

Cada  $n$  ou  $v$  representado é um número de ponto flutuante com um coeficiente ou mantissa e um expoente indicado pela letra "e". Por exemplo: 1.005000e-001.

Os números representados por  $n$  indicam a orientação da normal para cada triângulo. Isto significa para qual ponto, qual coordenada, o triângulo individual está apontando. Estes dados dão idéia de como representar objetos tridimensionais em plataformas bidimensionais como monitores de computador. Por exemplo, um triângulo de frente para o usuário do sistema deverá estar sendo mostrado na sua integralidade, pois está de frente, já aqueles adjacentes começam a ter mudanças que podem ser implementadas como efeitos de sombra.

Os números representados por  $v$  são os vértices tridimensionais, coordenadas  $x$ ,  $y$ , e  $z$ , que compõe o triângulo.

O arquivo finaliza com a seguinte sentença:

```
endsolid name
```

Onde *name*, assim como no cabeçalho é o nome do modelo.

Arquivos STL binários tem um cabeçalho de 80 caracteres, acrescido de um número inteiro sem sinal de 4 bytes, que indica o número de triângulos no arquivo. O arquivo finaliza ao chegar-se ao último triângulo informado. Cada triângulo é descrito por 12 números de ponto flutuante de 32 bits, sendo três para a normal e mais três para cada coordenada  $x$ ,  $y$ , e  $z$ . Estes números de ponto flutuante estão no formato IEEE 754. Depois de descritos os pontos de vértice de cada triângulo encontra-se um número inteiro sem sinal de 2 bytes chamado "atributo contador de bytes", o qual pode ser zero por que a maioria dos softwares não o suporta (STL, 2015). A seguir mostra-se o *layout* deste arquivo:

```

uint8[80] - Header
uint32 - Number of triangles

foreach triangle
    real32[3] - Normal vector
    real32[3] - Vertex 1
    real32[3] - Vertex 2
    real32[3] - Vertex 3

```



end

### 2.3.5.2. Formato GemCAD®

O GemCad® é um programa aplicativo para computadores desenvolvido para a área de desenho auxiliado por computador que é executado no sistema operacional Windows. Ele simula máquinas de lapidação chamadas *Faceting Head*, mostrando visões tridimensionais da gema sendo trabalhada. O trabalho final resultante do software é um diagrama de facetamento que mostra muitas visões da pedra com ângulos, índices e dimensões (STRICKLAND, 2002).

Há dois formatos nativos e mais quatro de exportação suportados pelo GemCad®. Os formatos nativos são ASC e GEM, enquanto os formatos de exportação são DXF da Autodesk, VRM (*Virtual Reality Markup Language*), B3D, e formato THB.

O formato GEM tem sua estrutura fechada para os usuários e eventuais concorrentes. Por outro lado, o formato ASC é um formato aberto e tem sua estrutura como mostrado na figura 29.

```
GemCad 5.0
g 96 0.0
y 8 y
I 1.54
H Standard Round Brilliant
H 9/5/02
a -90.000000 1.02653281 93 n G 87 81 75 69 63 57 51 45 39 33 27 21 15 9 3
G 16-sided outline
a -42.500000 0.61819401 93 n 1 87 81 75 69 63 57 51 45 39 33 27 21 15 9 3
a -41.500000 0.61701256 96 n 2 84 72 60 48 36 24 12
a 34.000000 0.68444470 3 n A 9 15 21 27 33 39 45 51 57 63 69 75 81 87 93
a 28.000000 0.60896430 96 n B 12 24 36 48 60 72 84
a 16.000000 0.50613241 6 n C 18 30 42 54 66 78 90
a 0.000000 0.36450932 96 n T
F GemCad for Windows Manual
```

**Figura 29: Arquivo .ASC gerado pelo GemCad® (STRICKLAND, 2002).**

De acordo com Strickland (2002), a estrutura de dados mostrada no arquivo possui as seguintes informações:

- A primeira linha é sempre o cabeçalho, e vai conter a seguinte informação "GemCad® 5.0";
- O "g" informa a quantidade de índices na engrenagem da máquina de lapidação;
- O "y" dá a simetria, uma informação interna do GemCad®;
- O "I" revela o índice de refração da pedra a ser lapidada;

- Linhas com "H" representam cabeçalho de diagramas de lapidação;
- As linhas com "a" representam os dados de facetas. Há uma linha para cada camada de facetas, que representam diferentes ângulo no cabeçote de lapidação. Depois do "a" tem-se o número que representa o ângulo de facetamento. A seguir, a distância do centro da peça até a faceta, finalizando com os números de índices de lapidação. Os eventuais nomes de facetas feitos no GemCad® aparecem depois do caractere "n";
- Instruções de facetamento podem aparecer seguidas do caractere "G"; e
- Instruções de rodapé são colocadas seguidas do caractere "F" (*footnote*).

### 2.3.5.3. ISO 6983

Um dos problemas mais sentidos no início do desenvolvimento de máquinas comandadas numericamente era a padronização da linguagem usada nos programas com instruções de fabricação. Cada fornecedor de máquina criava seu próprio padrão e não havia intercambialidade. Então a ISO (*International Organization for Standartization*), a partir de 1975, padronizou os códigos de máquinas CNC na norma ISO1056 (2010), que foi atualizada em 1982 para ISO 6983-1. (ISO 6983)

Os programas de fabricação são conhecidos pelos prefixos dos comandos usados na linguagem. Estes prefixos são códigos "G" ou "M" que originam um nome informal para a linguagem: "Código G". Os comandos cujo prefixo é "G" são usados para declarar funções preparatórias, enquanto os prefixos "M" são usados em funções designadas miscelâneas. Os dois comandos são compostos por 3 dígitos, no seguinte padrão: GXX ou MXX. Onde XX é o número da função programada (MUNDO CNC,2008). Alguns comandos de miscelânea, "M" são ilustrados no anexo 1, e exemplos de comandos "G" são mostrados no anexo 2.

Mesmo depois de estabelecida a padronização através da ISO e os fornecedores usarem algo do padrão, há particularidades em alguns detalhes implementados por eles, o que revela ainda uma necessidade adicional de padronização. Esta necessidade vem sendo superada com o uso de softwares pós-processadores (GONÇALVES, 2013).

# Capítulo 3

## METODOLOGIA PARA DESENVOLVIMENTO

---

No projeto de software há uma série de modelos organização de processos, chamados também de metodologias de desenvolvimento, para todo o ciclo de vida dos projetos. Estas metodologias fazem uma divisão do trabalho de desenvolvimento de software em fases (SOMMERVILLE, 2011).

Cada uma destas fases possui atividades específicas com o intento de melhor planejamento e gerenciamento. Há quatro atividades básicas que todas as metodologias possuem, mas as implementam de diferentes maneiras, a saber: Especificação, Desenvolvimento, Validação e Evolução.

Estas metodologias podem ser divididas em duas especialidades, aquelas que são orientadas ao planejamento e as chamadas metodologias ágeis.

As metodologias orientadas ao planejamento (*Plan-Driven*), direcionam mais tempo ao projeto ou planejamento, enquanto que as ágeis são mais orientadas ao desenvolvimento do software com o intuito de entregar rapidamente os artefatos devido às mudanças muito frequentes de requisitos.

O software pós-processador terá requisitos estáveis e irá ter foco em um projeto melhor para permitir que suas interfaces de integração com outros sistemas, na entrada sistemas CAD e na saída comandos CNC. Estas interfaces devem ser planejadas para maximizar o número de possibilidade de integração e minimizar eventuais manutenções corretivas do software. Desta forma, foi selecionado para o projeto a metodologia do Processo Unificado (UP), que além de ser orientada ao planejamento, permite mudanças nos requisitos desde que possíveis e planejadas dentro dos limites de tempo e custo do projeto.

### 3.1. Metodologia UP

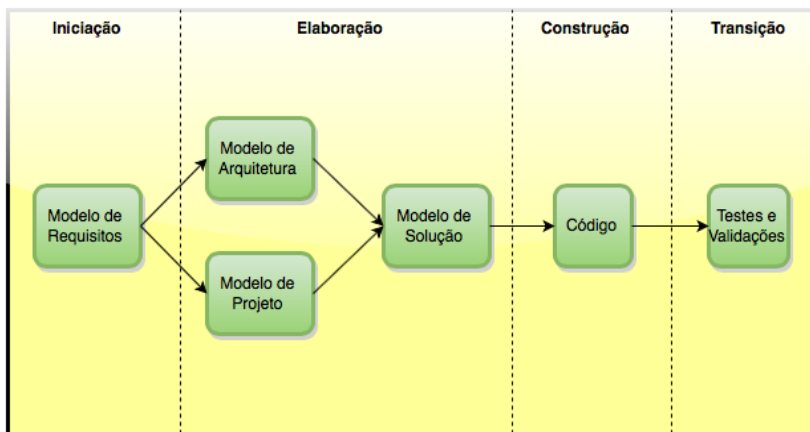
A metodologia UP permite alterações nos requisitos e prevê revisões ao fim de cada iteração. Estas iterações podem ser feitas em

qualquer etapa do processo. Assim, elas podem levar apenas alguns dias, podendo então o novo requisito ser atendido com uma certa rapidez, diferentemente de outras metodologias, como por exemplo a metodologia Cascata, que não permite alterações de requisitos até que se conclua todo um ciclo de desenvolvimento de software.

Quatro fases organizam a metodologia UP, de acordo com o guia OO-226 da Sun Microsystems (SUN, 2003), que são:

- Iniciação - Cria uma visão do software. Esta fase foca no entendimento do problema para o sistema proposto. Também uma arquitetura preliminar é proposta, riscos importantes do projeto são identificados, e a fase de elaboração é planejada em detalhes. A revisão bibliográfica ajudará a fornecer subsídios para esta fase, sendo que estudos realizados nela poderão trazer novos requisitos e detalhamento de processos importantes para a aplicação.
- Elaboração - É definida a maioria dos casos de uso e também a arquitetura que vai suportar o sistema. O propósito desta fase é reduzir o risco do projeto criando uma *baseline* sobre a qual todo o resto do software será construído. No final desta fase, o gerente de projetos terá informações suficientes para planejar as fases de construção e de transição.
- Construção - É onde o software é construído. Podendo haver muitas iterações, o software é desenvolvido incrementalmente pela adição de novos casos de uso à *baseline* de arquitetura. No final desta fase o sistema está pronto para uma versão beta.
- Transição - O software sai da versão beta para a produção. Esta fase prepara o sistema para o lançamento para a produção. Atividades relacionadas com esta fase incluem testes de aceitação, depuração, treinamento de usuários, e construção de ambiente de produção.

A figura 30 mostra a metodologia proposta para o desenvolvimento do software pós-processador:



**Figura 30: Metodologia Proposta do Projeto Pós-Processador (Adaptado de SUN, 2003)**

De acordo com a figura 30, o projeto inicia com o Modelo de Requisitos, que é originado de modelos mentais dos clientes a partir da aplicação, gerados a partir da coleta e análise dos requisitos. A coleta dos requisitos será mostrada em detalhes no próximo capítulo. Os requisitos não-funcionais são transformados no Modelo de Arquitetura, que definirá a estrutura de alto-nível da solução de software. Os requisitos funcionais do Modelo de Requisitos são transformados em um Modelo de Projeto, que define componentes abstratos do software. E então os modelos de Projeto e de Arquitetura são juntados para produzir o Modelo de Solução, que define a estrutura detalhada da solução de software. O Modelo de Solução é usado para guiar a construção do código durante a fase de Construção. Os testes e validações serão feitos na fase de Transição. Nela será verificado se o software não possui defeitos, bem como será efetuado a validação do software pós-processador com o auxílio de virtualização de uma máquina CNC de lapidação.

# Capítulo 4

## PROJETO DO SOFTWARE

---

O projeto do software, conforme a metodologia seguida no projeto, será composto de quatro modelos que darão uma visão em alto nível do software que será desenvolvido. Estes modelos são: modelo de requisitos, modelo de projeto, modelo arquitetural e modelo de solução. A seguir cada um destes modelos será descrito e implementado.

### 4.1. Modelo de Requisitos

O modelo de requisitos está situado na fase de iniciação da metodologia. É o primeiro modelo a ser preparado. Isto porque ele fornecerá uma interface entre as necessidades do usuário e a equipe de desenvolvimento do projeto, sendo fundamental para que se comece a desenvolver o software. Para atender a estas necessidades, o modelo será composto dos seguintes elementos:

- Especificação dos requisitos de software (*Software Requirement Specification - SRS*);
- Diagrama de Casos de Uso;
- Diagrama de Classes de Domínio.

Esta fase é uma fase de análise de sistema, também conhecida como fase de especificação de software ou de engenharia de requisitos. Para Sommerville (2013): "É o processo de entender e determinar que serviços são requeridos do sistema e identificar suas limitações na operação e desenvolvimento".

Em última análise, pode-se notar que elementos descritos, necessários no modelo de requisitos, tem a finalidade de elucidar quais processos serão necessários para automação e, não menos importante, qual estrutura de dados dará suporte aos processos. Simplificando, elucidam os processos e dados necessários para automatizar um dado negócio. Isto é concluído devido ao fato de que os dois primeiros elementos, SRS e Diagrama de Casos de Uso, descreverão os processos do software pós-processador, enquanto o diagrama de Classes de Domínio irá desvendar a estrutura de dados necessária para dar suporte para que os processos realizem suas funções.

#### 4.1.1. Especificação dos requisitos de software

Em sistemas comerciais, normalmente os requisitos de software são elaborados a partir de entrevistas realizadas com os *stakeholders*, para que estes informem as necessidades e como funcionam seus processos visando que eles sejam automatizados. Este trabalho seguiu esta linha e realizou a especificação baseada em entrevistas com lapidários. Porém, além disto, adicionou-se elementos do estudo realizado na revisão bibliográfica.

Os perfis profissionais dos *stakeholders* que auxiliaram na elucidação dos requisitos e desenvolvimento do sistema são os seguintes: lapidário, projetista mecânico e engenheiro de software.

O lapidário irá auxiliar nas informações sobre o processo de lapidação, o projetista mecânico vai ajudar passando informações sobre a cinemática e a geometria da máquina de lapidar, enquanto o engenheiro de software irá desenvolver a aplicação propriamente dita, mas baseado nos levantamentos de informações e algumas produções feitas pelos outros *stakeholders*.

Projetos de software em fase inicial de elaboração, assim como projetos de qualquer área, devem estabelecer os requisitos e priorizá-los para o posterior desenvolvimento. Estes requisitos de sistema de software são classificados frequentemente como requisitos funcionais e não-funcionais. Requisitos funcionais são declarações das funções que o sistema deverá realizar. Em alguns casos, os requisitos funcionais podem também dizer o que o sistema não deverá fazer. Os requisitos não-funcionais são limitações nos serviços ou funções oferecidas pelo sistema (SOMMERVILLE, 2011).

Os requisitos funcionais iniciais de desenvolvimento do pós-processador são listados e priorizados na tabela 3. As prioridades definidas foram "Obrigatório" ou "Desejado". "Obrigatório" é usado para definir requisitos essenciais para o funcionamento. E os categorizados como "Desejado" são aqueles que não existe uma obrigatoriedade no desenvolvimento, entretanto podem causar um impacto positivo caso sejam implementados.

**Tabela 3: Requisitos Funcionais do Software Pós-Processador  
(Elaborado pelo autor)**

Nº	Função	Cat.	Atributos, Detalhes e Restrições
1	O sistema deverá gerar programas CNC a partir de arquivos de descrição de geometria de gemas lapidadas para automatizar a lapidação e polimento das gemas.	Obrigatório	Este programa automatizará a parte mais trabalhosa do processo, nomeadamente a lapidação e o polimento.
2	O sistema deverá ter uma linguagem de descrição do processo de fabricação.	Obrigatório	Não dedicada a um comando CNC. Buscar algo existente baseado em XML.
3	O sistema deverá possibilitar a inserção de novas extensões de comandos CNC e seus respectivos padrões de comandos	Obrigatório	Usuário faz a inserção em tempo de execução. Por exemplo, comandos baseados em ISO 6983.
4	O sistema deverá mapear linguagem de descrição do processo de fabricação para comando CNC	Obrigatório	Após a inserção de novos comandos deve ser feito o mapeamento.
5	O sistema deverá possibilitar a inserção de novas extensões de formatos de descrições de geometria (CAD).	Obrigatório	Usuário faz a inserção em tempo de execução.
6	O sistema deverá possibilitar o mapeamento de novas extensões de formatos de descrições de geometria com a linguagem de descrição do processo de fabricação para comando CNC	Obrigatório	Após a inserção de novos formatos de geometria deve ser feito o mapeamento.



**Tabela 3: Requisitos Funcionais do Software Pós-Processador  
(Elaborado pelo autor) (continuação)**

Nº	Função	Cat.	Atributos, Detalhes e Restrições
7	O sistema deverá mostrar a geometria da lapidação em 3D	Desejado	
8	O sistema deverá possibilitar a movimentação da geometria de lapidação na visualização em 3D	Desejado	
9	O sistema deverá permitir a criação de programação do processo CNC através de componentes visuais (ícones).	Desejado	O programador não necessitaria conhecer comandos e sim componentes de diagrama que representa os comandos.
10	O sistema deverá possuir um ambiente de virtualização capaz de simular a máquina de lapidação e sua cinemática enquanto executa programas CNC gerados.	Desejado	O software poderá mostrar as trajetórias sem a necessidade de uso de outros softwares de máquina virtual.

A tabela 4 mostra a lista de requisitos não-funcionais do pós-processador.

**Tabela 4: Requisitos Não-Funcionais do Software Pós-Processador  
(Elaborado pelo autor)**

Nº	Requisito	Cat.	Atributo	Detalhes e Restrições
1	O sistema deverá funcionar em vários sistemas operacionais	Desejado		Linux, iOS ou Windows.
2	O sistema deverá suportar diversos formatos de arquivos de saída.	Desejado	Ilimitado número de comandos CNC.	Software deverá possuir uma interface de <i>plug-ins</i> de entrada.
3	O sistema deverá permitir vários formatos de arquivos de geometria de entrada (CAD).	Desejado		Software deverá possuir uma interface de <i>plug-ins</i> de saída.

No apêndice 1, elaborado pelo autor, é mostrado um diagrama com os requisitos do software pós-processador. Este diagrama usa uma associação, componente da linguagem UML (*Unified Modelling Language*), chamada agregação, para mostrar um relacionamento do tipo "todo-parte" entre os requisitos. Foi criada uma hierarquia com o requisito mais global sendo a raiz do diagrama, sendo este o nível mais alto e abstrato, e contendo os requisitos com maior detalhamento em níveis inferiores da hierarquia.

Nas seções a seguir será apresentada a sequência do trabalho do modelo de requisitos. Serão desenvolvidos o diagrama de casos de uso e as classes de domínio, tudo a partir dos requisitos que foram apresentados.

#### 4.1.2. Diagrama de Casos de Uso

Um caso de uso descreve um processo. Um processo, por sua vez, descreve uma sequência de eventos, ações e transações requeridas para produzir ou completar algo de valor para uma organização ou ator (LAIRMAN, 2000). Estes processos são as funcionalidades dos sistemas computacionais, as quais são descritas nos casos de uso, e que a UML

criou um diagrama específico para modelá-los.

Processos ou funcionalidades normalmente são descritos com verbos no infinitivo, como os exemplos a seguir (para descrever funcionalidades, normalmente inicia-se a frase com: "o sistema deverá"):

- Retirar dinheiro de um caixa automático (ATM);
- Encomendar um produto;
- Matricular-se em cursos numa escola;

Como os verbos são usados para descrever casos de uso, como mostrado, a técnica para extrair os casos de uso a partir dos requisitos funcionais, descritos na seção anterior, é a busca por verbos que possam levar às funcionalidades do software.

Para Lairman (2000), um diagrama de casos de uso mostra um conjunto de processos de um sistema, os atores do sistema, que são usuários e outras entidades que podem interagir com o sistema. Traz também as suas relações dos atores com os casos de uso. Os casos de uso são ilustrados com desenhos ovais e os atores são bonecos de palito. As linhas são associações entre os atores e os casos de uso em que eles interagem no sistema. O diagrama de casos de uso do software pós-processador pode ser visto no apêndice 2.

Os atores que irão interagir com o sistema são: operador CNC, designer de gemas e sistema de virtualização de máquina CNC. Eles são vistos no apêndice 2 e representam, respectivamente, a pessoa que irá preparar o sistema, carregando os dados necessários e fazendo a configuração, aquela que irá executar a geração do programa CNC a partir dos dados configurados, e por último o sistema que fará a virtualização da máquina CNC e será capaz de simular as trajetórias de acordo com o código CNC gerado.

#### 4.1.3. Diagrama de Classes de Domínio

O diagrama de domínio, também conhecido como diagrama conceitual, é uma das três principais visões do modelo de requisitos. Os outros são o SRS e o diagrama de casos de uso que foram elaborados anteriormente. De acordo com Sun (2003), ele representa o seguinte:

- As classes colocadas nele são abstrações-chave do sistema. O modelo é uma representação visual através de um diagrama de classes UML das abstrações que são descobertas durante a análise dos requisitos. Cada abstração torna-se uma classe no diagrama de classes.
- No diagrama mostra-se o relacionamento (colaboração) entre as abstrações-chave.

Para Lairman (2000) os seguintes passos são necessários para construir-se um modelo conceitual (diagrama de domínio do sistema):

1. Listar os conceitos candidatos, usando uma lista de categorias de conceitos e a identificação de substantivos relacionados com os requisitos que estão sendo considerados;
2. Desenhe-os em um modelo conceitual;
3. Acrescente as associações necessárias para mostrar os relacionamentos;
4. Acrescente os atributos necessários para completar os requisitos de informação.

Em resumo, a técnica consiste em procurar substantivos nos requisitos que poderão ser transformados em estruturas de dados que darão suporte aos processos estudados na seção anterior.

O modelo de domínio com a estrutura de dados do software pós-processador pode ser visto no apêndice 3.

## **4.2. Modelo de Projeto**

Na seção anterior, no modelo de requisitos, o objetivo era estudar o que o projeto iria realizar, descrevendo quais as funcionalidades seriam necessárias para obter um pós-processador para área de lapidação. Esta fase é chamada de fase de análise, e seu foco é responder à questão “o que?”. Agora, na fase chamada de Fase de Elaboração, vai ser tratada a questão “como?”. Ou seja, ela definirá como serão implementados artefatos que irão satisfazer os requisitos.

O modelo de projeto analisa os requisitos funcionais do modelo de requisitos para determinar a estrutura de componentes de projeto. Componentes de projeto são componentes abstratos de software que dão suporte ao comportamento especificado por um ou mais casos de uso. Exemplos de componentes de projeto são: elementos de interface com o usuário, serviços de negócio, e entidades de domínio. Um modelo de projeto captura os relacionamentos de colaboração entre os componentes de projeto para satisfazer um ou mais casos de uso (SUN, 2003).

Os diagramas desenvolvidos para o modelo de projeto serão: modelo de projeto propriamente dito, e diagramas de sequência. O modelo de projeto, desenvolvidos através de uma análise chamada análise de robustez (descrita por Jacobson em sua obra Engenharia de Software Orientada a Objetos, (JACOBSON, 1993)) darão uma visão da estrutura necessária para atender um dado caso de uso, enquanto os diagramas de sequência irão fornecer uma visão comportamental com a interação entre

componentes da aplicação.

#### 4.2.1. Modelos de Projeto

A análise de robustez é o nome dado ao processo que cria o modelo de projeto, e tem como entrada artefatos do modelo de requisitos como: diagrama de casos de uso e o modelo de domínio. Para ter uma maior abrangência no detalhamento dos componentes necessários para a aplicação o seguinte procedimento é realizado:

1. Selecciona-se um caso de uso;
2. Determina-se os cenários possíveis do caso de uso selecionado;
3. Determina-se os componentes necessários para atender cada cenário, usando-se do modelo de domínio para saber quais componentes dele irão compor um dado cenário;
4. Desenha-se o modelo de domínio para cada cenário possível.

Ao final deste procedimento tem-se o modelo de projeto que está ilustrado nos apêndices 4 até 9. A partir deste modelo foi possível obter os componentes básicos que serão desenvolvidos no sistema e suas interações com atores e entre componentes. Mais detalhes sobre estes componentes serão vistos na seção 4.3, onde serão detalhados a arquitetura com os componentes do pós-processador.

#### 4.2.2. Diagramas de Sequência

Diagramas de sequência são chamados diagramas de interação, e são usados para modelagem de aspectos dinâmicos da aplicação. Eles mostram interações entre objetos e seus relacionamentos, incluindo mensagens trocadas entre eles (BOOCH et.al., 2006).

Sendo assim, os diagramas de sequência, a partir dos componentes elicitados no modelo de projeto desenvolvido anteriormente, irão mostrar a dinâmica do sistema. Nele será possível identificar como será implementado os cenários mais importantes dos casos de uso da aplicação. Os diagramas são ilustrados nos apêndices 10 até 15, que encerram o modelo de projeto do software pós-processador.

### 4.3. Modelo de Arquitetura

Usado para o planejamento e projeto de sistemas, o modelo arquitetural faz a análise dos requisitos não-funcionais do modelo de requisitos para determinar os componentes de infraestrutura necessários para um sistema proposto. Estes componentes são componentes de software que darão suporte para o projeto de tal forma que irão satisfazer os requisitos não-funcionais (SUN, 2003).

Nesta fase do desenvolvimento aparecem também as primeiras seleções de tecnologias usadas na fase de construção, bem como a criação de uma estrutura de componentes e ou o uso de componentes de terceiros para atender a aplicação. O papel em uma corporação que é responsável por estas atividades, normalmente, é o arquiteto de software.

O modelo arquitetural do software pós-processador irá conter os seguintes artefatos:

- Um diagrama UML de pacotes. Este diagrama é útil na identificação de camadas de software e divisão conceitual de seus componentes.
- Um diagrama UML de componentes. Este diagrama mostra a distribuição dos componentes do sistema que irão dar suporte à execução do software e é usado para ilustrar em alto nível os componentes de software que serão implantados, bem como as interfaces destes componentes com outros, que permitirão a interatividade entre eles.

Antes de seguir com o desenvolvimento dos modelos, serão mostrados na próxima seção alguns conceitos básicos de padrões de projeto. Além da teoria a respeito de padrões de projeto, tendo em conta que são inúmeros padrões de projeto para área de computação, serão tratados alguns padrões de projeto em maior detalhe, justamente aqueles que serão usados no software.

#### 4.3.1. Padrões de Projeto

Cada padrão descreve um problema que ocorre recorrentemente em nosso ambiente, e então descreve o núcleo da solução para este problema de tal forma que se pode usá-la muitas vezes. Esta definição é usada na construção civil, porém é verdadeira para os padrões de projeto orientado a objeto (Gamma et. al., 1995). Esta definição de padrões de projeto foi citada no livro "Design Patterns" da conhecida "gang of four" ou Gof, e é bem clara em informar como a computação se fez valer das técnicas de construção para facilitar no projeto de software, especificamente o software orientado a objeto, onde componentes devem ser definidos para solucionar problemas do mundo real.

Para começar a descrever os padrões de projeto para o software inicia-se com o mais importante em qualquer projeto de software é estabelecer as responsabilidades necessárias e delegar a componentes responsabilidades bem específicas para haver coesão entre eles. Esta coesão significa que um deles deverá cumprir somente a sua

responsabilidade, nada menos e nada mais. Isto implica, por exemplo, que um componente que faz interface com o usuário não pode fazer a gravação de dados em um armazenamento. Quando mal elaborados, podem levar a sistemas frágeis, difíceis de manter, de reusar e de evoluir.

Padrões de projeto básicos na computação para realizar a desejada atribuição de responsabilidades é descrito por Lairman (2000), e são denominados padrões de projeto GRASP (*General Responsibility Assignment Software Patterns*). Os padrões GRASP são: Especialista (*Expert*), Criador (*Creator*), Alta Coesão (*High Cohesion*), Baixo Acoplamento (*Low Coupling*) e Controlador (*Controller*).

O padrão Especialista soluciona a atribuição de responsabilidade ao especialista da informação, ou seja, ele delega a responsabilidade ao componente que possui a informação necessária para realizar uma dada atividade.

O padrão de projeto Criador resolve o problema de qual componente tem a responsabilidade de criar instâncias de outro.

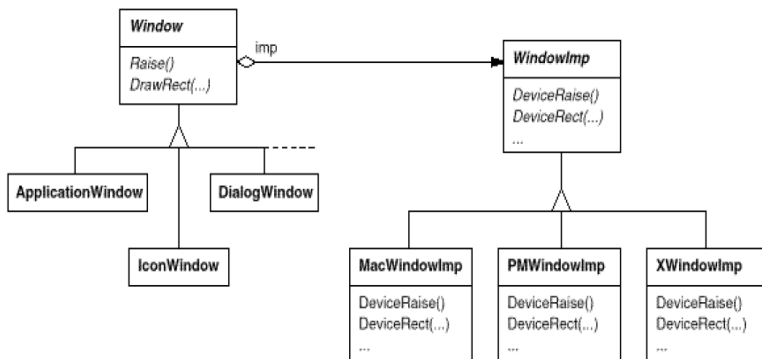
O padrão de projeto Alta Coesão é aquele que soluciona problemas de delegação de atribuições a componentes de software onde estas atribuições não pertencem a ele, ou seja, o componente não deveria estar recebendo tais atribuições. A alta coesão deverá ser mantida no sistema para que cada objeto realize somente aquilo para o qual ele é destinado. Um exemplo prático seria um dado componente de interface com o usuário sendo usado para interface com o usuário mais gravação de dados em um dispositivo de persistência. O efeito deste mal feito seria que ele realizaria estas duas atribuições e seria inviabilizado seu reuso individualmente, muitas vezes o reuso de todo ele seria comprometido.

Baixo Acoplamento é o padrão de projeto destinado a solucionar o problema de componentes que possuem uma forte dependência de entre eles. O software deverá ser arquitetado de maneira a manter em um nível baixo de dependência entre componentes do sistema. O exemplo visto anteriormente relacionado com o reuso da interface com o usuário e a gravação de dados revela um alto acoplamento entre as duas atribuições que resulta em um projeto de software fraco.

O Controlador é um padrão de projeto que resolve a localização do tratamento de eventos de uma aplicação. Todos os eventos que realizam operações reveladas nos casos de uso, usando o padrão Controlador, deverão ser localizados em um componente específico para este fim. Ou seja, a responsabilidade do tratamento de uma mensagem de um evento de um sistema será atribuída ao componente o qual será chamado Controlador.

Estes foram os padrões de projeto mais fundamentais para construção do software pós-processador. Além deles destaca-se os seguintes: *Abstract Factory* e MVC.

O *Abstract Factory* tem a seguinte função, conforme descrito por Gamma et. al. (1995): Fornecer uma interface para criar famílias de objetos relacionados ou dependentes sem especificar suas classes concretas. Este padrão de projeto permite a criação de elementos de software sem que o componente Criador conheça previamente aqueles que serão criados, o que ele conhece é somente a interface entre eles. Este padrão é importante para o projeto do pós-processador na implementação das extensões de entrada e saída declarados nos requisitos não-funcionais número 2 e 3, os quais podem ser vistos na tabela 4. Um exemplo de *Abstract Factory* pode ser visto na figura 31.



**Figura 31: Padrão de Projeto Abstract Factory**  
(Adaptado de GAMMA et. al., 1995)

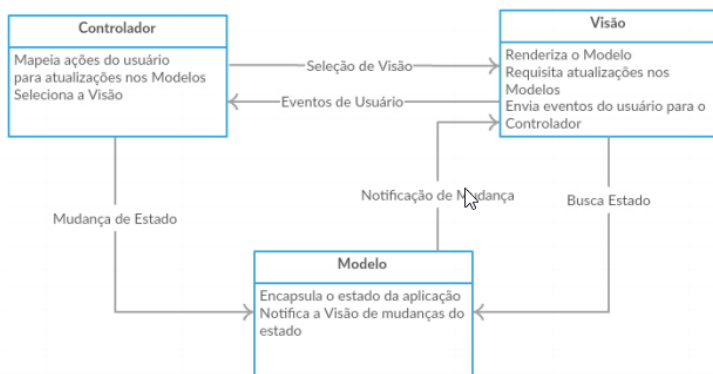
Na figura 31 é mostrado um diagrama de classes usando o padrão *Abstract Factory* para criar uma família de classes para implementar a interface *WindowImp*, que está sendo implementada em Mac, PM e XWindow. O padrão vai permitir que se instancie o elemento adequado para cada plataforma. Normalmente, detecta-se a plataforma antes e então seleciona-se qual dos componentes que implementam a interface será executado. Uma característica importante deste padrão é a possibilidade de inserção de novas implementações, novos plug-ins, em tempo de execução e sem a necessidade de alterar a aplicação. Isto é muito útil para realizar os requisitos de extensões de entrada e saída.



Para Gamma et. al. (1995), o padrão MVC consiste de três tipos de objetos (ou componentes). O *Model* é o objeto de aplicação, a *View* é sua representação em tela (tela de computador), e o *Controller* define a forma que a interface do usuário reage às entradas do usuário. O objeto ou elemento *Controller* é o mesmo visto anteriormente com o nome Controlador. Os demais são padrões inseridos agora.

O uso do padrão de projeto MVC é fundamental para que a aplicação fique modular. Através dele haverá a atribuição de responsabilidades dos componentes para interface com o usuário (*View*), os dados de domínio da aplicação (*Model*) e o tratamento das mensagens de interação de usuário.

Para que se possam elementos separados e específicos para eles, mantendo assim a coesão e o baixo acoplamento desejado na aplicação. Estes componentes e algumas interações entre eles pode ser visto na figura 32.



**Figura 32: Padrão Model-View-Controller**  
(Adaptado de SOMMERVILLE,2011)

Nas próximas seções serão usados na prática estes padrões de projeto, que ajudarão na manutenibilidade do pós-processador. Isto é alcançado na medida que mais desenvolvedores de software poderão imediatamente reconhecer a estrutura em que foi desenvolvido o pós-processador pelo fato de ele ser construído baseado em padrões de projeto amplamente conhecidos.

#### 4.3.2. Diagrama de Pacotes

Outro diagrama que ajuda na definição da estrutura de uma

aplicação é o diagrama de pacotes. Os pacotes ajudam a organizar os elementos de modelagem, e consistem em um mecanismo para agrupar elementos da modelagem, facilitando com isto o trabalho e a organização do projeto. O modelo de pacotes bem estruturado, onde seu conteúdo terá componentes com semântica semelhante e que se alteram em conjunto, ajudará na criação de aplicações com alta coesão e acesso controlado aos elementos internos.

O diagrama UML de pacotes está mostrado no apêndice 16. Nele pode-se inferir quanto ao uso do padrão de projeto MVC, representado pelos seguintes pacotes: *br.edu.ifsc.domain*, *br.edu.ifsc.ui* e *br.edu.ifsc.controller*. Estes pacotes conterão os elementos do domínio da aplicação ou modelos (*Model*), os de interface com o usuário (*View*) e os controladores, respectivamente. No diagrama também fica claro o uso de pacote específico para componentes de extensão do aplicativo, em *br.edu.ifsc.geometry*, para aqueles que tratarão de formatos CAD e *br.edu.ifsc.cnc*, que conterà outros que irão implementar diferentes comandos CNC.

Há também o pacote *br.edu.ifsc.dao*, que conterà os elementos que irão realizar a abstração do acesso a dados. Este pacote será explicado em mais detalhes na próxima seção, que tratará diretamente dos componentes que irão pertencer aos pacotes definidos.

### 4.3.3. Diagrama de Componentes

Para Booch et. al. (2006), um componente é a parte lógica substituível de um sistema ao qual se adapta e fornece a realização de um conjunto de interfaces. Um bom projeto de software permite a substituição daqueles mais antigos por mais novos, sendo eles compatíveis com as interfaces, isto sem alterações no funcionamento do núcleo da aplicação.

O apêndice 17 mostra o diagrama de componentes elaborado para o pós-processador. Nele encontram-se os seguintes elementos de software: FaceterUI, FaceterData, FaceterService, DAOFactory, GemetryFactory e CNCFactory. Este diagrama é de alto nível de abstração e será detalhado no modelo de solução que fará um refinamento deste diagrama.

O FaceterUI irá implementar a interface com o usuário do sistema, a parte “*View*” do padrão de projeto MVC.

O FaceterData irá implementar internamente os componentes de domínio do sistema, descritos no diagrama de domínio, eles serão parte “*Model*” do MVC.

O FaceterService irá implementar os controladores que vão tratar

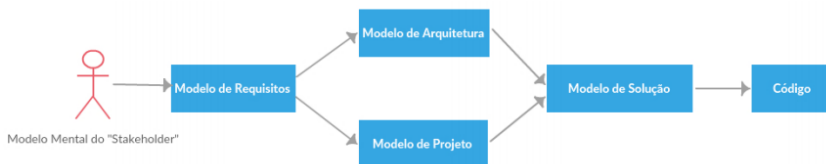
as interações dos usuários do sistema, como descrito no padrão de projeto Controller e faz parter do “*Controller*” do padrão MVC.

DAOFactory será aquele responsável por abstrair do elemento de serviço o método de persistência de dados junto a um dado armazenamento, que poderá ser um banco de dados ou arquivos gravados em disco. Em resumo, o componente de serviço FaceterService envia dados para serem gravados ou recuperados através do DAOFactory. Este, por sua vez, irá responsabilizar-se pela atividade, enquanto o chamador não deverá se preocupar como será implementado a gravação, a única preocupação é receber o resultado da operação. O DAOFactory será baseado no padrão de projeto AbstractFactory, visto anteriormente. Portanto, ele poderá ter várias famílias de implementação de persistência, causando uma flexibilidade na aplicação relacionada ao mecanismo de persistência, ou seja, a persistência poderá ser implementada usando um banco de dados relacional, ou hierárquico, ou, simplesmente gravação de dados em arquivos no sistema de arquivos.

Outros dois componentes que implementam o AbstractFactory são: GeometryFactory e CNCFactory. Isto por que eles também terão famílias de implementações diferentes para fazer a leitura de fontes de dados com estruturas distintas. Em outras palavras, formatos de dados gravados em arquivo com *layouts* diferentes. O GeometryFactory possuirá implementações para arquivos CAD, especialmente o formato GemCad®, que é a ferramenta CAD mais popular na lapidação de Gemas, enquanto o CNCFactory irá conter implementações para diferentes comandos CNC, sendo que a implementação mais importante será a do formato ISO 6983. Estes dois formatos estão detalhados na seção 2.2.5 do capítulo 2.

#### **4.4. Modelo de Solução**

Os modelos de projeto e arquiteturais são unidos para fornecer o primeiro rascunho do modelo de solução, que então poderá ser refinado para o modelo final. O modelo de solução é a base sobre a qual a equipe de desenvolvimento irá construir o código da solução do sistema, e como descrito anteriormente, o modelo é derivado da junção dos modelos arquiteturais e de projeto, conforme a figura 33 (SUN, 2003).



**Figura 33: Geração do Modelo de Solução (SUN, 2003)**

Nesta seção, além do modelo de solução propriamente dito, serão desenvolvidos os protótipos das interfaces com usuários. As interfaces com os usuários são interessantes porque a usabilidade pode ser crítica para o sistema. Do ponto de vista do usuário da aplicação torna-se crítico e agradável poder interagir com um sistema de fácil operação. Do ponto de vista da equipe de engenharia, a qual irá codificar o sistema, é de fundamental importância ter algo mais concreto do que apenas diagramas. Isto tornará o ciclo de vida do sistema mais produtivo e, principalmente, mais assertivo.

#### 4.4.1. Modelo de Solução

O modelo de solução é uma junção dos modelos de projeto, também conhecido como modelos de análise, com o diagrama de componentes visto no modelo arquitetural. Haverá uma junção dos modelos de projeto em um único diagrama, bem como a expansão dos componentes oriundos do diagrama de componentes. Em consequência disto teremos um modelo estrutural expandido da aplicação. Esta expansão é conveniente para o nível de detalhamento que se deseja neste ponto do projeto, entretanto, o tamanho do diagrama será elevado proporcionalmente, o que causa dificuldade para leitura, mas nada comprometedor.

O modelo de solução é mostrado no apêndice 18. Nele é possível identificar: os atores, os componentes de interface com o usuário, serviços, DAO, de integração e de estrutura de dados. Mostra também em alto nível as interfaces de interação entre eles. Os componentes de interface com o usuário ficam dentro de um componente container chamado FaceterUI, e mostra no diagrama que haverão, basicamente, 7 telas diferentes que vão compor a interface com o usuário do sistema, todos os componentes dentro do container. Estas telas serão detalhadas na seção 4.4.2.

Os componentes de serviço, no container FaceterService, vão

atender requisições dos usuários feitas através da interface com o usuário do componente FaceterUI. Basicamente, haverá um componente de serviço para atender cada tela do sistema.

DAOFactory é o componente container para os componentes que irão abstrair o acesso a dados da camada de serviço. Através deste desenho usando DAO, é possível realizar a substituição sem impacto no sistema de persistência, podendo assim o sistema usar bancos de dados, sistema de arquivo, ou outro tipo de armazenamento ou persistência.

FaceterData é o componente que contém as classes de domínio do sistema. Todos os estes elementos possuem dependência (representado pela seta descontínua) do FaceterData, pois os dados precisam estar disponíveis para serem apresentados, processados e armazenados, que são responsabilidade dos elementos FaceterUI, FaceterService, e DAOFactory, respectivamente.

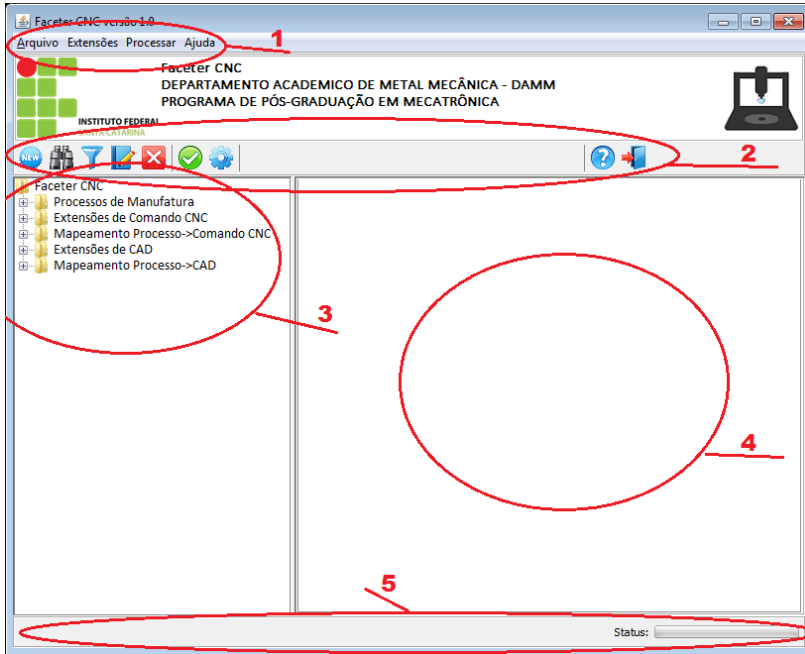
Finalmente temos os componentes de extensão do software pós-processador: GeometryFactory e CNCFactory. Neles pode-se ver que há extensões para o software GemCad® e ISO6983 (discutidos no capítulo 2). O FaceterService fornecerá uma interface que poderá ser implementada criando-se diferentes *plug-ins* para a aplicação. Estes dois *plug-ins*, GemCad® e ISO6983, serão implementados neste projeto, mas haverá a possibilidade de criação de novos e também inclusão no sistema em tempo de execução. Esta é uma flexibilidade visando permissão de expansão do software, o que configura um sistema extensível.

#### 4.4.2. Desenho de Interface com o Usuário

O usuário irá interagir com o sistema através de uma interface gráfica que deverá mostrar o estado da aplicação com o objetivo de gerar novos programas CNC baseados em modelos feitos em sistemas CAD. Nesta seção serão mostrados os desenhos de protótipos iniciais das telas, que são baseados nos requisitos do sistema e nos dados que deverão ser processados pelo sistema, mostrados no modelo de domínio no apêndice 3. As interfaces com o usuário são mostradas como componentes no modelo de solução do software, apêndice 18. Os oito componentes vistos no modelo são: MainUI, ManufacturingProcessUI, NCCControlUI, MPNCCControlMappingUI, GeometryFormatUI, MPGeometryMapping, MPSpecificationUI e NCCCodeGenerationUI. Estas são as interfaces ou telas mais importantes do sistema e serão apresentadas a seguir.

A tela principal do software, componente chamado de MainUI, tem seu protótipo mostrado na figura 34. O protótipo mostra que a tela principal será dividida em 5 partes principais, a barra de menus (1), a barra

de ferramentas (2), a árvore com os objetos (3), quadro edição de objetos (4), e a barra de estado da aplicação (5). A barra de menus contém acesso às funcionalidades da aplicação, onde todas as funcionalidades terão acesso através de menu. Na barra de ferramentas haverá acesso às



**Figura 34: Protótipo da tela principal do software Pós-Processador (elaborado pelo autor)**

principais funcionalidades oriundas das telas que serão apresentadas no quadro de edição de objetos (4). Esta barra é para acesso rápido às funcionalidades.

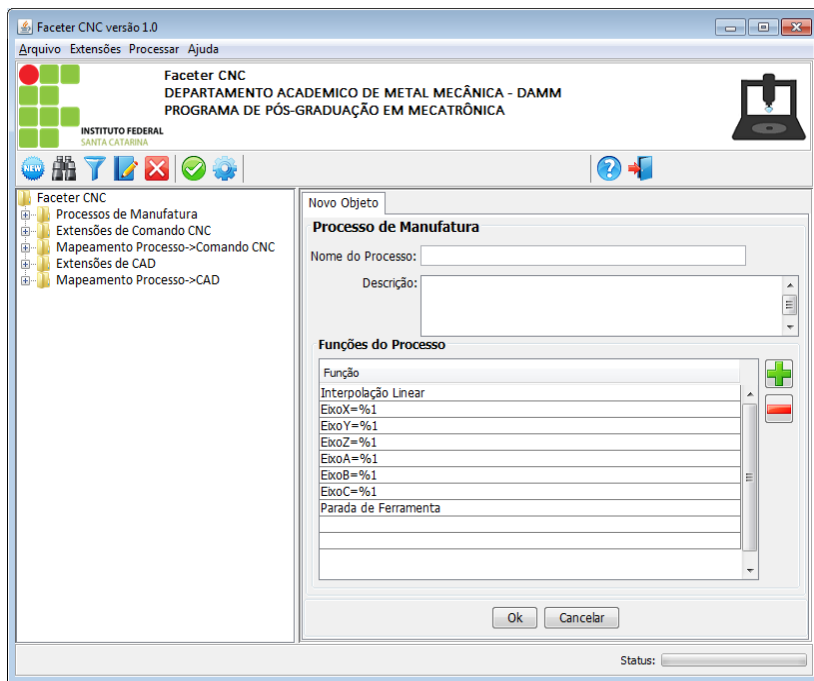
A árvore de objetos irá possibilitar ao usuário navegar pelos objetos que estão no sistema e verificar seus estados, bem como possibilita o acionamento de solicitações de inclusão, recuperação, atualização e exclusão dos mesmos.

O quadro de edição de objetos é onde alguns painéis de edição serão mostrados para possibilitar a manutenção de objetos do sistema. Alguns dos protótipos de telas mostrados serão exibidos neste espaço como painéis de edição do sistema.

Para finalizar a barra de estado do sistema tem finalidade de

mostrar ao usuário que o sistema está em processamento de suas solicitações, assim os usuários poderão saber quando e como as suas solicitações estão sendo processadas pelo aplicativo.

O componente chamado ManufacturingProcessUI, é também uma interface com o usuário que é mostrada como um painel de edição no quadro de edição de objetos da tela principal, seu protótipo pode ser visto na figura 35.

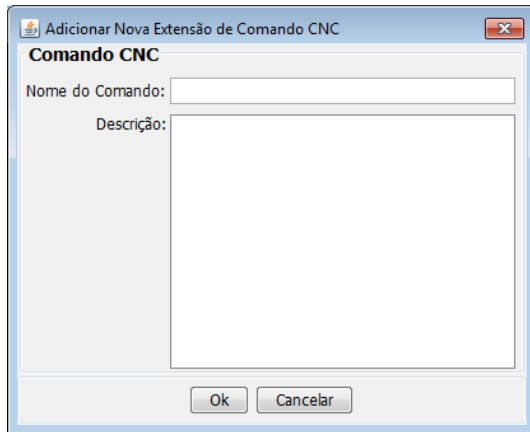


**Figura 35: Protótipo de edição de atividade de manufatura (Elaborado pelo autor)**

Na figura é mostrado o protótipo do painel de edição de manufatura. Este componente permitirá a edição de dados dos processos, onde os principais dados são as funções do processo, que são comandos que serão usados na definição dos processos em linguagem agnóstica. A linguagem é estabelecida neste ponto, onde pode-se ver a palavra “EixoX=%1” na tela. Isto estabelece esta sentença como declaradora de eixos X em uma máquina, e argumento “%1” será substituído pelo valor em que tal eixo será deslocado. Para cada comando necessário uma entrada nesta tabela deverá ser definida.

O próximo componente é o NCControlUI. Ele permitirá a inclusão de novos elementos de comando CNC ou controle CNC, chamados neste projeto de extensão de software.

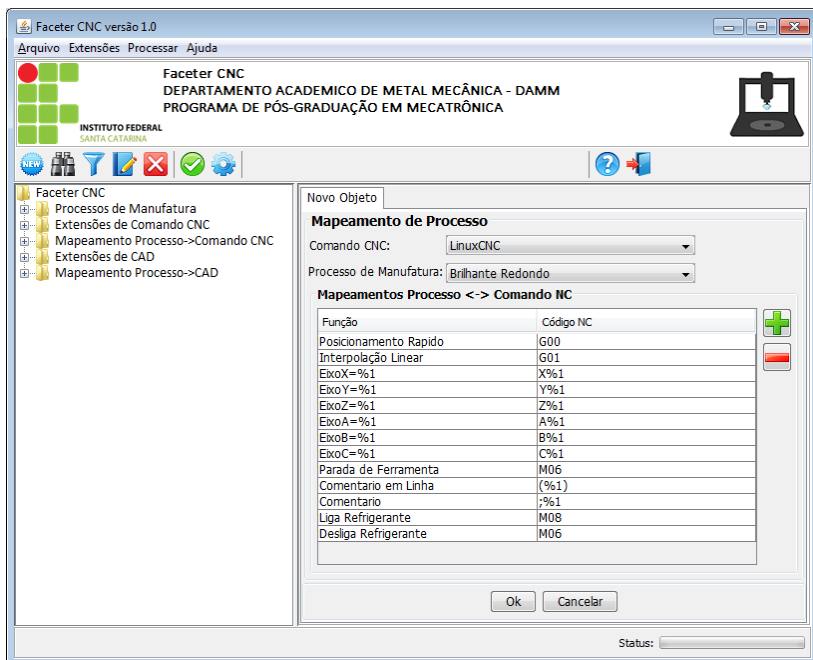
Os controles CNC possuem comandos particulares dependentes do fabricante, muitas vezes baseado no ISO 6983, mas com variantes que os tornam incompatíveis. Para isto, cada comando NC que vai controlar máquinas de lapidação precisa ser inserido no sistema. Eles são agrupados pelos dados fornecidos no NCControlUI, e seus comandos específicos serão definidos no componente MPNCControlMappingUI, o qual faz o mapeamento de comandos definidos no processo de fabricação, visto anteriormente, para códigos NC específicos dos controles CNC definidos na tela mostrada na figura 36.



**Figura 36: Protótipo de tela da edição de Comandos CNC  
(Elaborado pelo autor)**



No componente MPNCCControlMappingUI serão definidos o mapeamento entre o código de programa do controle CNC com os códigos das atividades de manufatura. O protótipo da interface pode ser visto na figura 37. Neste painel de edição poderá ser selecionado qual é o comando CNC e a atividade de manufatura, inserindo-se uma tabela de

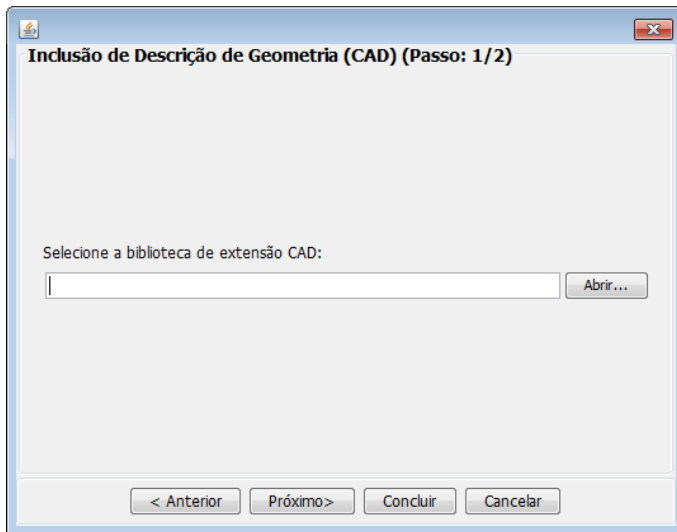


**Figura 37: Mapeamento de atividade de manufatura x Comando CNC (Elaborado pelo autor)**

mapeamento de comandos que faz a relação entre eles. Um exemplo mostrado na figura é que a palavra “Interpolação Linear”, usada nos processos de manufatura será substituída por “G01”, que é o código ISO para realizar interpolações lineares em comandos CNC, na figura foi selecionado o Linux CNC, por exemplo.

Como visto anteriormente, haverá uma extensão de software para incluir novos comandos CNC, e outra extensão que poderá ser incluída são formatos de geometria feitos em sistemas CAD. O componente de interface com o usuário responsável pela inclusão de geometrias é o GeometryFormatUI, e seu protótipo pode ser visto nas figuras 38 e 39.

No passo 1, mostrado na figura 38, o usuário irá dizer qual é o

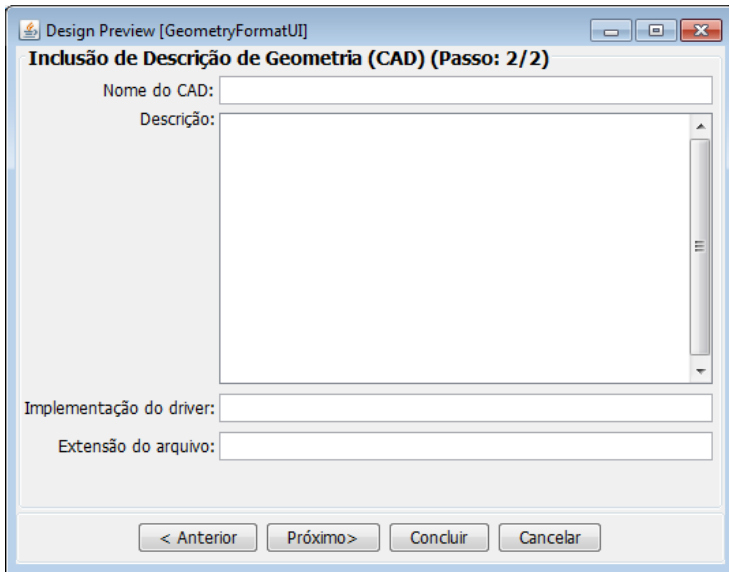


**Figura 38: Protótipo de tela de inclusão de extensão de CAD, passo 1 (Elaborado pelo autor).**

arquivo que implementa a extensão de software. Neste arquivo deverá haver código em linguagem de programação Java que implemente a interface GeometryProvider mostrada no modelo de solução, apêndice 18. Concluído o passo 1, o usuário vai pressionar o botão “Próximo” e o passo 2, mostrado na figura 39, irá aparecer. Os demais dados pertencentes à extensão CAD poderão ser preenchidos. São eles: nome do

CAD, descrição, implementação do driver, e extensão do arquivo. O campo nome é auto-explicativo, conterá o nome do CAD, por exemplo, GemCad®; o campo descrição é uma breve descrição do CAD inserido; o campo implementação do driver é onde se vai declarar o nome da classe que implementa a interface a qual deve estar incluída na biblioteca inserida no passo 1; E para finalizar, o campo extensão do arquivo é a extensão padrão do arquivo CAD, por exemplo, ASC, que é a

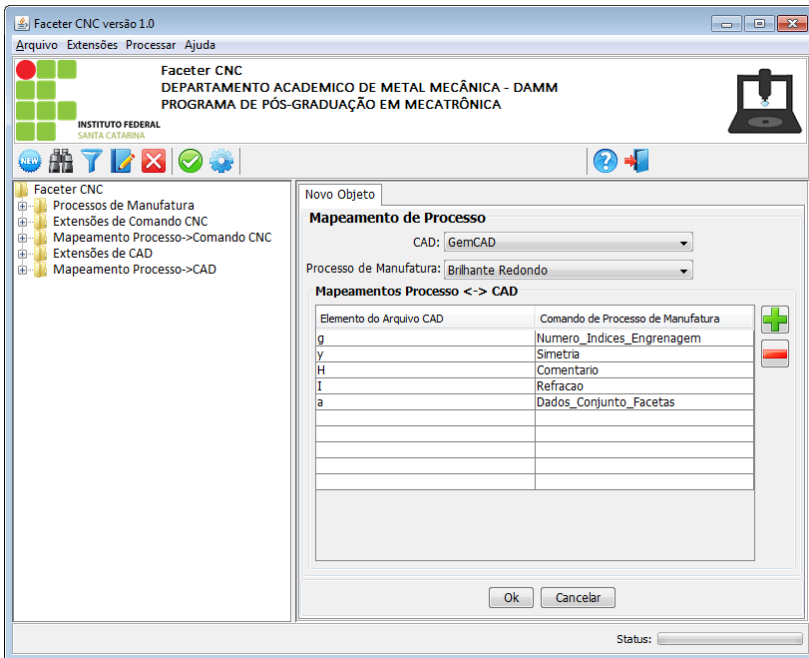
extensão do software GemCad®.



**Figura 39: Protótipo de tela de inclusão de extensão de CAD, passo 2 (Elaborado pelo autor).**

Tão logo a geometria tenha sido inserida no aplicativo, o próximo passo seria mapear a geometria para elementos de memória que serão usados na atividade de manufatura. O componente de interface com o usuário responsável por este mapeamento é o MPGeometryMapping, que é um painel de edição no quadro de edição de objetos da tela principal. O seu protótipo pode ser visto na figura 40.

Na tela de mapeamento de atividade de manufatura com extensões CAD mostrada pode-se ver os seguintes campos: CAD e atividade de manufatura, seguidos da tabela de mapeamento propriamente dita. No campo CAD é selecionado o CAD que será um dos incluídos através da interface mostrada na seção anterior. Para o campo de atividade de manufatura deverá ser selecionada qual atividade de manufatura será mapeada, este processo foi incluído através da interface da figura 34, mostrada anteriormente.

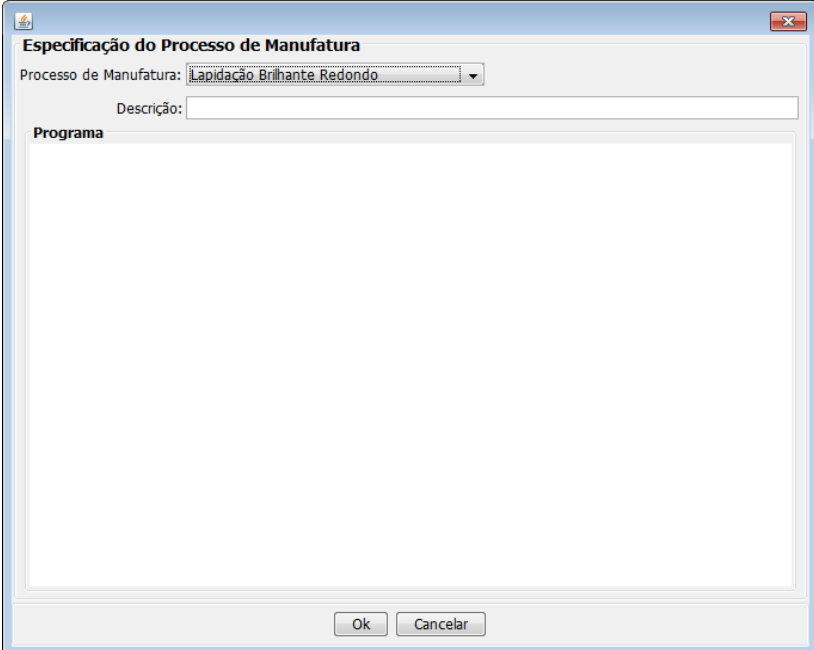


**Figura 40: Mapeamento de atividade de manufatura x CAD (Elaborado pelo autor).**

Na tabela de mapeamento deverá constar o elemento de origem e o de destino. Na tabela de origem, definido como “Elemento do Arquivo CAD”, haverá um elemento gerado pelo aplicativo a partir do arquivo CAD, na figura ve-se o “g”. Este símbolo referencia um valor oriundo do arquivo CAD e poderá ser usado na definição da atividade de manufatura através do comando definido na coluna “Comando de Atividade de Manufatura”.

O penúltimo componente de interface com o usuário listado é o MPSpecificationUI, e é um dos mais importantes. Nele, se especificam os processos de manufatura. Na lapidação de gemas, por exemplo, é neste componente que haverá a entrada de dados do programa que irá fazer uso dos comandos de linguagem agnóstica criada a partir da inclusão das extensões de software vistas anteriormente, e a partir deste comando se definir como será por exemplo a fabricação de uma gema lapidada no *design Simple Jack*, ou qualquer outro *design*. O protótipo da interface é

mostrado na figura 41. Nesta tela serão mostrados três campos, a saber: atividade de manufatura, descrição e programa.

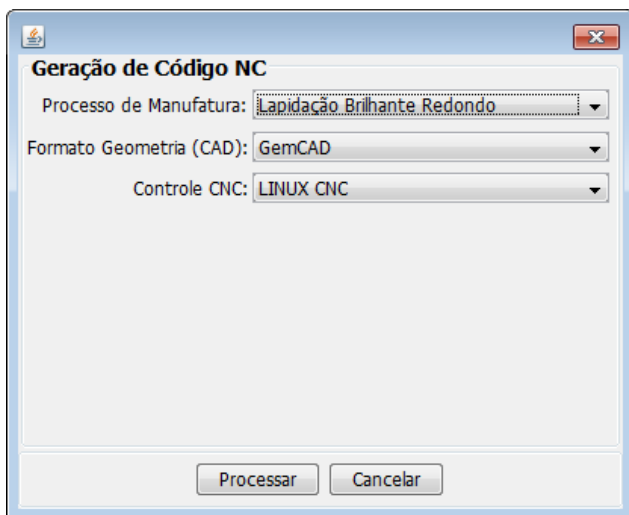


O protótipo da interface, intitulado "Especificação do Processo de Manufatura", apresenta um layout claro e funcional. No topo, há um menu suspenso rotulado "Processo de Manufatura:" com o valor "Lapidação Brilhante Redondo" selecionado. Abaixo dele, um campo de texto rotulado "Descrição:" está atualmente vazio. O maior elemento da interface é um campo de texto rotulado "Programa", que ocupa a maior parte da área central e está também vazio. Na base da janela, há dois botões de ação: "Ok" e "Cancelar".

**Figura 41: Protótipo da interface de especificação da atividade de manufatura (Elaborado pelo autor)**

No primeiro seleciona-se a atividade de manufatura que será usado. No seguinte, a descrição, que é uma descrição do processo para facilitar o reconhecimento. No campo programa, o último, é onde o programa em linguagem não dedicada a comando CNC irá ser inserido. A criação desta linguagem ocorre no processo de definição dos componentes de mapeamento, visto anteriormente. E esta linguagem permite que o software seja independente de comando CNC na definição dos comandos para executar a manufatura.

O último elemento de interface com usuário a ser discutido é o NCCodeGenerationUI, que será responsável pela interação com o usuário quando este solicitar a geração de código NC. É uma tela simples e seu protótipo pode ser visto na figura 42.



**Figura 42: Protótipo da interface de geração de código NC (Elaborado pelo autor).**

Apenas três campos simples compõem esta tela, vistos do topo para baixo, o primeiro para selecionar a atividade de manufatura, depois o formato CAD e então o último seleciona o controle CNC de destino do código.

# Capítulo 5

## AVALIAÇÃO DAS FUNCIONALIDADES

---

A avaliação das funcionalidades do software pós-processador passa pela construção de um modelo virtual de referência de uma máquina lapidadora CNC, que dará apoio à geometria da máquina, ou seja, nas dimensões, para gerar as trajetórias de corte partindo do software de controle. A definição detalhada do escopo do trabalho encontra-se na seção 1.1.

Nas seções a seguir serão abordados os resultados obtidos e uma avaliação da realização destas funcionalidades durante o projeto.

### 5.1. Modelo 3D de Máquina de Lapidar CNC

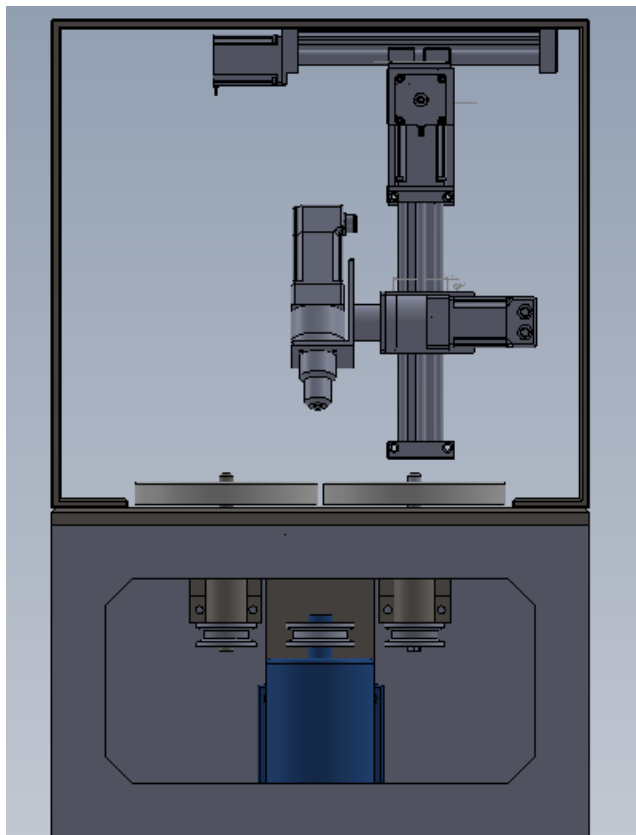
Para possibilitar ao software executar a geração do código CNC a equipe do projeto desenvolveu um modelo virtual de lapidadora com a geometria necessária para a máquina de lapidação. Esta máquina, por usar a tecnologia de corte conhecida como *Faceting Head*, recebeu o nome de *Faceting Head CNC*, e tem sua vista frontal mostrada na figura 43.

A máquina teve seu *design* baseado nas características de uma máquina *Faceting Head* manual, prevendo automação em cada um dos seus eixos motores para posicionamento linear ou rotacional, bem como um motor na parte inferior da máquina para o eixo árvore. Este último motor já existente nos modelos de máquina de lapidar *Faceting Head* convencionais.

Embora para o desenvolvimento do software não seja necessário pormenorizar os componentes da máquina, algumas informações da geometria são de suma importância para possibilitar a geração do código CNC e cumprir, com isto, o requisito de software mais importante, discutido na seção 4.1.1, que é: “O sistema deverá gerar programas CNC a partir de arquivos de descrição de geometria de gemas lapidadas”. Os dados mais importantes obtidos através deste modelo da lapidadora são: as dimensões da mesa da lapidadora, comprimento do curso de cada eixo, dimensões dos discos, e a cadeia cinemática da lapidadora.

Vale salientar que o modelamento da máquina foi realizado com enfoque na cinemática gerada a partir dos graus de liberdade necessários à realização das operações de lapidação. Não houve uma preocupação,

neste momento, de detalhar o projeto e a seleção de componentes visando otimizar as características de operação da máquina. O projeto detalhado da máquina foge ao escopo deste trabalho.

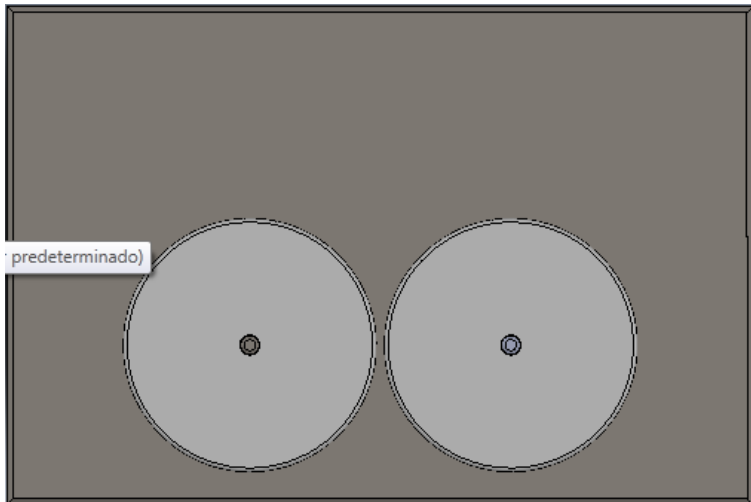


**Figura 43: Vista frontal do modelo de máquina de lapidar Faceting Head CNC (Elaborado pela Equipe do Projeto)**



### 5.1.1. Mesa da Lapidadora

A mesa da lapidadora é onde ocorre o corte das pedras, ou em outras palavras, a lapidação. O projeto inclui na mesa duas unidades de corte, representadas por dois discos abrasivos rotativos. Um possui maior granulometria do abrasivo para efetuar a lapidação (operação de desbaste) e outro com menor granulometria para o polimento (operação de acabamento), como pode ser visto na figura 44.



**Figura 44: Mesa da Lapidadora (Elaborado pela Equipe do Projeto)**

O software pós processador vai se basear na geometria desta mesa para realizar as trajetórias de lapidação nos discos abrasivos. Estas trajetórias executarão a operação de desbaste em  $2 \frac{1}{2}$  eixos, como visto na seção 2.1.2. Ele prevê o posicionamento da pedra e realiza a operação de usinagem através de deslocamento no eixo Z.

A mesa da lapidadora virtual mede 590 mm de comprimento e tem 390 mm de profundidade, por onde o cabeçote irá se orientar nos eixos nomeados X e Y. X é o comprimento e Y é a profundidade.

As posições para os discos lapidação foram projetadas para discos de 8 polegadas (200 mm), que seguem um tamanho padrão da indústria de lapidação. Porém, outra opção seria usar discos de 6 polegadas (150 mm) que adaptariam-se à geometria. Entretanto isto teria reflexo imediato na programação das atividades de manufatura feita na interface do sistema mostrada na figura 41, pois a mudança do tamanho

do disco interfere na trajetória do cabeçote para executar a lapidação.

## 5.2. Funcionalidades do software

Para que o pós-processador funcione perfeitamente, dois atores, ou operadores, são necessários no sistema. Eles podem ser percebidos no diagrama de casos de uso no apêndice 2.

Um dos atores é necessário para a preparação das configurações, um “*setup*” da aplicação, que é feito pelo ator chamado “Operador CNC”. Ele será responsável por executar a inserção e mapeamento de novos formatos CAD de entrada, ou seja, inserir novas extensões de software CAD. Além disso, será responsável por realizar a inserção e mapeamento de novos comandos CNC e a definição das atividades de manufatura. Tudo o que é desenvolvido pelo operador CNC são rotinas mais técnicas, que exigem conhecimentos específicos de maquinaria e comandos CNC.

Outro perfil de ator que irá usar o sistema, descrito no apêndice 2, é o “Designer de Gemas”. Este usuário irá ter um perfil menos técnico, irá realizar a construção de peças no CAD e irá contar com o “*setup*” da máquina para, com o arquivo de saída do seu design CAD, dar entrada no software pós-processador para geração do código CNC que poderá ser então manufaturado na máquina lapidadora.

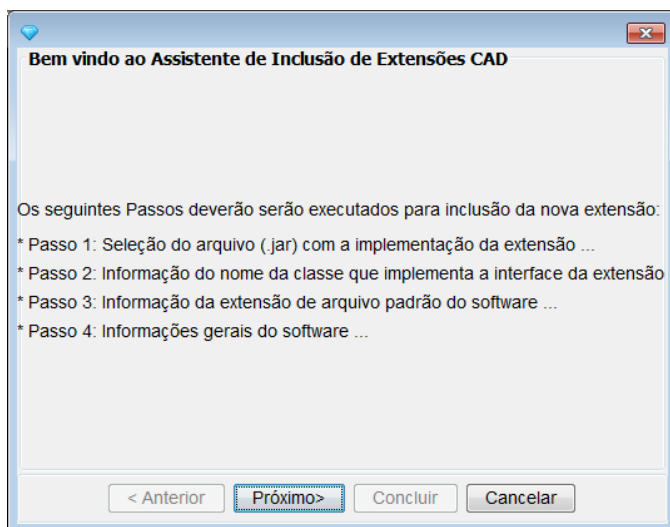
Estas funcionalidades executadas pelos atores, envolvidas na fase do software pós-processador da cadeia de manufatura elaborada neste trabalho para lapidação de gemas, são discutidas a seguir.

### 5.2.1. Inserção e Mapeamento de Novos Formatos CAD de Entrada

Os módulos de entrada, discutidos na seção 2.2.5 e mostrados na figura 10, são fornecidas para o software pós-processador através da inserção e mapeamento de novos formatos CAD de entrada.

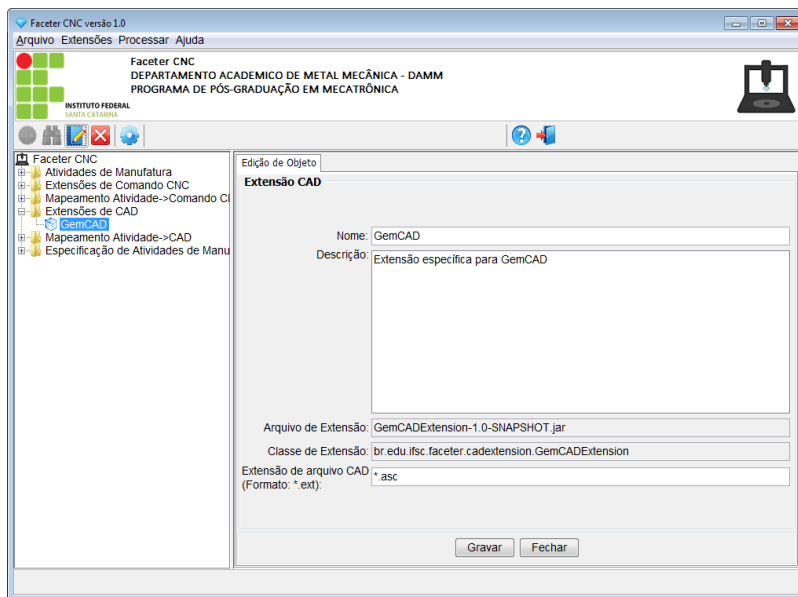
Para executar a inserção, o usuário operador CNC irá encontrar no software, através do menu Extensões -> Extensões de Geometria CAD -> Adicionar, a funcionalidade de inserir novos “*plug-ins*” no software pós processador, cujas informações permitirão ao software reconhecer diferentes formatos de arquivos de software CAD. Na versão inicial do software foi concebido um “*plug-in*” para o software GemCad®, que é específico da área de lapidação. Nos apêndices 17 e 18 pode ser visto a arquitetura que proporcionou a flexibilidade de inclusão de novas extensões com uso do padrão de projeto chamado *Abstract Factory*, discutido na seção 4.3.1.

Depois de selecionar o menu, o usuário irá ver um assistente de inclusão de extensões CAD, conforme pode ser observado na figura 45.



**Figura 45: Assistente de Inclusão de Extensões CAD (Elaborado pelo autor)**

Este assistente possui quatro passos e, ao final de sua execução, a nova extensão estará incluída no software, como pode ser visto na figura 46. Note-se que, na figura 46 pode ser vista a extensão do arquivo CAD, que no caso do GemCad® a extensão é “.asc”. A extensão do arquivo é um dos dados coletados durante a inclusão do “*plug-in*” e tem como finalidade filtrar os arquivos durante o processo de geração do programa CNC que será discutido posteriormente.



**Figura 46: Extensão do Software GemCad® incluída (Elaborado pelo autor)**

Outra informação importante coletada durante o processo de inclusão é a biblioteca que irá implementar a interface de extensão GeometryProvider (pode ser vista nos apêndices 17 e 18). Esta biblioteca deverá ser implementada para cada fornecedor de CAD, para que sejam lidas as informações dos arquivos e disponibilizadas através da interface GeometryProvider fornecida. É uma camada de integração que foi feita para o software que permite que lhe sejam incorporados novos software CAD em tempo de execução. O que significa, em última instância, que o software pós-processador é uma ferramenta extensível.

A implementação da *Abstract Factory* mencionado permite que seja inserido em tempo de execução uma nova extensão de formato de arquivo de software CAD. Para implementar isto em linguagem Java foi necessário a criação de uma interface, mostrada no diagrama de modelo de solução (Apêndice 18) chamada GeometryProvider.java.

Esta interface é de suma importância, pois informa quais métodos devem ser implementados por diferentes softwares CAD que podem ser incluídos no pós-processador em tempo de execução. Quatro métodos são definidos para serem implementados: getVariablesTypes(),

`getVariables()`, `setCADFile()`, e `execute()`.

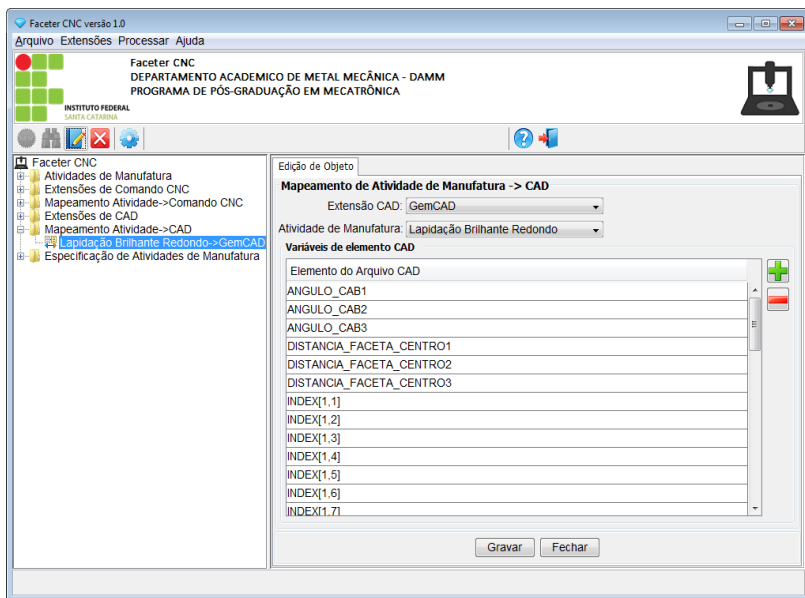
A implementação de `getVariablesTypes()` deverá ser usada para o software pós-processador obter um mapa com os tipos de dados das variáveis criadas durante a leitura do arquivo CAD. Este mapa usa como chave do tipo *String* e tem o valor do tipo *Class*. O valor do mapa diz de qual classe é a variável, ou seja, diz o tipo do dado guardado pelo mapa através da chave *String*. Um exemplo de instância seria “A”->Java.lang.Long. Que significa que a chave “A” está apontando para um tipo de dado *Long*.

`getVariables()` deverá ser usado para definir os valores dos dados obtidos dos arquivos CAD. Uma instância seria “A”->1500. Onde a chave “A”, definida anteriormente, teria o valor 1500. Este valor deverá ser pego do arquivo CAD que pode ser o GemCad®, SolidWorks®, ou qualquer um que esteja sendo implementado.

`setCADFile()` deverá ser usado para definir o nome do arquivo CAD que deverá ser processado. Isto ocorre quando o software pós-processador seleciona um arquivo e envia ao componente que implementa a interface (mostrado no apêndice 18) para que ele processe e entregue os mapas definidos anteriormente devidamente preenchido de acordo com a lógica definida no método `execute()`.

O método `execute()`, como mencionado, faz o processamento do arquivo CAD e preenche os mapas que serão expostos através dos métodos *getters* recém mostrados.

Após a finalização da inclusão da extensão CAD o software necessita que seja feito o mapeamento das variáveis lidas do modelo CAD para que elas sejam depois usadas na descrição das atividades de manufatura. Isto permite, de forma desacoplada, trazer dados do arquivo CAD, independente do fornecedor, para serem processadas em uma linguagem não-dedicada que fornece as atividades a serem realizadas durante a manufatura da peça. Isto é útil não só para lapidação de gemas, mas é usada em outras áreas da manufatura computadorizada através de softwares CAM. O software pós-processador realizará este mapeamento associando uma extensão CAD incluída, conforme discutido anteriormente, com uma atividade de manufatura, tal atividade posteriormente será descrita através de uma linguagem não-dedicada, discutida na seção 5.2.3. A figura 47 mostra o mapeamento.



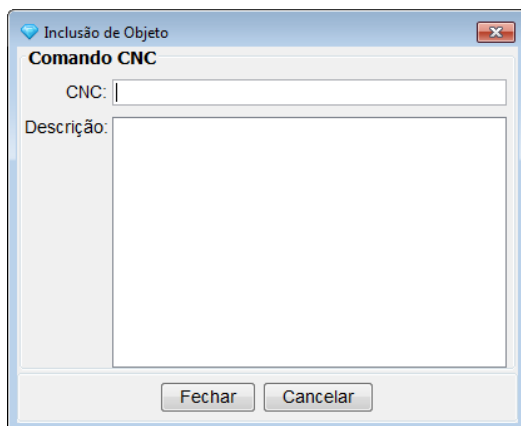
**Figura 47: Mapeamento de Atividades de Manufatura para elementos CAD (Elaborado pelo autor)**

Na figura 47 percebe-se que, depois de vincular uma extensão CAD a uma atividade de manufatura, define-se as variáveis que serão preenchidas pela biblioteca de extensão ou *plug-in* e poderão ser usadas na descrição da atividade (descrita na seção 5.2.3). Sendo assim, o mapeamento feito é justamente trazer da geometria CAD informações e mapear para as variáveis criadas na lista mostrada na figura, lista dos “Elementos do Arquivo CAD”. Este mapeamento deverá ocorrer em consonância com a equipe de desenvolvimento da biblioteca de extensão, os quais irão trabalhar com uma estrutura de dados capaz de efetuar buscas através de uma chave. A chave seria exatamente o nome da variável ou elemento do arquivo CAD. Nomeadamente esta estrutura de dados é uma tabela de *hash*.

### 5.2.2. Inserção e Mapeamento de Novos Comandos CNC

Da mesma maneira que feito com módulos de entrada, conforme mostrado na figura 10, pode-se incluir extensões ou módulos de saída. Estes módulos de saída são adaptadores para diferentes comandos CNC. O modelo de requisitos, visto na seção 4.1 tem, na tabela 3, seu requisito

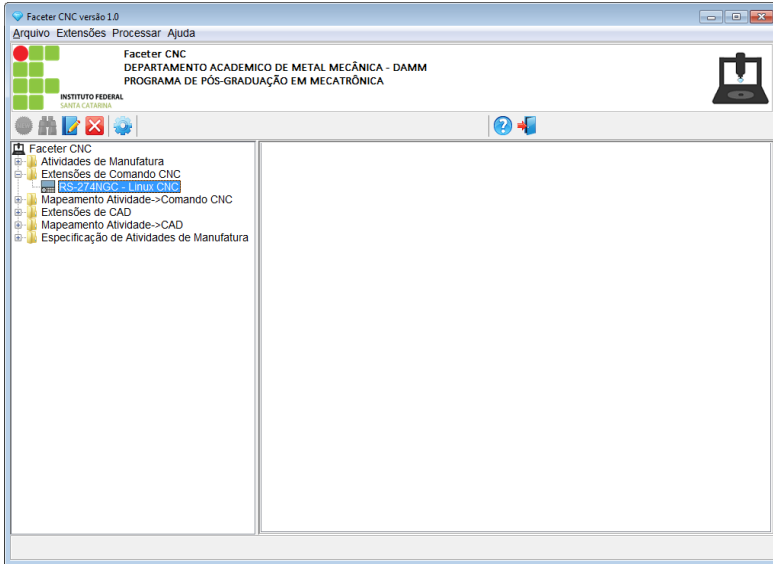
número 4, que solicita que “O sistema deverá mapear linguagem de descrição do processo de fabricação para comando CNC”. Além da inclusão de novas extensões de comando CNC, este requisito também menciona o mecanismo de mapeamento entre a linguagem de descrição da atividade de manufatura e o código CNC correspondente. A realização destes requisitos é acionada através do menu Extensões -> Extensões de Comando CNC -> Adicionar, que mostrará a tela ilustrada na figura 48.



**Figura 48: Inclusão de Comando CNC (Elaborado pelo autor)**

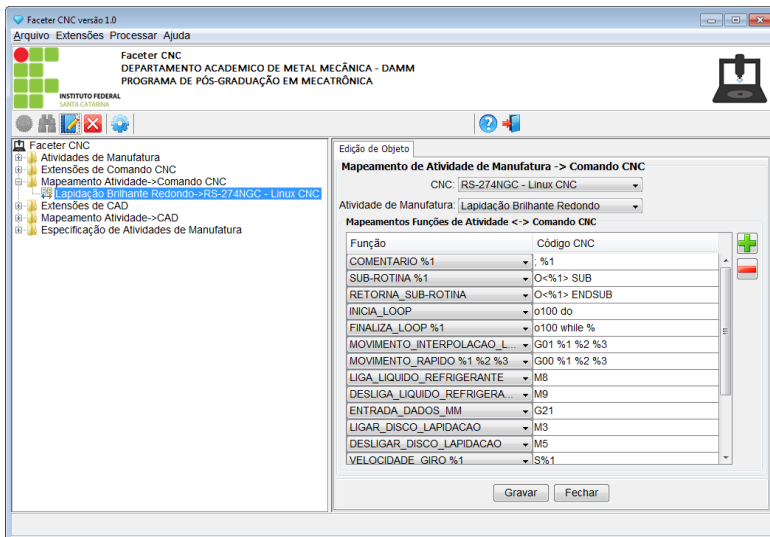
Na figura 48 pode-se ver que apenas um nome para o comando CNC e uma descrição são o suficiente para realizar a inclusão de um novo comando CNC ao software e possibilitar, a partir dele, a criação de um mapeamento entre os comandos em linguagem não-dedicada e os códigos CNC específicos da plataforma inserida no software.

Depois de incluída a nova extensão CNC o novo objeto de comando CNC foi incluído ao sistema (figura 49), nomeadamente foi incluído o Linux CNC aos objetos do software.



**Figura 49: Novo comando CNC incluído no Pós-Processador (Elaborado pelo autor)**

O próximo passo é fazer o mapeamento dos comandos CNC. A realização deste requisito é mostrada na figura 50.



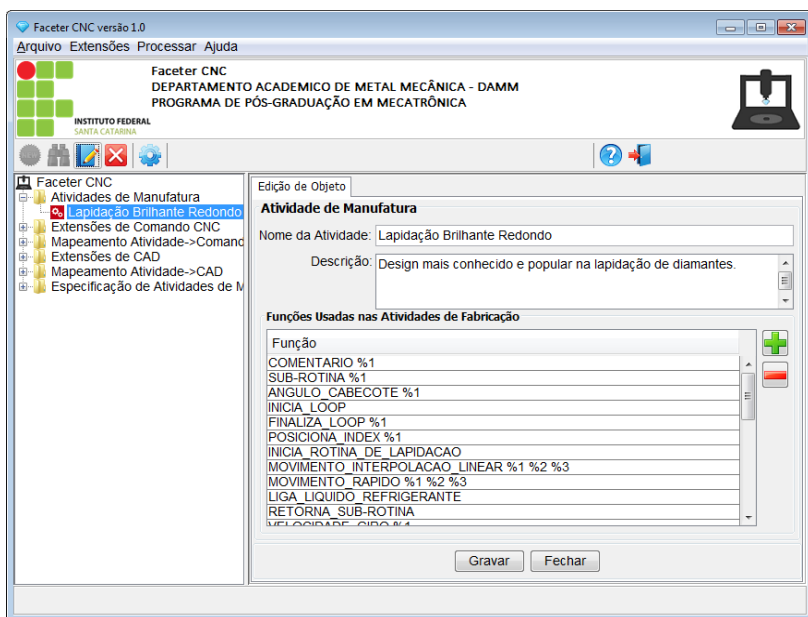
**Figura 50: Mapeamento de Atividades de Manufatura com Comandos CNC (Elaborado pelo autor)**



### 5.2.3. Linguagem de Descrição da Atividade de Manufatura

Para cumprir o requisito funcional número 2, obrigatório, mostrado na tabela 3, que exige que o sistema tenha uma linguagem de descrição do processo de fabricação de gemas - processo este que está descrito na seção 2.3 – foram criados alguns comandos em linguagem não-dedicada fixos, ou seja, comandos pré-estabelecidos, e outros comandos são inseridos pelo operador do sistema. Para estabelecer uma linguagem de descrição da atividade de manufatura dois passos são necessários, a saber: criar e manter novas atividades de manufatura, e criar e manter a descrição de atividades de manufatura.

Para criar e manter novas atividades de manufatura foi criada a interface com o usuário mostrada na figura 51. Nela há uma tabela de funções usadas na fabricação de gemas. Estas funções são os comandos em linguagem não-dedicada mencionados anteriormente. Note-se que estes comandos são criados em tempo de execução do software pelo operador CNC, ator que executa este caso de uso, conforme pode ser visto no apêndice 2, caso de uso número 1 (UC01).

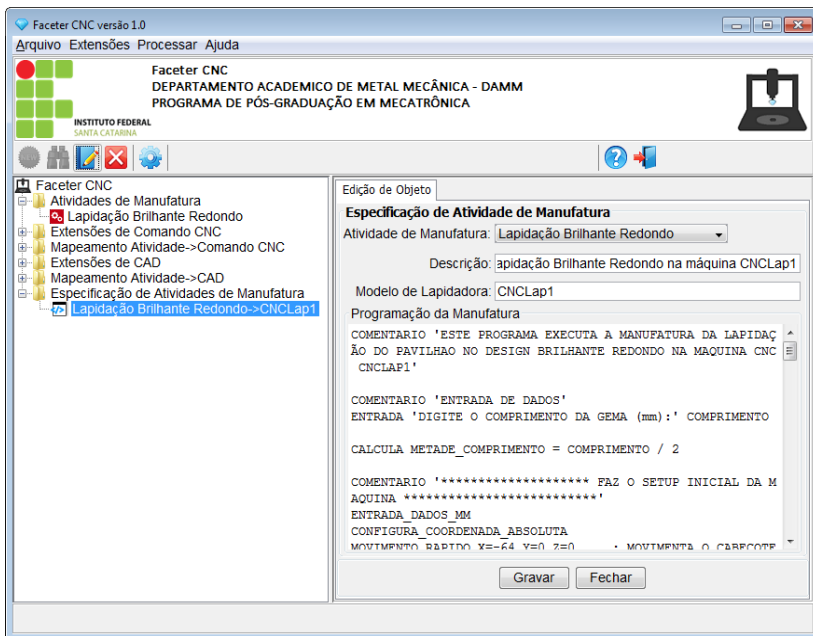


**Figura 51: Criação e manutenção de atividades de manufatura (Elaborado pelo autor)**

A figura 51 mostra a possibilidade, através do botão “+”, em verde, de inserir novos comandos de acordo com a necessidade do operador CNC.

A descrição das atividades de manufatura é o último estágio que envolve o operador CNC. Este ator, como visto antes faz o “*setup*” do software de lapidação.

A figura 52 ilustra como é a realização do requisito funcional número 2. Nesta realização o software permitirá a criação da descrição da atividade de manufatura para uma máquina específica. Neste caso, o modelo de lapidadora usado é o CNCLap1, que foi a nomenclatura usada para a máquina cuja geometria é descrita na seção anterior, 5.1.



**Figura 52: Especificação das atividades de manufatura (Elaborado pelo autor)**

O sistema também vincula, como pode ser notado na figura, a vinculação da atividade de manufatura a uma atividade de manufatura pré catalogada conforme mostrado na figura 50.

A seguir, a listagem 1 (a qual reflete a programação da manufatura mostrada na figura 52), mostra os comandos criados na

linguagem de descrição das atividades de manufatura. Nela é mostrada a descrição necessária para executar a lapidação mais popular, descrita na seção 5.2.3, a “lapidação brilhante redondo”, popularmente conhecida como lapidação brilhante, ou, simplesmente, brilhante.

```
1: COMENTARIO 'ESTE PROGRAMA EXECUTA A MANUFATURA DA LAPIDAÇÃO
DO PAVILHAO NO DESIGN BRILHANTE REDONDO NA MAQUINA CNC CNCLAP1'

2: COMENTARIO 'ENTRADA DE DADOS'
3: ENTRADA 'DIGITE O COMPRIMENTO DA GEMA (mm):' COMPRIMENTO
4:
5: CALCULA METADE_COMPRIMENTO = COMPRIMENTO / 2
6:
7: COMENTARIO '*** FAZ O SETUP INICIAL DA MAQUINA ***'
9: ENTRADA_DADOS_MM
10: CONFIGURA_COORDENADA_ABSOLUTA
11: MOVIMENTO_RAPIDO X=-64 Y=0 Z=0 ;MOVIMENTA O CABEÇOTE EM
DIREÇÃO AO DISCO DE LAPIDAÇÃO
12: LIGA_LIQUIDO_REFRIGERANTE
13: LIGAR_DISCO_LAPIDACAO
14: VELOCIDADE_GIRO 2400 ;VELOCIDADE MINIMA PARA LAPIDAÇÃO DE
GEMAS (2400-3000 RPMS)
15: VELOCIDADE_AVANCO 60 ;VELOCIDADE DE AVANÇO DE 1MM/S OU
60MM/MIN
16:
17: COMENTARIO '*** EXECUTA A PRIMEIRA CAMADA DE FACETAS ***'
18: ANGULO_CABECOTE %ANGULO_CAB1% ;COLOCA O CABEÇOTE NO ANGULO
PASSADO PELO CAD PARA A PRIMEIRA CAMADA DE FACETAS
19: CALCULA REMOCAO_Z1 = METADE_COMPRIMENTO -
%DISTANCIA_FACETA_CENTRO1% ;CALCULA A QUANTIDADE PARA REMOVER
20: CALCULA POSICAO_FACETAMENTO_Z = 85 + %REMOCAO_Z1%
21:
22: COMENTARIO 'CAMADA = 1, INDEX = 1'
23: MOVIMENTO_RAPIDO Z=85 ; POSICIONA A PEDRA NO LIMIAR DO DISCO
24: CALCULA INDEX_ATUAL = %INDEX[1,1]% * 3.75 ;EM UMA MAQUINA COM
INDEX 96 CADA INDEX SAO 3.75 GRAUS
25: POSICIONA_INDEX %INDEX_ATUAL% ;POSICIONA O INDEX O CAMADA =
1 INDEX = 1
26: MOVIMENTO_INTERPOLACAO_LINEAR Z=%POSICAO_FACETAMENTO_Z%
;EFETUA A CRIACAO DA FACETA
27:
28: COMENTARIO 'CAMADA = 1, INDEX = 2'
29: MOVIMENTO_RAPIDO Z=85 ;POSICIONA A PEDRA NO LIMIAR DO DISCO
30: CALCULA INDEX_ATUAL = %INDEX[1,2]% * 3.75 ;EM UMA MAQUINA COM
INDEX 96 CADA INDEX SAO 3.75 GRAUS
31: POSICIONA_INDEX %INDEX_ATUAL% ;POSICIONA O INDEX O CAMADA =
1 INDEX = 2
32: MOVIMENTO_INTERPOLACAO_LINEAR Z=%POSICAO_FACETAMENTO_Z%
;EFETUA A CRIACAO DA FACETA
32:
33: COMENTARIO 'CAMADA = 1, INDEX = 3'
34: MOVIMENTO_RAPIDO Z=85 ;POSICIONA A PEDRA NO LIMIAR DO DISCO
```

```

35: CALCULA INDEX_ATUAL = %INDEX[1,3]% * 3.75 ;EM UMA MAQUINA COM
INDEX 96 CADA INDEX SAO 3.75 GRAUS
36: POSICIONA_INDEX %INDEX_ATUAL% ;POSICIONA O INDEX O CAMADA =
1 INDEX = 3
37: MOVIMENTO_INTERPOLACAO_LINEAR Z=%POSICAO_FACETAMENTO_Z%
;EFETUA A CRIACAO DA FACETA
38:
...
273: FIM_PROGRAMA

```

### Listagem 1: Excerto da descrição da manufatura do pavilhão do design de lapidação brilhante (RB)

Na listagem 1 é possível identificar comandos fixos e comandos criados pelo usuário, da forma vista anteriormente nesta seção, precisamente na definição da atividade de manufatura (ver figura 50).

Os comandos fixos mostrados são: COMENTARIO, CALCULA e ENTRADA, que tem objetivo de comentar ou documentar o código, calcular expressões matemáticas e solicitar a entrada de dados do usuário, respectivamente.

Por outro lado, os comandos usados na definição da atividade de manufatura, criados em tempo de execução, são mapeados diretamente para comandos em linguagem CNC, através do mapeamento de atividades de manufatura com os comandos CNC, conforme visto na figura 50. No processo de manufatura para geração de programas CNC para lapidação de gemas, devido à simplicidade relativa da lapidação de gemas, levando em conta que se trata somente de fazer linhas retas em uma operação de desbaste de 2 ½ eixos, descreve-se trajetórias simples baseadas em posicionamento da gema e posterior avanço em um único grau de liberdade para a realização da ação de lapidação ou polimento.

A seguir será mostrado como executar o último processo do software pós-processador, a geração dos programas CNC, tal processo é feito tendo como base o arquivo CAD somado da descrição de manufatura mostrada.

#### 5.2.4. Geração de Programas CNC a Partir de Arquivos CAD

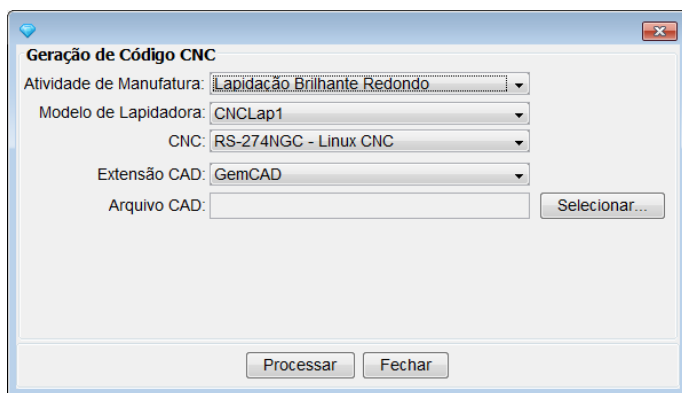
O objetivo principal do software e, por consequência deste trabalho, é alcançado: gerar código CNC. Isto é o requisito visto na seção 1.1, trata-se da meta a ser alcançada. É a realização do requisito número 1, especificado na tabela 3, o qual encontra-se modelado no caso de uso número 6 do apêndice 2.

Este caso de uso é executado pelo ator chamado “*designer* de

gemas”, conforme mostra o apêndice 2, e integra-se com a máquina lapidadora CNC, ou como visto no apêndice 2, com uma virtualização computadorizada desta máquina para que se possa simular os movimentos em computador antes da execução real.

A geração do código CNC no software pós-processador é acessada através do menu Processar->Gerar Código CNC, ou através da tecla de atalho CTRL+G, o que acionará a interface de geração de código CNC, como mostrado na figura 53.

Este código pode agora ser utilizado na máquina para realizar uma operação real de fabricação de gema, desde que respeitadas as operações de preparação da peça e da máquina.



**Figura 53: Interface do usuário de geração de código CNC (Elaborado pelo autor)**

Um excerto do código gerado após a execução é mostrado na listagem 2, a seguir:

```

...
; '***** FAZ O SETUP INICIAL DA MAQUINA *****'
G21
G90
G00 X=-64 Y=0 Z=0
M8
M3
S2400
F60

; '***** EXECUTA A PRIMEIRA CAMADA DE FACETAS *****'
G00 -90.00
#3 = #2 - 1.00000
#4 = 85 + #3

```

```
; 'CAMADA = 1, INDEX = 1'  
G00 Z=85  
#5 = 93 * 3.75  
G00 #5  
G01 Z=#4  
  
; 'CAMADA = 1, INDEX = 2'  
G00 Z=85  
#5 = 87 * 3.75  
G00 #5  
G01 Z=#4  
...
```

**Listagem 2: Excerto do código CNC gerado pelo software Pós-Processador**

# Capítulo 6

## CONCLUSÃO

---

Este capítulo é dividido em duas seções. A primeira destaca a forma com que os objetivos do trabalho foram resolvidos. A segunda traz perspectivas de desenvolvimento futuro do software pós-processador e para a área de manufatura de gemas com tecnologia CNC.

### 6.1 Conclusão

O objetivo principal de interpretar arquivos software CAD da área de lapidação de gemas e gerar programa específico para cada máquina de lapidação CNC foi cumprido. O código mostrado na listagem 2 no capítulo anterior é o resultado da execução da geração de programa CNC baseado no software CAD GemCad®, para gerar a lapidação Brilhante Redondo. Foi feita uma análise de cada comando em linguagem não-dedicada (Figura 50) relacionando ao código gerado e este foi gerado em sua totalidade, com a possibilidade de ajustes com edições na descrição do processo de manufatura.

O software pós-processador arquitetado e desenvolvido demonstrou-se ser uma ferramenta extremamente flexível e extensível devido às seguintes capacidades: inserir módulos (“*plug-ins*”) de entrada, usar uma linguagem não-dedicada para descrever as atividades ou processos de manufatura, a possibilidade de uso de novos modelos de máquina de lapidar, e adaptação de novos comandos CNC de saída. Este fato pode ser visto na inclusão de novas extensões CAD (Figura 45) e novos comandos CNC (Figura 49). Isto comprova o cumprimento dos objetivos de extensibilidade da arquitetura (Apêndice 18) e os múltiplos formatos de entrada e saída.

A descrição das atividades ou processos de manufatura na forma que foi elaborada no projeto permite: a entrada de variáveis que serão usadas no processo de lapidação CNC, inserir dados relativos a trajetória de lapidação com as restrições da geometria da máquina. Através do mapeamento das funções em linguagem não-dedicada (Figura 50) com comandos CNC, fornecem um sistema neutro em termos de linguagem para desenvolver os processos de lapidação. Isto confere ao operador do software mais um ponto onde há total poder de criação para a construção de modelos de lapidação, o que cumpre com o objetivo de

desenvolvimento de uma linguagem não-dedicada para descrição do processo de lapidação.

## 6.2 Perspectivas Futuras

Após ter a integração da manufatura de lapidação de gemas consolidada em uma etapa importante com o desenvolvimento do software pós-processador, o processo proposto para lapidação CNC, mostrado na figura 6, foi até o nível de “Facetador” ilustrado, que seria o software pós-processador. A perspectiva mais importante sob a ótica do autor, para o momento, seria a finalização da cadeia de manufatura para lapidação de gemas que passaria pela integração de um comando CNC e a criação efetiva de uma máquina CNC de lapidação de gemas para executar os programas CNC e poder realizar concretamente a lapidação CNC aproveitando da rapidez e flexibilidade fornecida pelo software.

Como vimos durante este trabalho, o software pós-processador desenvolvido permite a adequação de geometrias diferentes de máquina de lapidação CNC, portanto há uma boa flexibilidade para novas ideias de modelos de máquina CNC que podem ter melhorias se comparadas com a máquina *Faceting Head* mostrada na figura 43. Isto demonstra que o software é flexível em suportar diferentes modelos de máquina CNC que podem ser explorados em outros projetos futuros.

Com relação ao software pós-processador e o modelo de desenvolvimento do projeto que seguiu neste trabalho, usando as atividades de manufatura (ou processos de manufatura), que são definidos usando uma linguagem não-dedicada (ver figura 52), podem ficar mais ágeis se forem gerados os programas CNC diretamente a partir do CAD, assim como é feito nos softwares fatiadores (*slicers*, ver seção 2.1.1), isto teria um efeito positivo de agilidade do trabalho, afinal uma etapa de escrever código em linguagem não-dedicada seria extinta. Entretanto tem um efeito negativo de perda de flexibilidade e controle do código gerado e a inserção de mais desenvolvimento do software pós processador.

A geração do código CNC a partir do modelo CAD se daria através de um grande número de linhas de código, ou seja, programação de software. Isto por que teria que se colocar a lógica de geração das trajetórias dentro do software e a variável de qual máquina de saída também seria inserida nesta lógica de programação. Seriam dispendidas muitas horas de desenvolvimento para chegar neste patamar e levaria a uma perda de flexibilidade.

Além da construção da máquina de lapidação, a geração de programa CNC diretamente do arquivo CAD, outros itens que poderiam



ser explorados são a implementação de novas extensões de entrada e saída e o desenvolvimento de requisitos explorados na seção 4.1 e que foram marcados na Tabela 3, cuja categoria é “Desejado”. Estes requisitos não têm seu desenvolvimento obrigatório no escopo do projeto devido às limitações de tempo que houve para conclusão do mesmo. Como exemplos poderiam ser desenvolvidos: uma extensão de entrada para suportar o software CAD SolidWorks®, que é muito popular na atualidade; e/ou aprimorar o desenvolvimento de novas atividades de manufatura através de componentes visuais; e/ou criar um ambiente de virtualização para simular a cinemática da máquina de lapidar. Seria muito ousado ter assumido o compromisso de desenvolver estas funcionalidades com os recursos disponibilizados neste projeto somado ao tempo de execução disponível. Sendo assim, ao final ainda há a possibilidade de criar-se muitas funcionalidades complementares ao projeto do pós-processador e até criar projetos de mecatrônica que possam executar programas CNC gerados por ele.

# REFERÊNCIAS

---

ADIVAREKAR, M.; LIOU, F. **Developing a General Postprocessor for Multi-Axis CNC Milling Centers**. Missouri University of Science and Technology. 2012.

ALLABOUTGEMSTONES. Disponível em: <[www.allaboutgemstones.com](http://www.allaboutgemstones.com)> Acesso em 2014.

AZEVEDO, Américo L. -Mundo CNC, Disponível em <<http://www.mundocnc.com.br/basic3.php>>. Acesso em: 10 de janeiro de 2016

BOOCH, Grady. **UML: guia do usuário**. 2. ed. Rio de Janeiro: Elsevier, 2006.

BLODGETT, T.; GEURTS, R.; LUCAS, A.; PAY, D.; REINITZ, I.; SHIGLEY, J.; YANTZER, K.; ZINC, C. 2009. Finish, Culet Size and Girdle Thickness: Categories of the GIA Diamond Cut Grading System. Disponível em: <[http://www.diamondcut.gia.edu/pdfs/booklet\\_finish\\_culet\\_girdle.pdf](http://www.diamondcut.gia.edu/pdfs/booklet_finish_culet_girdle.pdf)>. Acesso em: 2015.

CHAMBERLAIN, M.; KULKARNI, R.; WILLIAMS, C. STL Viewer. Disponível em <<http://srl.gatech.edu/Members/cwilliams/classes/CBW.ME6104.Report.pdf>>, Acesso em 2014.

CHEN, F. X.; WENGMAN, E. J. **Foundations of 3D Graphics Programming**. 1 ed., London:Springer. 2006.

CARVALHO, F. O. G. L. A. **Utilização de sistemas CAM para programação de máquinas-ferramenta**, Dissertação de Mestrado, Faculdade de Engenharia da Universidade do Porto. Porto. 2013.

Direct3D tutorial sample. Disponível em <<https://msdn.microsoft.com/pt-br/library/cc518041.aspx>> Acesso em 2014.

FAVACHO, M. Quartzo. In: CASTAÑEDA, C. et al. (org.) *Gemas de Minas Gerais*. Belo Horizonte, Soc. Brasil. Geologia. p. 220-233. 2001.

FRAUNHOFER. Optimal Dissection of the Raw Material. Disponível em:

<<http://www.itwm.fraunhofer.de/en/departments/optimization/optimization-in-virtual-engineering/optimal-dissection-of-the-raw-material.html>>. Acesso em 2015.

GAMMA, Erich. PRICE, Roberto Tom. **Padrões de projeto: soluções reutilizáveis de software orientado a objetos**. Porto Alegre: Bookman, 2000.

GESSER, J. F.; **Proposta de um Sistema de Baixo Custo para Fresamento em Cinco Eixos**, Dissertação de Mestrado, Instituto Federal de Santa Catarina. Florianópolis. 2012.

GEURTS, R. H.; REINITZ, I. M.; BLODGETT, T.; GILBERTSON, A. M. GIA's **Symmetry Grading Boundaries for Round Brilliant Cut Diamond**. *Gems & Gemology*. Vol. 47, No. 4, pp. 286-295. Gemological Institute of America. 2011.

GONÇALVES, M. A. F. **Geração de Programas CNC Através da Implementação de Funções Direcionadas às Características do Processo Produtivo**, Tese de Doutorado, Universidade Federal do Rio Grande do Sul, UFRGS, Porto Alegre, 2013.

HARTMANN, L.A., **Geração dos Diagramas de Execução de Modelos de Lapidação de Gemas**. Revista Brasileira de Computação Aplicada. Vol. 2, No. 1.

IBGM. Exportação Brasileira do Setor de Gemas e Metais Preciosos Ano 2013. Disponível em: <[http://www.ibgm.com.br/pdf/estatisticas\\_estaduais\\_geral.pdf](http://www.ibgm.com.br/pdf/estatisticas_estaduais_geral.pdf)>. Acesso em: 26 fev 2014.

ISO 6983. Disponível em:

<[http://www.iso.org/iso/home/store/catalogue\\_tc/catalogue\\_detail.ht](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.ht)

[m?csnumber=34608](#)> Acesso em 2014

JACOBSON, I. **Object-Oriented Software Engineering**. Harlow: Addison Wesley Longman, Inc., 1993.

KLIPPEL, A. F. (2007) **Implementação da gestão enxuta em empresas de mineração a partir de um modelo de gestão integrada: uma perspectiva de sinergia entre a engenharia de minas e a engenharia de produção**. Tese de Doutorado. Universidade Federal do Rio Grande do Sul.

KLM TECHNOLOGY, INC. 2013. Disponível em: <<http://www.klmtechgroup.com/>> Acesso em 2015.

LARMAN, Craig. **Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3. ed. Porto Alegre: Bookman, 2000.

LIMA, D. O. **Geração da Trajetória no Fresamento CNC Utilizando Diagrama de Voronoi**. Tese de Mestrado. Programa de Pós-Graduação de Engenharia UFRJ. Rio de Janeiro. 2004.

MUNDO CNC. Conceitos Básicos. Mundo CNC, 2008. Disponível em: <<http://www.mundocnc.com.br/basic4.php>>. Acesso em: 29 Jul. 2014.

NADUR, V.A. **A lapidação de gemas no panorama brasileiro**. Dissertação de Mestrado. Instituto de Geociências. Universidade de São Paulo. USP. Programa de Pós-Graduação em Mineralogia e Petrologia. São Paulo. p. 102-103. 2009.

NARAYAN, L.K., RAO, K.M., SACAR, M.M.M. **Computer Aided Design and Manufacturing**. New Delhi: Prentice Hall of India. pp. 286-334, 2008.

NEVADA. A little on faceting a gemstone: A step by step process. Disponível em: <[http://nevada-outback-gems.com/faceting\\_example/FACETING.HTM](http://nevada-outback-gems.com/faceting_example/FACETING.HTM)>. Acesso em 04 de Agosto de 2015.

RIBEIRO, H.M.D . **Fatores relevantes no desempenho brasileiro no mercado internacional de pedras preciosas.** Departamento de Economia. Universidade Federal de Viçosa. Programa de Pós-Graduação em Economia. Viçosa. 2008.

ROMI, Indústrias Romi S/A, “Manual de Programação e Operação Linha Romi D”.Disponível em <[http://www.romi.com.br/fileadmin/Editores/MF/MANUAL\\_PROGR\\_OPERACAO\\_WORLDSKILLS/LINHA\\_D\\_SIEMENS\\_PORTUGUES\\_Progr\\_Operacao.pdf](http://www.romi.com.br/fileadmin/Editores/MF/MANUAL_PROGR_OPERACAO_WORLDSKILLS/LINHA_D_SIEMENS_PORTUGUES_Progr_Operacao.pdf)>. Acesso em: 13/10/2014. p.p. 11-13.

SCHUMANN, Walter. **Gemas do mundo.** 9. ed. São Paulo: Disal, 2006.

SELMAN, D. **Java 3D Programming.** 1 ed., Manning, Greenwich, 2002.

SMEDA. Pre-Feasibility Study - Gemstone Lapidary. Disponível em: <[http://www.smeda.org/index.php?option=com\\_phocadownload&view=category&download=926:gems-stone-lapidary&id=105:minerals](http://www.smeda.org/index.php?option=com_phocadownload&view=category&download=926:gems-stone-lapidary&id=105:minerals)>. Acesso em: 10 dez 2013.

SOMMERVILLE, I. **Software Engineering.** 9 ed., Addison-Wesley, Boston, 2011.

SOLANKI MACHANIC WORKS INC, 2013. Disponível em: <<http://www.solankimechanicworks.com/contact-us.htm>> Acesso em 2014.

SOUZA, A. F.; ULBRICH, C. B. L. **Engenharia Integrada por Computador e Sistemas CAD/CAM/CNC.** 2. ed., Artlibre, São Paulo, 2013.

STL. In: WIKIPÉDIA: a enciclopédia livre. Wikimedia, 2015. Disponível em: <[http://en.wikipedia.org/wiki/STL\\_\(file\\_format\)](http://en.wikipedia.org/wiki/STL_(file_format))>. Acesso em: 20 de mai. 2015.

STRICKLAND, R. W. **GemCad® para Windows,** versão 1.0. Guia

do Usuário. 2002.

SUH, S.; KANG S.; CHUNG D.; STROUD I. **Theory and Design of CNC Systems**, 1. ed., Springer, London, 2008.

SUN. Sun Microsystems, **Object-Oriented Analysis and Design Using UML OO-226**. Revision C. Broomfield, 2003.

ULTRATECFACET INC. Maquina de Lapidar Ultratec V5. Disponível em: <<http://www.ultratec-facet.com/V5.htm>>. Acesso em 2013.

YOON, J. Getting Started with JavaFX 3D Graphics. Disponível em: <<https://docs.oracle.com/javase/8/javafx/graphics-tutorial/javafx-3d-graphics.htm>>. Acesso em 2015.

Zanatta, A. L. et. al. **Tecnologias para o setor de gemas, joias e mineração**. Porto Alegre: IGEO/UFRGS, 2010. Cap. Tecnologia 3D Gemas: visualização de projetos virtuais de lapidação, p. 53-76.

# ANEXOS

---

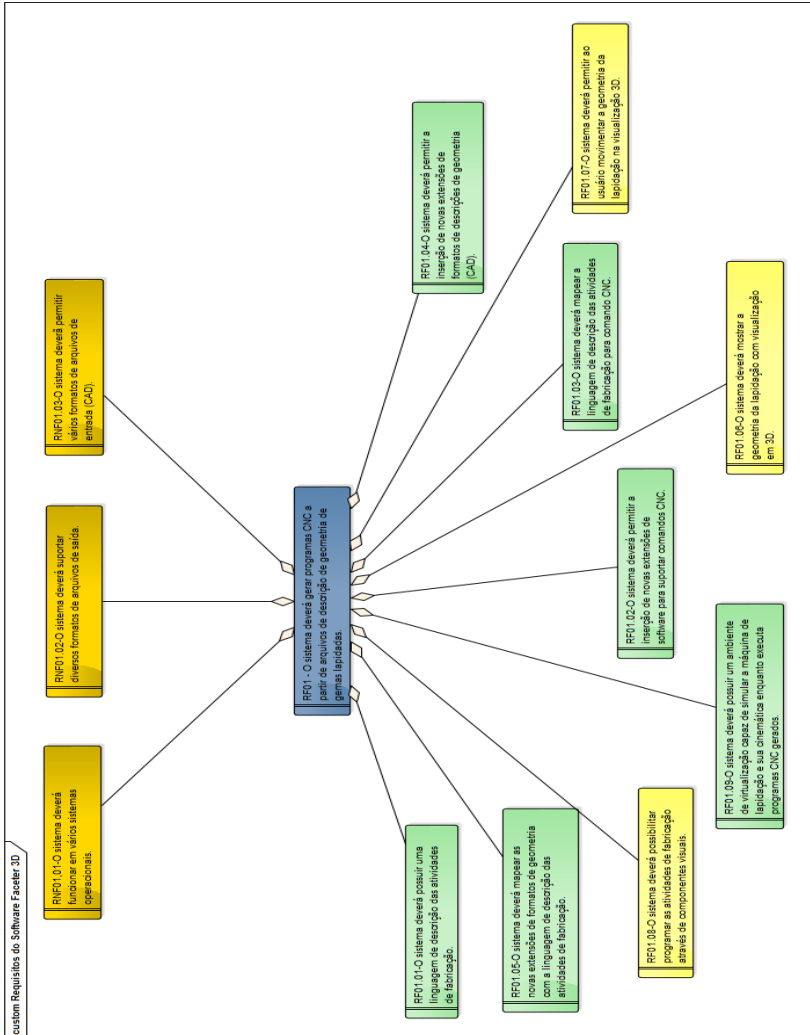
## **Anexo 1: Tabela com Algumas Funções de Miscelânea** (NARAYAN et. al., 2008).

Número	Definição
M00	Parar o Programa
M01	Similar ao anterior, exceto que o controle ignora o comando a menos que o operador tenha previamente validado o comando
M02	Fim do programa
M03	Inicia a rotação do eixo para avançar na peça
M04	Inicia a rotação do eixo sem retrair na peça
M05	Para o giro do eixo
M06	Para o giro e refrigerante e retrai a ferramenta para posição totalmente retraído.
M08	Liga o refrigerante
M09	Desliga o refrigerante

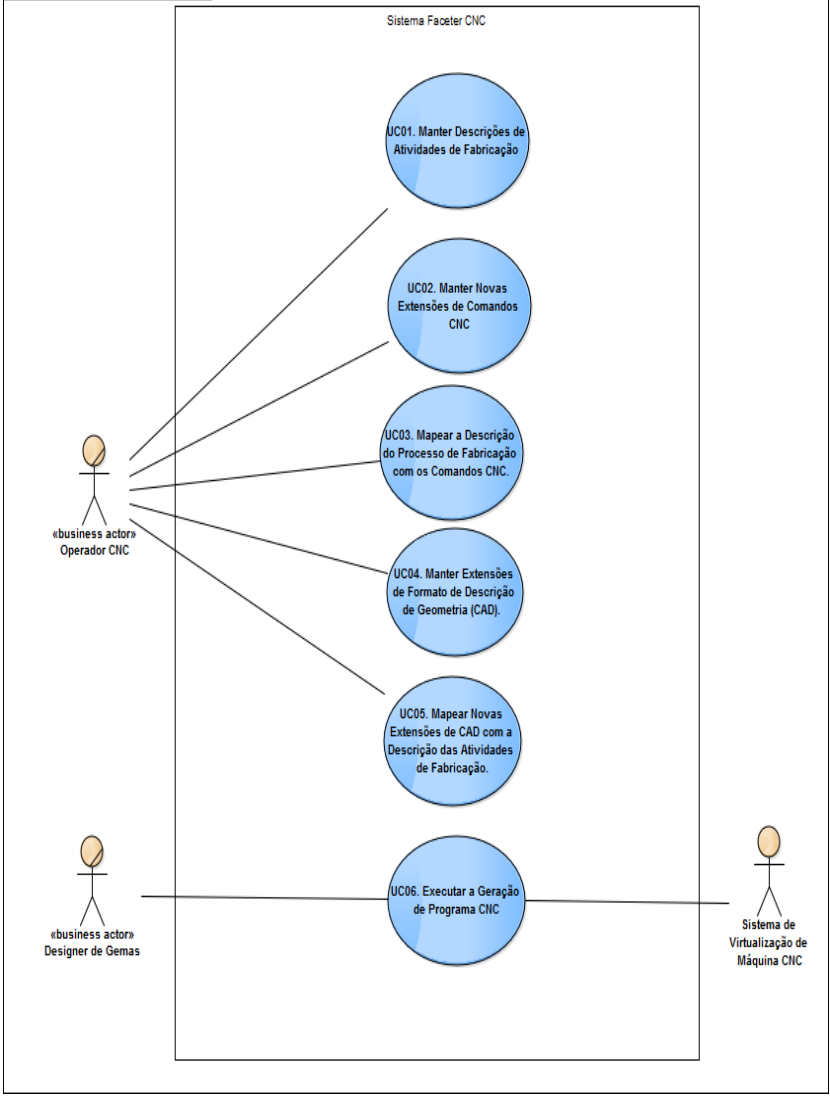
**Anexo 2: Tabela com as Principais Funções Preparatórias  
(ROMI, 2014).**

Número	Definição
G00	Posicionamento Rápido
G01	01 Interpolação Linear
G02	Interpolação Circular no Sentido Horário
G03	Interpolação Circular no Sentido Anti-Horário
G04	Tempo de permanência (Dwell)
G10	Entrada de Dados
G11	Cancela Entrada de Dados
G15	Cancela Sistema de Coordenadas Polares
G16	Ativa Sistema de Coordenadas Polares
G17	Seleciona o Plano de Trabalho “XY”
G18	Seleciona o Plano de Trabalho “XZ”
G19	Seleciona o Plano de Trabalho “YZ”
G20	Entrada de Dados em Polegadas
G21	Entrada de Dados em Milímetros
G28	Retorna o Eixo Programado para o Ponto de Referência
G40	Cancela a Compensação de Raio de Ferramenta
G41	Ativa a Compensação de Raio de Ferramenta (à esquerda do perfil)
G42	Ativa a Compensação de Raio de Ferramenta (à direita do perfil)
G43	Ativa a Compensação do Comprimento da Ferramenta (direção +)
G44	Ativa a Compensação do Comprimento da Ferramenta (direção -)
G49	Cancela a Compensação de Comprimento de Ferramenta
G50.1	Cancela Imagem de Espelho
G51.1	Ativa Imagem de Espelho
G52	Sistema de Coordenadas Local (Mudança de Ponto Zero)

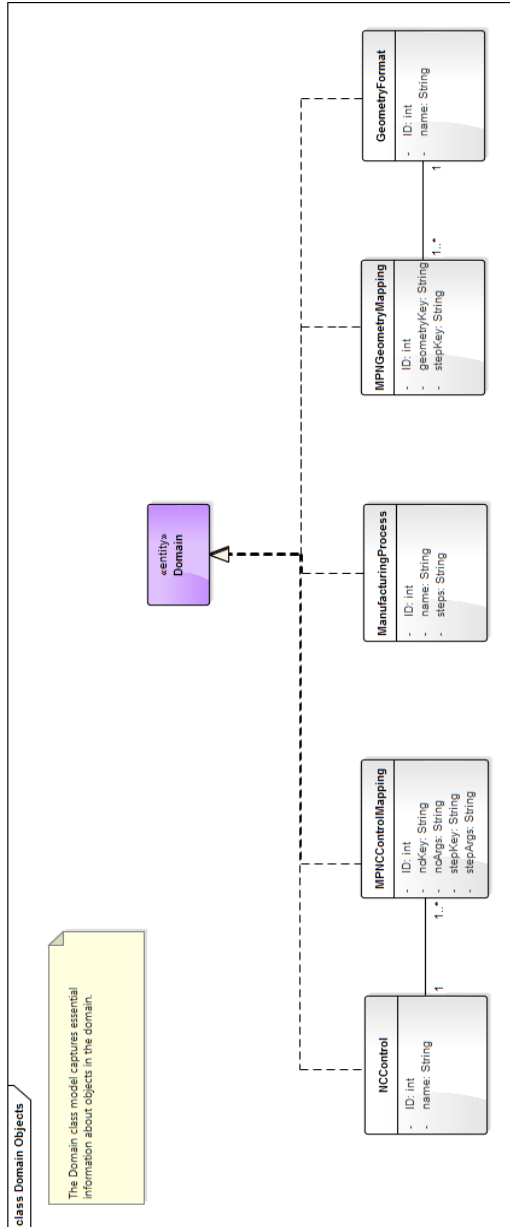




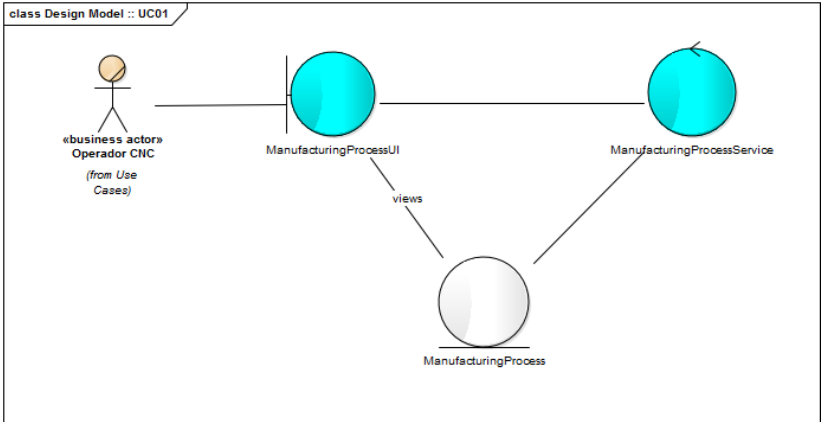
**Apêndice 3: Diagrama de Requisitos (Elaborado pelo Autor)**



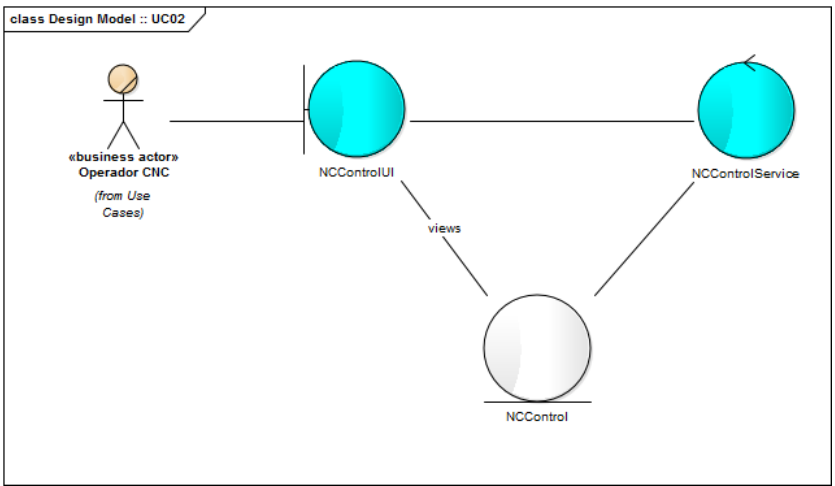
**Apêndice 4: Diagrama de Casos de Uso do Software (Elaborado pelo Autor)**



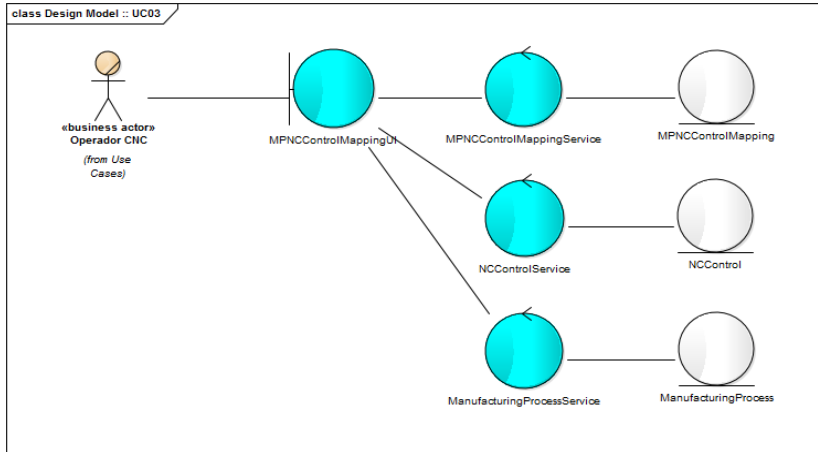
**Apêndice 5: Diagrama de Domínio (Elaborado pelo Autor)**



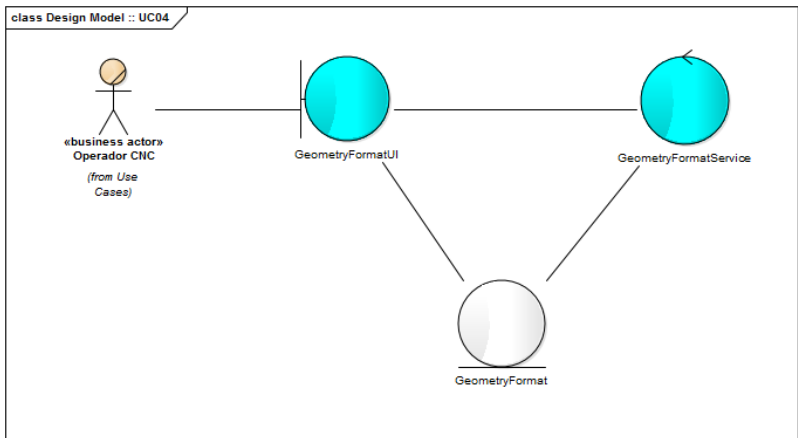
**Apêndice 6: Modelo de Projeto do Caso de Uso 01  
(Elaborado pelo autor)**



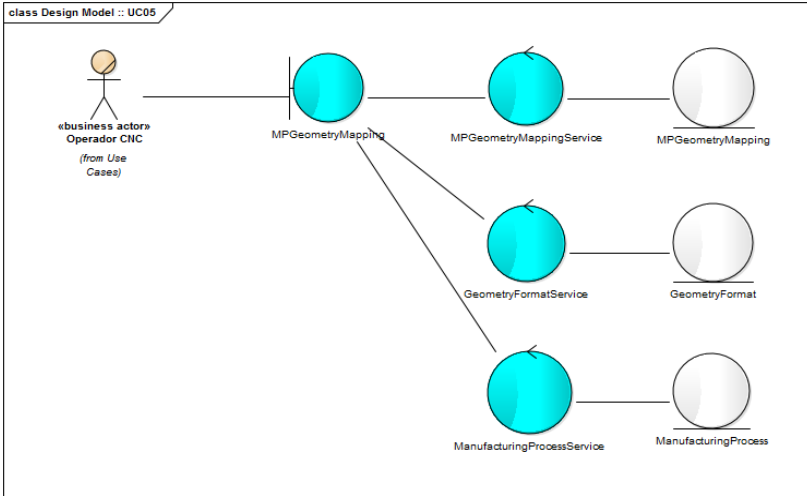
**Apêndice 7: Modelo de Projeto do Caso de Uso 02  
(Elaborado pelo autor)**



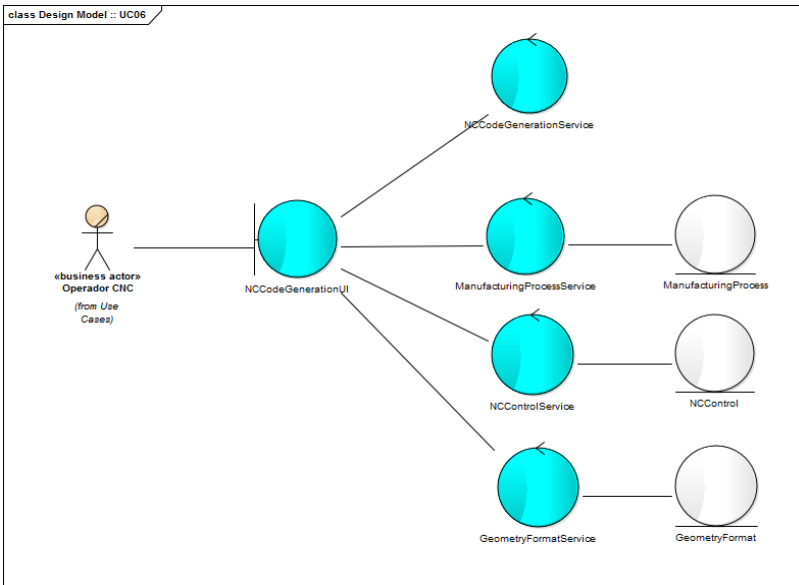
**Apêndice 8: Modelo de Projeto do Caso de Uso 03  
(Elaborado pelo autor)**



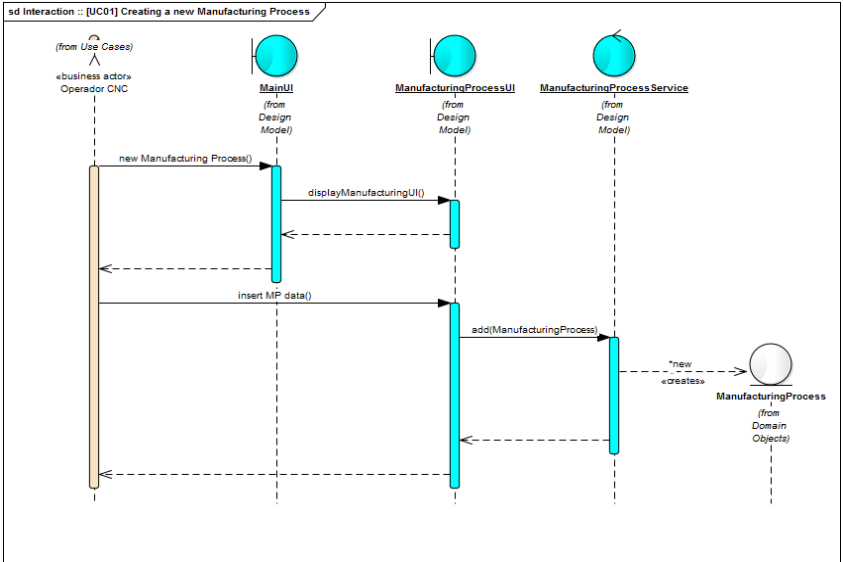
**Apêndice 9: Modelo de Projeto do Caso de Uso 04  
(Elaborado pelo autor)**



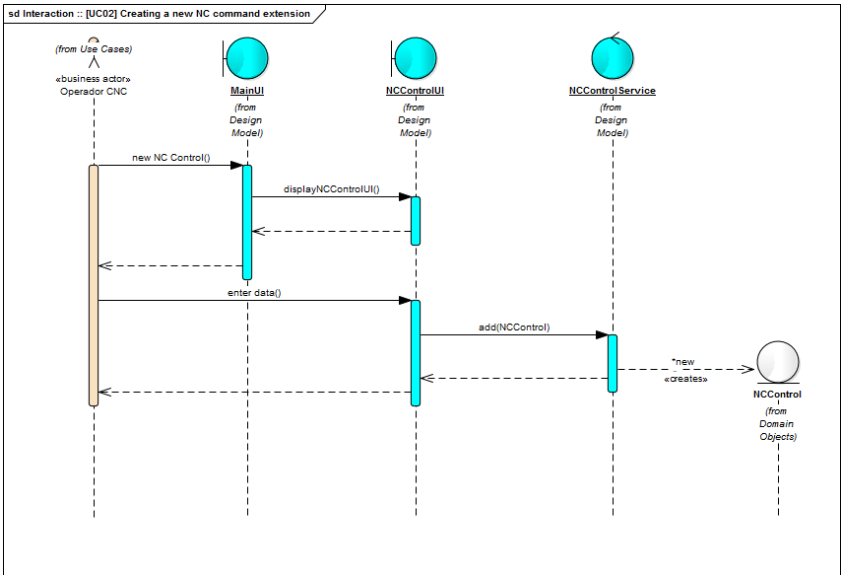
**Apêndice 10: Modelo de Projeto do Caso de Uso 05  
(Elaborado pelo autor)**



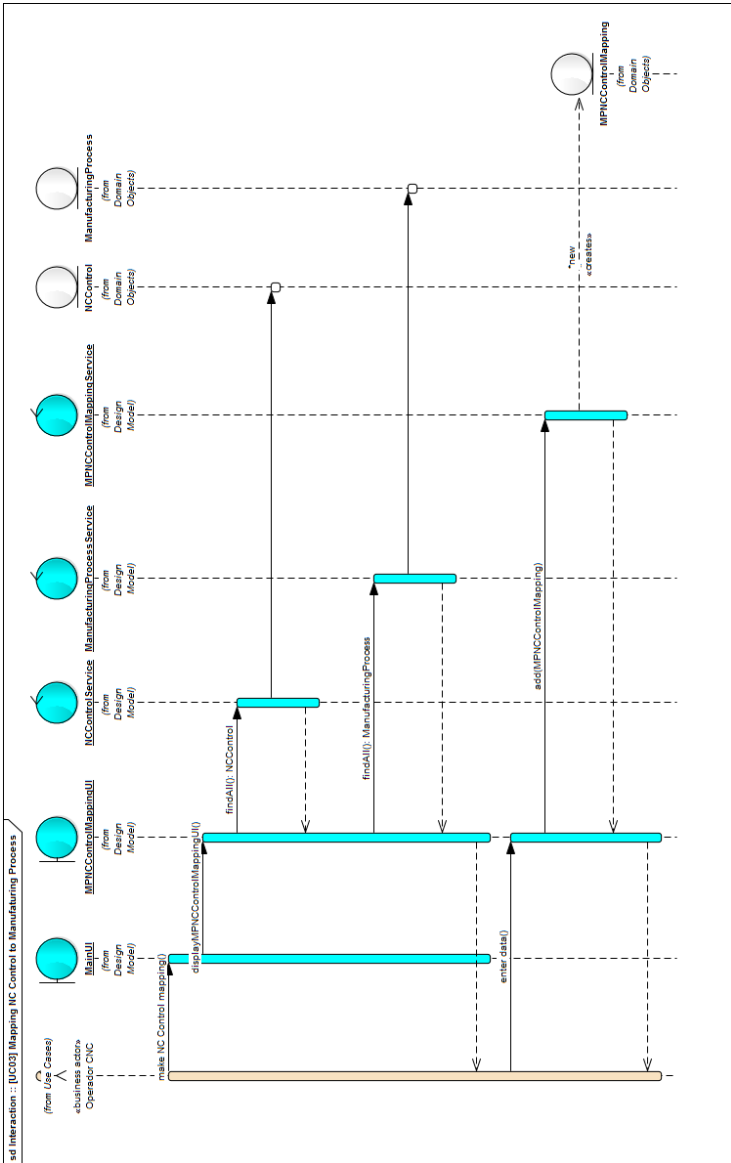
**Apêndice 11: Modelo de Projeto do Caso de Uso 06  
(Elaborado pelo autor)**



**Apêndice 12: Diagrama Sequencia cenário de inclusão de Atividade de Manufatura (Elaborado pelo autor)**

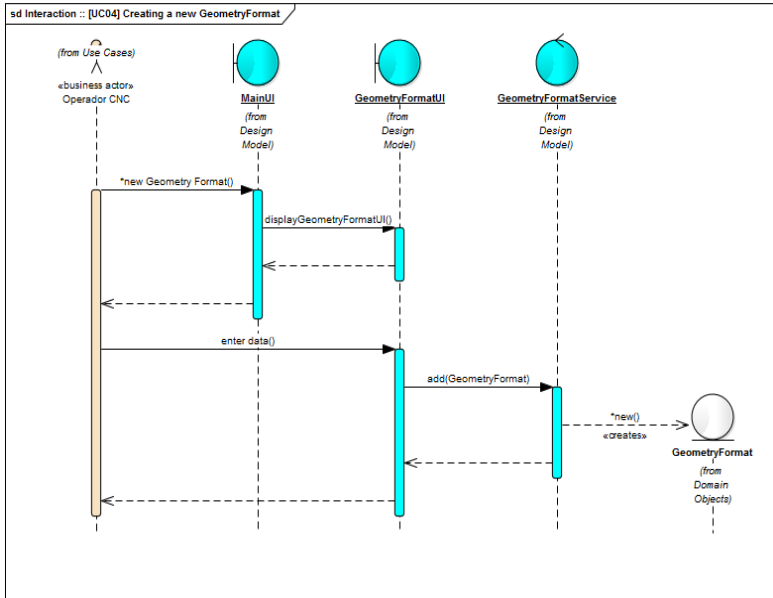


**Apêndice 13: Diagrama de Sequência cenário inclusão de extensão de comando CNC (Elaborado pelo autor)**

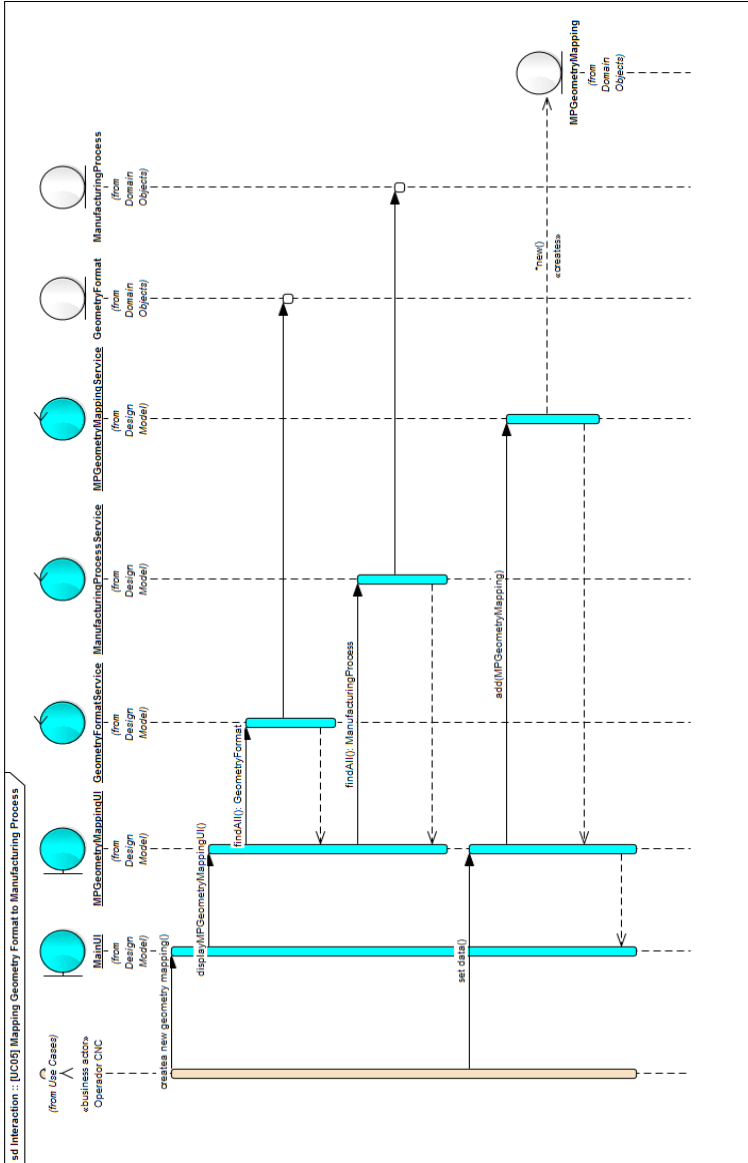


**Apêndice 14: Diagrama de Sequência cenário mapeamento de Comando NC com Atividade de Manufatura (Elaborado pelo autor)**



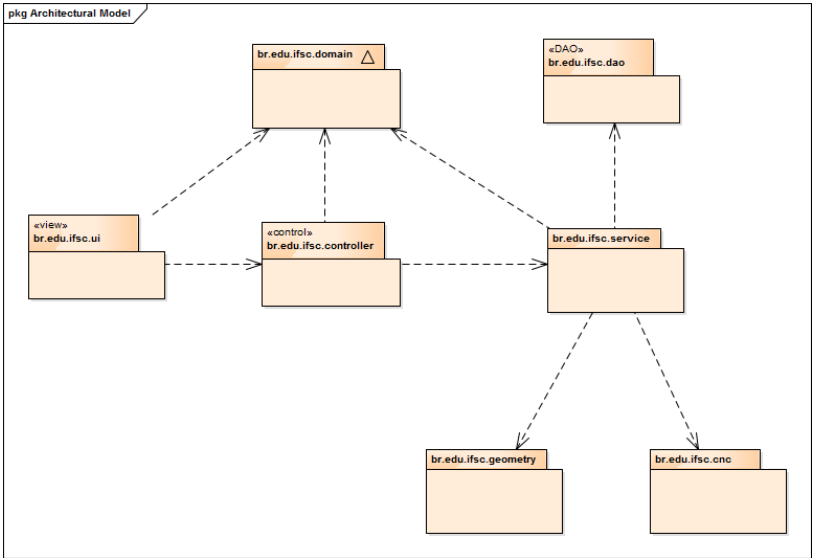


**Apêndice 15: Diagrama de Sequência cenário inclusão formato de geometria (CAD) (Elaborado pelo autor)**

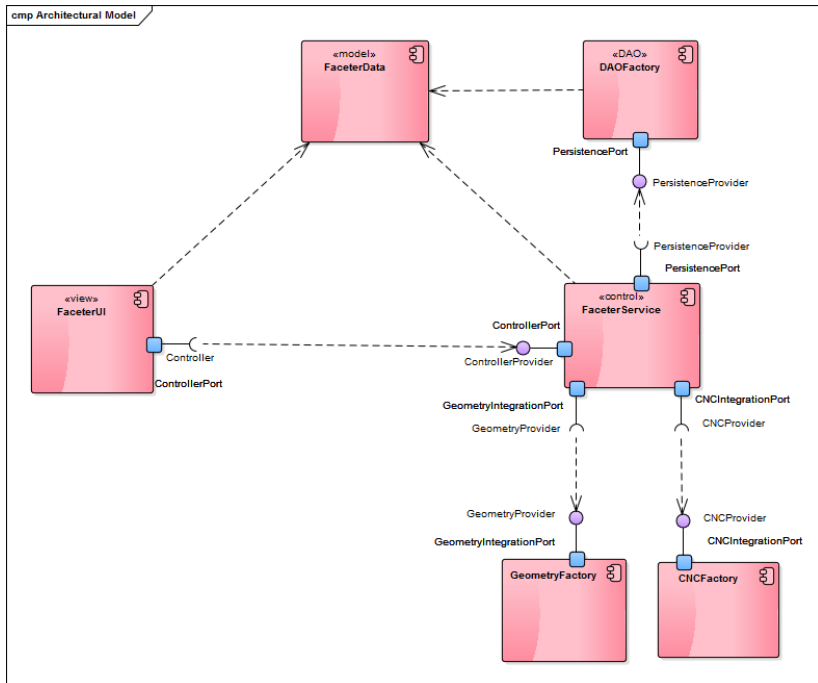


**Apêndice 16: Diagrama de sequência cenário mapeamento Geometria para Atividade de Manufatura (Elaborado pelo autor)**

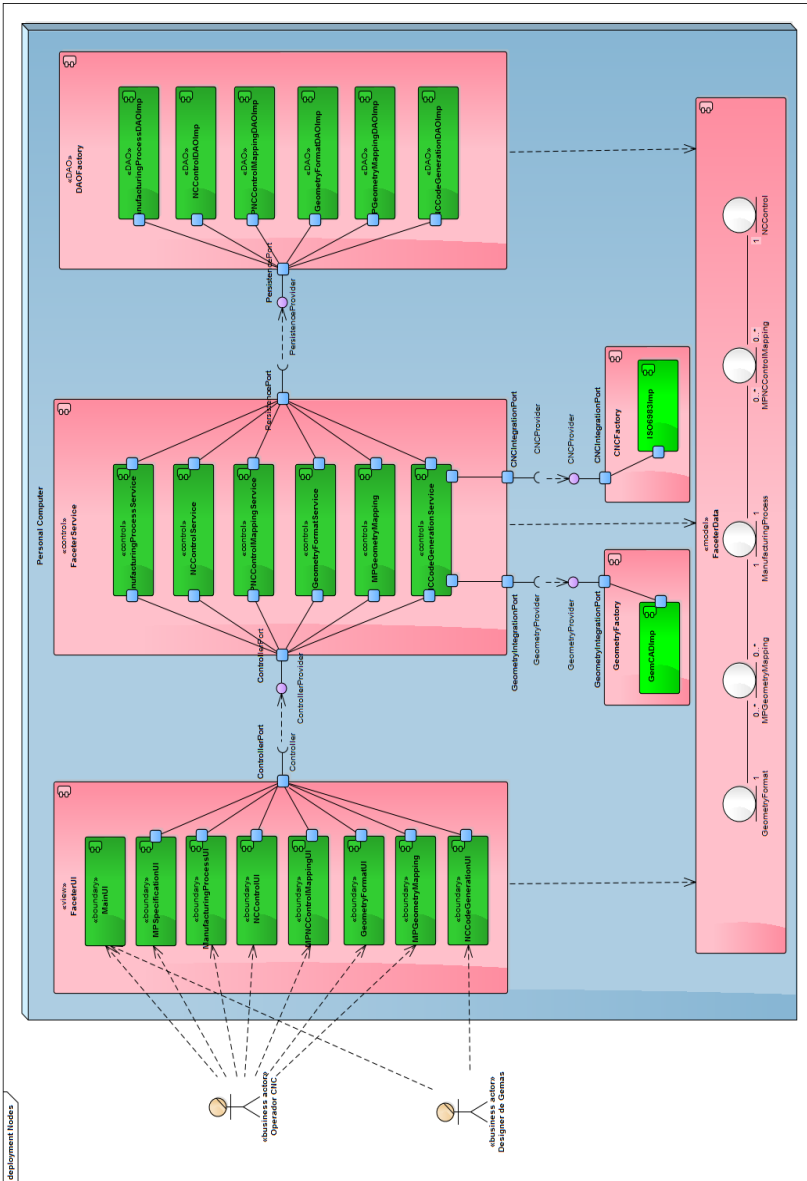




**Apêndice 18: Diagrama de Pacotes (Elaborado pelo autor)**



**Apêndice 19: Diagrama de Componentes de Alto Nível (Elaborado pelo autor)**



**Apêndice 20: Modelo de Solução Completo do Pós-Processador (Elaborado pelo autor)**