

Projeto Final TP1

Editor de Diagramas UML

Enzzo Morais de Olinda, 211042720

¹Dep. Ciência da Computação – Universidade de Brasília (UnB)
CIC0197 - Técnicas de Programação I

211042720@aluno.unb.br

1. Descrição do Problema

O Editor de Diagramas UML se trata de uma ferramenta para a elaboração de diagramas de software seguindo o padrão UML. A UML (Unified Modeling Language) se trata de um mecanismo de modelagem desenvolvida com o objetivo de estabelecer uma linguagem comum pelo qual é possível representar visualmente os diferentes aspectos da estrutura de um sistema de software. Para isso a UML apresenta uma série de convenções e notações para descrever o sistema de maneira simples, facilitando a compreensão do software pelos desenvolvedores e por qualquer outro envolvido no projeto do sistema. Dessa forma, a UML fornece uma poderosa ferramenta para a especificação, elaboração e construção de softwares.

Nesse sentido, o editor disponibiliza uma ferramenta para a elaboração de diagramas UML. A UML pode especificar uma ampla variedade de sistemas e seus componentes, podendo representa-los em diferentes níveis de detalhes, mas dentre esses o aplicativo se foca na construção de diagramas de classe simples. Os diagramas de classe são um dos mais importantes especificados pela UML, eles partem do paradigma de programação orientada a objetos permitindo a visualização das classes que compõem o software, com seus atributos e métodos específicos, bem como as diferentes relações que possuem entre si.

Dessa forma, o projeto visa unir o conteúdo aprendido ao longo da disciplina para criar um sistema capaz de fornecer ao usuário a capacidade de elaborar diagramas simples, mas ainda sim flexíveis o bastante para representar um grande conjunto de sistemas.

2. Regras de Negócio

O funcionamento do aplicativo é bastante simples e intuitivo. Ao iniciar é dado ao usuário a opção de criar um novo diagrama do zero ou abrir um digrama feito, para abrir um diagrama é necessário selecionar um arquivo de texto, sendo que é necessário que o arquivo tenha sido gerado próprio aplicativo e não tenha sofrido nenhuma modificação, caso contrário erros irão acontecer. De qualquer forma, o usuário será levado para uma segunda tela onde será possível montar e/ou modificar o diagrama.

O projeto foi inspirado pelo software DIA, uma aplicação gratuita que também permite a construção de diagramas, portanto o funcionamento do ?? segue padrões similares. O aplicativo funciona através de um esquema de quadro branco, serão apresentados alguns componentes para o usuário e estes podem ser colocados e movimentados livremente pelo quadro de acordo com a posição desejado pelo usuário.

O principal componente que pode ser utilizado é o de Classe. Nele o usuário pode interagir e realizar modificações, colocando todas as informações que julgar necessárias, seguindo o estilo especificado na UML. Podem ser acrescentados atributos, cada um com nome, tipo, valor padrão e visibilidade customizáveis, e também métodos, possuindo também nome, visibilidade, tipo e parâmetros específicos. Por fim, as classes em si também podem ser modificadas acrescentando comentários ou as declarando como abstratas ou como uma interface, embora as interfaces não sejam classes em si.

Também é permitido adicionar outros dois componentes simples ao diagrama, uma anotação que permite realizar comentários em locais específicos, e uma estrutura de pacote, que pode ser redimensionado de forma a agrupar outros componentes.

Todos esses componentes podem ser conectados através de linhas especiais que denotam um relacionamento entre componentes. Para isso é necessário selecionar dois pontos distintos em um ou dois componentes e o aplicativo vai se encarregar de gerar automaticamente a linha de relação, sendo que algumas podem ser modificadas posteriormente para adicionar navegabilidade, multiplicidade, entre outros.

Por fim, caso seja de interesse do usuário o diagrama pode ser exportado em uma imagem .png ou salvo em formato de texto em qualquer diretório de escolha.

3. Diagrama de classes e implementação

Abaixo é possível ver o diagrama resumido do projeto com as principais classes. Como prova da funcionalidade do editor o diagrama abaixo foi feito utilizando o próprio aplicativo desenvolvido neste projeto.

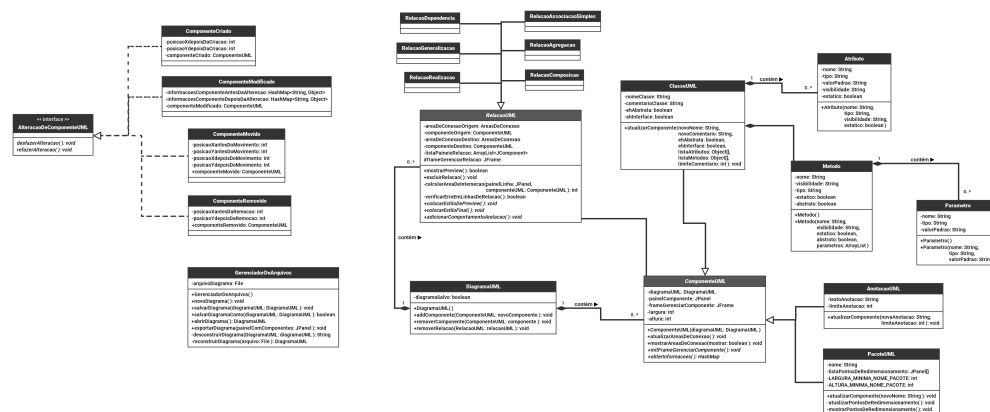


Figure 1. Diagrama de Classes do projeto

3.1. Gerenciador de Arquivos

Se trata de uma classe auxiliar como o objetivo de prestar serviços a interface gráfica. Tem como atributo principal um objeto do tipo File que indica o arquivo em que o diagrama está salvo. Assim, de acordo com a interação do usuário os métodos da classe serão chamados para realizar os procedimentos de abrir, salvar ou exportar um determinado diagrama recebido como parâmetro. Para salvar um diagrama é necessário escolher um diretório e o aplicativo irá gerar um arquivo de texto com as informações necessárias,

para abrir um diagrama é necessário escolher esse mesmo arquivo de texto gerado pelo aplicativo. Para realizar essas ações a classe conta com dois métodos privados, desconstruir e reconstruir diagrama, que ficam responsáveis por transformar o diagrama em texto para ser salvo e vice versa. Por fim, a classe também pode exportar o diagrama em formato .png através do método exportarDiagrama.

3.2. Classes de alteração

Essas se tratam de classes simples que tem como objetivo ajudar a interface gráfica a implementar os comportamentos dos recursos de refazer e desfazer alteração. Toda vez que o usuário realizar uma ação, como adicionar, excluir, mover ou modificar um componente, um novo objeto de uma dessas classes será criado. Elas guardam as informações do componente antes e depois que ele sofreu essa alteração, e através dos métodos herdados da interface AlteracaoDeElementosUML, o aplicativo consegue reconstruir o estado do componente de acordo com a vontade do usuário.

3.3. DiagramaUML

Essa classe representa o diagrama em si. Ela apresenta uma relação de composição com as classes do tipo ComponenteUML e RelacaoUML. Através das ações do usuário os métodos dessa classe são chamados para que os componentes ou relações possam ser adicionados ou removidos. ComponenteUML e subclasses

A classe ComponenteUML se trata de uma classe abstrata da qual as classes ClasseUML, PacoteUML e AnotacaoUML herdam as principais características que vão permitir suas funcionalidades básicas no aplicativo. O ComponenteUML tem como atributo principal um JPanel chamado painelComponente que serve de base para a adição de todos os outros componentes gráficos em cada uma das subclasses. A superclasse acrescenta a esse painel uma série de comportamentos especiais de modo a permitir a adição, remoção e movimentação do componente no quadro. Além disso, a classe também especifica alguns métodos abstratos. O método initFrameGerenciarComponente serve para inicializar a interface gráfica de gerenciamento específica do componente, o obterInformacoes retorna um mapa com as informações do componente. Com relação as subclasses temos a ClasseUML, PacoteUML e AnotacaoUML. A ClasseUML possui como atributos nome, comentário, e dois booleanos indicando se a classe é abstrata ou uma interface. Essa classe por sua vez ainda é composta por outras duas classes que representam seus atributos e métodos. A classe Atributo possui nome, tipo, visibilidade, valor padrão e um booleano indicando se o atributo é estático. Já a classe Método possui ainda um atributo indicando se é abstrato além de outra relação de composição com a Classe Parâmetro. A classe de AnotacaoUML é bastante simples, tendo como atributo principal o textoAnotacao e assim como a ClasseUML um inteiro representando o limite de caracteres para a quebra de linha, além de mais alguns atributos referentes aos componentes gráficos da classe. Por fim, o PacoteUML segue padrões similares as das classes descritas anteriormente, com a diferença da presença de atributos e métodos especiais referentes a pontos de redimensionamento, isso porque essa classe coloca no painelComponente a capacidade especial de poder ser redimensionado de acordo com o arraste do mouse pelo usuário.

4. Telas

Abaixo é possível observar a tela de menu principal, a primeira tela que é apresentada ao usuário, ela possui três botões, um para sair do aplicativo, abrir um diagrama

ou criar um novo diagrama.



Figure 2. Menu Principal

Ao escolher abrir ou criar um diagrama o usuário é levado a tela principal do aplicativo, ela apresenta na parte superior uma barra de opções pela qual é possível realizar algumas ações como salvar, abrir ou exportar diagrama, do lado esquerdo é mostrado uma barra de componentes e relações que o usuário pode adicionar ao quadro que fica no centro da tela, acima desse quadro é possível encontrar outras ferramentas como a de refazer ou desfazer alteração.

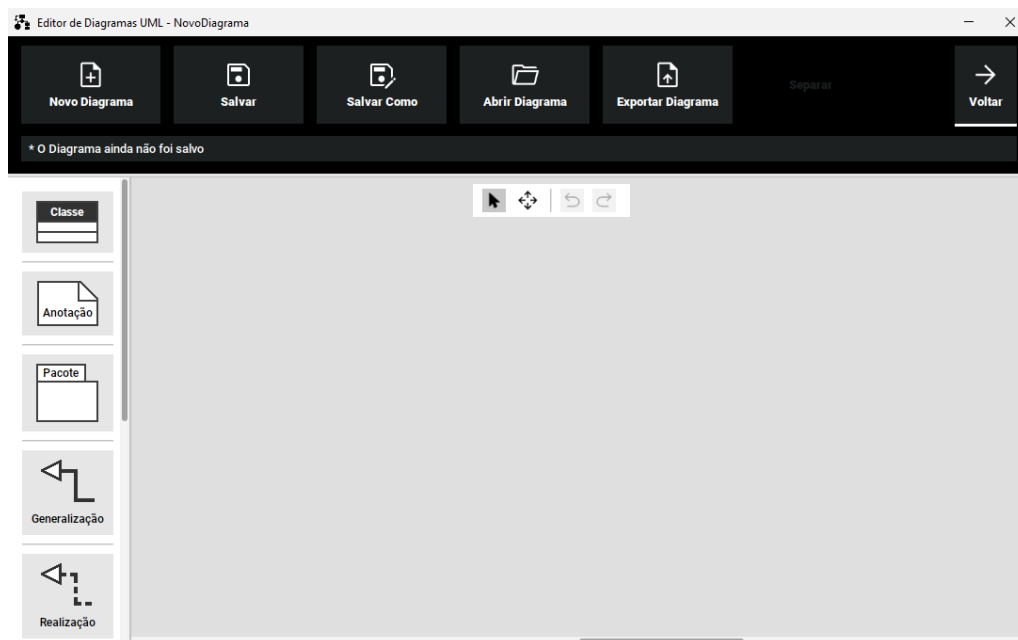


Figure 3. Area de Diagramas

As outras telas se referem a telas de gerenciamento de componentes e relações. Elas possuem campos específicos pelos quais é possível realizar modificações, a de ClasseUML por exemplo possui três abas diferentes pelas quais é possível adicionar e modificar atributos, métodos ou configurar a classe em si.

Já na tela de gerenciamento de algumas relações é possível modificar o sentido da relação ou a excluir por completo, as relações de associação simples, agregação e composição são mais complexas do que as outras, aceitando ainda um nome e sentido customizáveis, além da associação aceitar uma forma de navegabilidade e multiplicidade.

The figure displays five screenshots of a software configuration interface, organized into two rows. The top row contains three windows titled 'Configurar Componente', each with tabs for 'CLASSE', 'ATRIBUTOS', and 'MÉTODOS'. The first window is in the 'CLASSE' tab, showing fields for 'Nome da Classe' (containing 'Nova Classe'), 'Comentários', and checkboxes for 'Abstrata' and 'Interface'. The second window is in the 'ATRIBUTOS' tab, showing a list of attributes with a '+' button, and fields for 'Nome', 'Tipo', 'Valor Padrão', and 'Visibilidade' (set to 'Público'). The third window is in the 'MÉTODOS' tab, showing a list of methods with a '+' button, and fields for 'Nome', 'Tipo', 'Valor Padrão', and 'Visibilidade' (set to 'Público'). Each of these three windows has an 'APLICAR' button at the bottom. The bottom row contains two windows. The first is titled 'Configurar Componente' and is in the 'ANOTAÇÃO' tab, showing a 'Texto' field and a character limit of 20. The second is titled 'Configurar Relação' and is in the 'ASSOCIAÇÃO SIMPLES' tab, showing fields for 'Nome', 'Direção' (set to 'Nenhuma'), 'Mostrar Seta' (with 'Não' selected for both 'Lado A' and 'Lado B'), 'Multiplicidade', and buttons for 'Excluir Relação' and 'Aplicar'.

Figure 4. Alguns exemplos de telas de configuração de componente.