**Texto base**

**10**

**Introdução a UML**

Prof. João de Deus Freire Junior

***Resumo***

*A UML é a linguagem de modelagem unificada. Trata-se de uma linguagem de modelagem de soluções e sistemas. Ela auxilia na compreensão da solução sistêmica na perspectiva interna estática e iterativa do sistema e na perspectiva do cliente. Ela é formada por diagramas que possibilitam essa visão completa de solução. Nesta aula, será exibida as principais definições da UML e seus diagramas.*

**1.1. Introdução**

Há alguma linguagem de modelagem de sistemas? É possível projetarmos uma solução antes de a construirmos? O que significa UML? Quais são os principais diagramas da UML? Todas essas perguntas serão respondidas nesta aula. O entendimento da linguagem UML é essencial para modelagem adequada e abrangente de sistemas, esse conhecimento possibilita conhecer várias perspectivas que um sistema pode ser modelado e projetado antes do seu desenvolvimento. Desta forma, podemos mitigar e diminuir significativamente as falhas e fracassos no desenvolvimentos de projetos de soluções tecnológicas.

**1.2. O que é UML**

UML (Unified Modeling Language – linguagem de modelagem unificada) é uma linguagem-padrão para descrever/documentar projeto de software. A UML pode ser usada para visualizar, especificar, construir e documentar os artefatos de um sistema de software. (Pressman, 2011)

Grady Booch, Jim Rumbaugh e Ivar Jacobson desenvolveram a UML na década de 1990 com muito feedback da comunidade de desenvolvimento de software. A UML combinou um grupo de notações de modelagem concorrentes usadas pela indústria do software na época. (Pressman, 2011)

Em 1997, a UML 1.0 foi apresentada ao OMG (Object Management Group), uma associação sem fins lucrativos dedicada a manter especificações para ser usadas pela indústria de computadores. A UML 1.0 foi revisada tornando-se a UML 1.1 e adotada mais tarde naquele ano. O padrão atual é a UML 2.0 e agora é um padrão ISO. (Pressman, 2011)

**1.3. Principais Diagramas da UML**

A UML 2.0 fornece 13 diferentes diagramas para uso na modelagem de software. Os principais diagramas são:

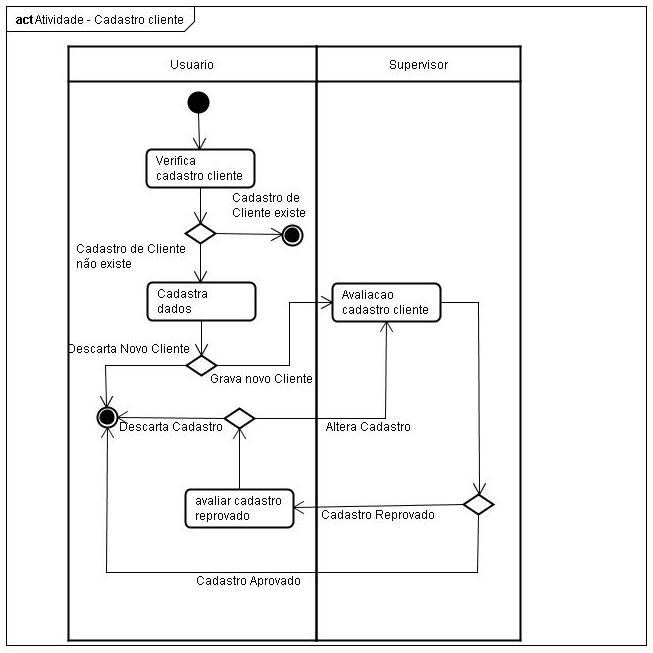
* De classe;
* Distribuição;
* Caso de uso;
* Sequência;
* Comunicação;
* Atividade;
* Estado.

De acordo com Sommerville (2011), através desses diagramas podemos modelar os sistemas em perspectivas diferentes. Por exemplo:

1. Uma perspectiva externa, em que você modela o contexto ou o ambiente do sistema.
2. Uma perspectiva de interação, em que você modela as interações entre um sistema e seu ambiente, ou entre os componentes de um sistema.
3. Uma perspectiva estrutural, em que você modela a organização de um sistema ou a estrutura dos dados processados pelo sistema.
4. Uma perspectiva comportamental, em que você modela o comportamento dinâmico do sistema e como ele reage aos eventos.

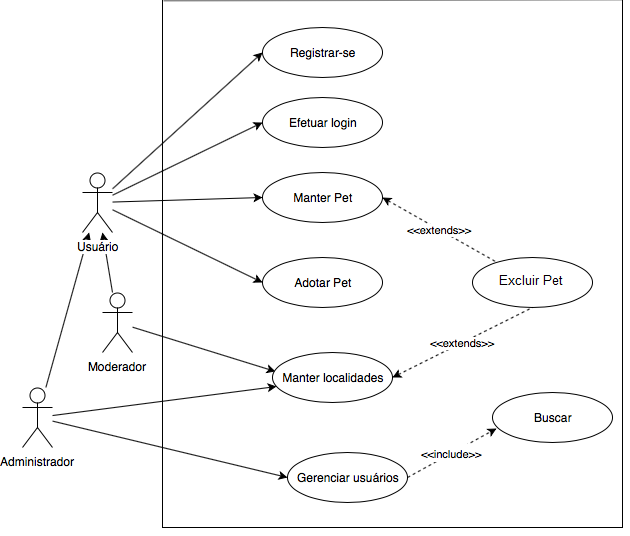
Sommerville (2011) afirma também que os cincos principais diagramas da UML podem representar a essência de um sistema. Eles os fazem da seguinte forma:

1. Diagramas de atividades, que mostram as atividades envolvidas em um processo ou no processamento de dados.



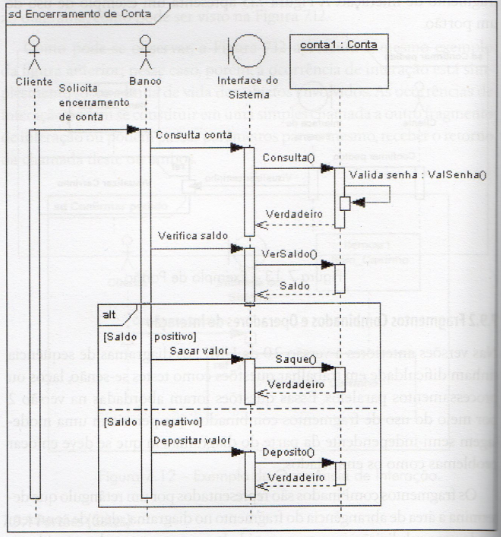
**Figura 1.1. Diagrama de Atividades. Fonte: SOMMERVILLE, Ian.(2011)**

1. Diagramas de casos de uso, que mostram as interações entre um sistema e seu ambiente.



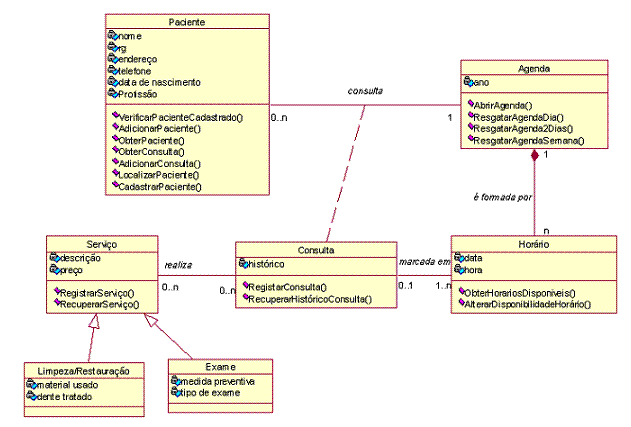
**Figura 1.2. Diagrama de Caso de Uso. Fonte: SOMMERVILLE, Ian.(2011)**

1. Diagramas de sequência, que mostram as interações entre os atores e o sistema, e entre os componentes do sistema.



**Figura 1.3. Diagrama de Sequência. Fonte: SOMMERVILLE, Ian.(2011)**

1. Diagramas de classe, que mostram as classes de objeto no sistema e as associações entre elas.



**Figura 1.4. Diagrama de Classe. Fonte: SOMMERVILLE, Ian.(2011)**

1. Diagramas de estado, que mostram como o sistema reage aos eventos internos e externos.



**Figura 1.5. Diagrama de Estados. Fonte: SOMMERVILLE, Ian.(2011)**

**1.4. Você quer ler?**

Segue uma indicação de estudo complementar. Trata-se de um breve guia da linguagem UML.

* FOWLER, Martin. UML Essencial: Um breve guia para a linguagem-padrão de modelagem de objetos.. 2005. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788560031382/epubcfi/6/8[idloc\_003.xhtml-itemref]!/4[eid604]/20[eid709]/6[eid716]@0:49.4>. Acesso em: 20 dez. 2019.

**1.5. Referências**

* PRESSMAN, R. S.(2011) Engenharia de Software: uma abordagem profissional. 7.ed. Porto Alegre: Bookman, 2016.
* SOMMERVILLE, Ian.(2011) Engenharia de Software. 9.ed. São Paulo: Pearson Prentice Hall, 2011.
* Figuras 1.1, 1.2, 1.3, 1.4 e 1.5: SOMMERVILLE, Ian.(2011)