



Centro Universitário Senac Santo Amaro



POO-Programação Orientada a Objetos

Atividade Discente Orientada ADO #1

Eric Paixão Andrade

Professor: Carlos Veríssimo

São Paulo

2023

1.0 Objetivo desta ADO/Pesquisa.

O objetivo dessa pesquisa é proporcionar conhecimentos sólidos nos fundamentos do paradigma da Orientação a objeto.

O objetivo subjacente é proporcionar condições de desenvolver soluções robustas e com qualidade.

2.0 Introdução a UML.

Em seu primórdio, a UML (Unified Modeling Language) surgiu com o objetivo de juntar as diversas linguagens de modelagem da época, tornando-se, desta forma, a linguagem universal de modelagem de sistemas. Desde sua criação, mais de 25 anos se passaram e, no cenário atual, com a crescente tendência de adoção de metodologias ágeis, pouco se sabe sobre o quão é utilizada a UML no contexto profissional na atualidade, grande polo de tecnologia. Por outro lado, no cenário acadêmico, a UML continua sendo ensinada no contexto dos cursos que envolvem, de alguma forma, os conceitos de engenharia de software. Já por outro ponto de vista, o cliente necessita entender o que os projetistas estão construindo e precisam ter condições de inferir os seus conhecimentos nos projetos para atender plenamente suas necessidades. Para isso é necessário estabelecer um canal formal de interação onde a linguagem natural do cliente é transformada em linguagem técnica para a equipe de desenvolvimento. Com esse objetivo a UML se propõe a ser uma linguagem padrão aceita e compreendida por todos os stakeholders. A modelagem utilizando a UML é focada no paradigma Orientado a Objetos (OO) cujos conceitos são classe, objeto, herança, polimorfismo, encapsulamento de atributos e métodos, alta coesão e baixo acoplamento. É usado para a análise não focando na codificação do software ou hardware e sim no entendimento do problema (análise) e na sua solução (projeto). A UML representa símbolos, esses usados em diagramas que assim representam uma linguagem simbólica com regras claras e precisas para a utilização desses símbolos nos diversos diagramas. O objetivo dos diagramas é apresentar múltiplas visões do sistema chamado de modelo. Assim, um modelo UML é um conjunto de diagramas que servem para compreender e desenvolver um projeto de software, descrevendo o que o software deve fazer.

3.0 Dissertação, elementos históricos e linha do tempo da UML.

Produzir software de qualidade, com custos e prazos reduzidos deixou de ser desejável e passou a ser uma necessidade. Diversas técnicas, metodologias e ferramentas vêm sendo propostas com o intuito de dar suporte e auxiliar a produção de software de qualidade. Um dos mais importantes avanços na comunidade de engenharia de software foi o surgimento da Linguagem de Modelagem Unificada (BOOCH et al., 2000).

A UML teve origens em uma fusão de várias linguagens de modelagem que existiam nas décadas de 1980 e 1990. Essas linguagens incluíam a linguagem de diagramas de classes do método Booch, a linguagem de notação de estados e transições do método Statechart de Harel e a linguagem de diagramas de objetos do método OMT (Object Modeling Technique) de Rumbaugh. Os principais contribuintes para a criação da UML foram Grady Booch, Ivar Jacobson e James Rumbaugh, que eram líderes em suas respectivas áreas de métodos de modelagem. Em 1997, eles se uniram para desenvolver uma linguagem de modelagem unificada, o que levou à criação da UML 1.0.

A primeira versão da UML, a UML 1.0, foi lançada em 1997. Ela incluiu uma variedade de diagramas para representar diferentes aspectos de sistemas, como diagramas de classes, diagramas de sequência, diagramas de estados e muitos outros. É importante observar que havia uma dificuldade inicial, por parte dos desenvolvedores, devido a diversos métodos, visto que não atendiam completamente às suas necessidades. Neste contexto alguns métodos se destacaram: Notação de Booch; o OOSE (Object-Oriented Software Engineering) de Ivar Jacobson e o OMT (Object Modeling Technique) de James Rumbaugh.

A UML 2.0 foi lançada em 2005 e representou uma revisão significativa da UML 1.0. Ela introduziu uma série de melhorias, incluindo novos tipos de diagramas, como diagramas de componentes e diagramas de implantação, além de refinamento nos diagramas existentes. A UML é uma linguagem padronizada, e suas especificações são mantidas pela Object Management Group (OMG), uma organização internacional que desenvolve padrões para tecnologias de modelagem e sistemas distribuídos.

A versão mais recente da UML é a 2.5, que foi lançada em 2015. Ela trouxe algumas melhorias adicionais, como suporte para modelagem de tempo e aprimoramento em diagramas existentes.



A UML se tornou uma linguagem padrão amplamente adotada na indústria de desenvolvimento de software. Ela é usada para projetar sistemas complexos, documentar arquiteturas de software e auxiliar na comunicação entre desenvolvedores e partes interessadas.

É importante observar que, ao longo de sua evolução, a UML tem passado por várias revisões e atualizações para manter sua relevância e utilidade no campo do desenvolvimento de software. Portanto com bases em minhas pesquisas pode ser que algo tenha passado ou esteja fora desta pesquisa com base nas fontes utilizadas.

4.0 Referencial Bibliográfico

- Linguagem de Modelagem Unificada (UML) (Revista Fundamentos da Engenharia de Software Livro 1)
- UML: Um estudo sobre o uso em empresas de desenvolvimento de software em São Carlos - SP e região - por Luiz Henrique Ten Caten Bento.

fonte: [https://repositorio.usp.br/directbitstream/ec859ca0-3786-451d-80ba-16b2ade86bf6/TCC LuizHenrique.pdf](https://repositorio.usp.br/directbitstream/ec859ca0-3786-451d-80ba-16b2ade86bf6/TCC%20LuizHenrique.pdf)

- The Unified Modeling Language User Guide - Grady Booch, James Rumbaugh.
- Uma Abordagem para a Construção de Diagramas da UML Concomitante à Prototipação de Interface - Rodrigo M. Bacurau, Braulio G. Leal, Ricardo A. Ramos

fonte: <http://www.univasf.edu.br/~brauliro.leal/pesquisa/Bacurau-5.pdf>

- UML - Facom/UFU

fonte: <https://www.facom.ufu.br/~flavio/poo/files/2011-02/uml>