

## **PROCESSOS DE ENGENHARIA DE SOFTWARE NO DESENVOLVIMENTO DE PROJETOS DE WEBSITES E SISTEMAS WEB**

André Bessa da Silva<sup>1</sup>

### **RESUMO**

Este artigo busca apresentar conceitos norteadores de Engenharia de Software aplicados a WEB, métodos e metodologias e demais ferramentas que podem ser utilizadas pelos profissionais no desenvolvimento de sites e sistemas para internet, tendo enfoque especial nas técnicas de Engenharia WEB no processo de criação de soluções que priorizem sobretudo a qualidade nas atividades de desenvolvimento de aplicações que rodam sobre a plataforma da internet.

**PALAVRAS-CHAVE:** Engenharia de Software, Engenharia WEB, Sistemas WEB.

### **INTRODUÇÃO**

Com o advento da internet e a revolução propiciada por este veículo de comunicação possibilitou as pessoas uma nova maneira de se relacionar e permitiu que empresas e profissionais dos mais diversos ramos de atividade pudessem de maneira singular interagir com seu público-alvo. Hoje é correto afirmar que a internet se configura em um dos principais meios de prestação de serviços e tem atingindo um número cada vez maior de pessoas.

Atividades tais como, realizar compras on-line, cursos a distâncias, participação em eventos, realização de transações bancárias via portal ou smartphones e entretenimento são exemplos dos benefícios propiciados pela utilização da internet. Neste cenário a web é uma das principais plataformas de interação entre pessoas e organizações, como era de se esperar muitas empresas dos mais variados portes e profissionais dos mais diversos ramos de atuação tem utilizado o poder da rede para alavancar seus negócios como: venda de produtos e

---

<sup>1</sup> Especialista em Engenharia de Software pela Universidade Estácio de Sá – Rio de Janeiro/RJ. Bacharel em Sistemas de Informação pelo Centro Universitário São Camilo-ES. E-mail: andrebessax@gmail.com.

serviços; visando atingir um contingente cada vez maior e mais exigente consumidores.

Independente da clientela que se deseja atingir ou atender com uma solução web, deve-se garantir um mínimo de qualidade e satisfação dos usuários ao interagir com os diversos sites e sistemas web hoje disponibilizados na rede.

Com está crescente necessidade de novas soluções para plataformas WEB, se faz necessário a utilização dos recursos disponibilizados pela Engenharia de Software, como processos e métodos, que além de focar em satisfação do usuário com o produto final, propiciam ao desenvolvedor da solução técnicas de documentação e manutenção, que serão de vital importância ao projeto durante o desenvolvimento e posterior implantação.

A Engenharia de Software deve ser adaptada as necessidades e particularidades de cada projeto, e se encontram sempre em rápida evolução e atualização tecnológica. Como veremos no decorrer deste trabalho conceitos de Engenharia de Software aplicada ao desenvolvimento de aplicações web, Engenharia Web e metodologias como a Engenharia Web Baseada em UML (UML-based Web Engineering – UWE).

O mercado de desenvolvimento de sistemas de maneira geral tem se voltado para a criação de soluções que rodem na web, portanto profissionais que atuem no desenvolvimento de sites e sistemas web que não aplicam técnicas de Engenharia de Software em seus projetos, correm um risco grande de não atingirem a qualidade técnica mínima necessária à implementação de seus projetos.

Este artigo tem como objetivo apresentar os principais processos da Engenharia de Software aplicados ao desenvolvimento de sistemas web e websites, conceitos relacionados a metodologias de desenvolvimento como a orientação a objetos e os principais diagramas da UML que pode ser aplicado no desenvolvimento de projetos voltados para internet.

## **1 ENGENHARIA DE SOFTWARE**

A Engenharia de Software de acordo com LOBO (2008) é definida como a ciência que busca investigar e catalogar as metodologias e padrões para o desenvolvimento de Softwares, e tem sido de grande importância para Fábricas de Softwares e empresas desenvolvedoras por definirem métodos sistêmicos que buscam amadurecer as técnicas e ferramentas além de todas as atividades inerentes ao ambiente de desenvolvimento.

Ainda de acordo com LOBO (2008), para a criação de uma solução de qualidade, deve ser necessária a aplicação de uma disciplina por toda a equipe que participa do projeto. Uma vez que a produção de um software de qualidade não é tarefa das mais fáceis, daí a grande importância da aplicação da Engenharia de Software.

Em Engenharia de Software a adoção de processos é de suma importância como ressalta LARMAN (2007), os processos de desenvolvimento de softwares descrevem uma abordagem para a construção, implantação e manutenção, atividades estas que muito bem se aplicam ao desenvolvimento de websites e sistemas web.

Por fim entende-se como Engenharia de Software como o emprego de processos, metodologias e padrões a serem aplicados no desenvolvimento de sistemas. Sempre empregando as melhores práticas de maneira ordenada e sistêmica por toda a equipe envolvida na construção do projeto.

## **2 ENGENHARIA WEB**

O termo Engenharia WEB é uma designação relativamente nova, e é a adoção sistêmica, disciplinada e quantificável para o desenvolvimento de aplicações WEB, com alta qualidade e um custo efetivo DZENDZIK (2005).

A Engenharia WEB envolve várias disciplinas como: Comunicação hipermídia, Banco de dados, Interface Homem-máquina, Gerência, Computação gráfica, Linguística computacional, Análise de dados.

A Engenharia WEB absorve muito dos conceitos e práticas da Engenharia de Software tradicional, mas as duas não são iguais, pois muitos tópicos usados em processos de desenvolvimento de software tradicional não podem e não são, aplicadas a soluções WEB.

Um sistema WEB possui características próprias como a gerência de grandes volumes de dados, interação e navegação controlada pelo usuário da aplicação, atualizações tecnológicas constantes, que buscam proporcionar a estes mesmos usuários uma experiência confortável, para que continue a usar a aplicação ou Website.

Os websites podem variar de páginas estáticas até verdadeiros sistemas baseados na Web (lojas virtuais, ambientes cooperativos, sistemas de gerência empresarial, etc.). Este tipo de website é definido como Sistemas de Informação Web (Web-based Information Systems - WISs) e ao conjunto amplo de todas as aplicações Web como Web Applications (WebApps). (KLEEMANN, 2016).

Portanto o ambiente de aplicações WEB ou WebApps possui várias características, que são intrínsecas ao próprio meio como: concorrência, carga imprevisível, disponibilidade, sensibilidade ao conteúdo, evolução dinâmica, imediatismo, segurança e estética, tudo isso necessita de uma nova abordagem ao que tange os processos já existentes na tradicional Engenharia de Software.

### **3 PROCESSO DE DESENVOLVIMENTO**

Um processo de desenvolvimento, no que tange as atividades de produção de software, pode ser definido como as atividades em fases ordenadas, que objetivam definir e organizar o ambiente de desenvolvimento de sistemas, podendo ser pesados, leves ou customizáveis (LOBO, 2008).

PRESSMAN (2011) define processo como sendo um conjunto de atividades, ações, tarefas combinado para a criação de algum produto de trabalho, como apresentado na Figura 1. Segundo a Engenharia de Software, um processo não é uma rígida descrição de como desenvolver um software, mas ao contrário disso, é algo adaptável que possibilita aos envolvidos no projeto realizar um trabalho de maneira a selecionar as melhores ações e tarefas, sempre com o intuito de entregar um produto com o máximo de qualidade e dentro dos prazos estipulados.

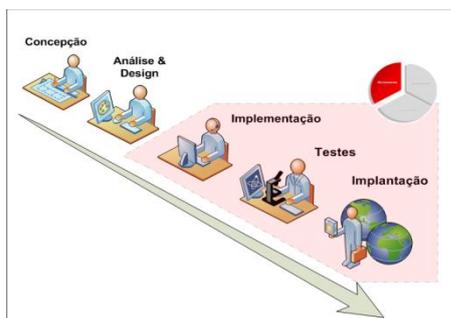


Figura 1 – Processo de Desenvolvimento de um software.

Dentre os processos de desenvolvimento utilizados no mercado um bastante popular é o Processo Unificado (PU), que apresenta um conjunto de disciplinas ao longo de fases, como explicitado na Figura 2. Este processo surgiu como iterativo para a construção de sistemas Orientados a Objetos.

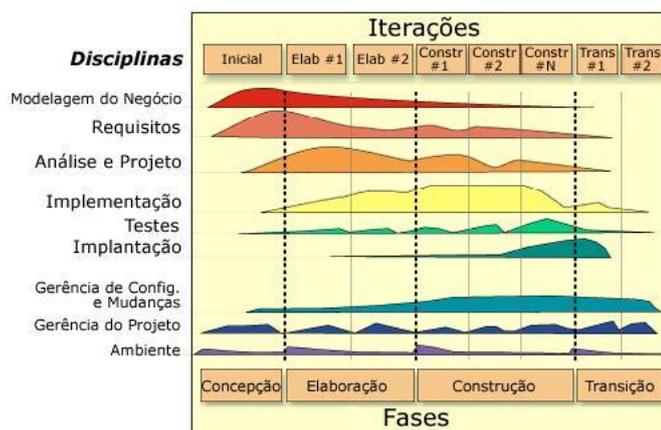


Figura 2 – Exemplo fases e disciplinas de um processo unificado, RUP.

O PU é bastante flexível e adaptativo e corrobora com a adoção de outras práticas e métodos como: eXtreming Programming(XP), Scrum. Por exemplo, o desenvolvimento guiado por testes e integração contínua práticas comuns do XP podem ser aplicadas a processos da PU.

Uma prática fundamental do PU é o desenvolvimento iterativo, que nada mais é que a abordagem do ciclo de vida de desenvolvimento organizado em uma série de miniprojetos de curta duração chamada iteração, ao fim de cada iteração temos uma parte funcional do projeto em construção. Cada iteração refina e incrementa a iteração anterior.

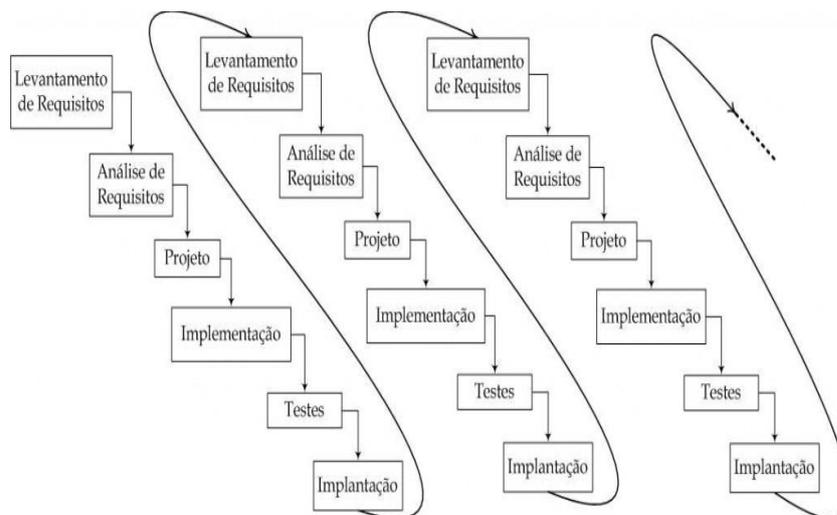


Figura 3– Processo Iterativo.

Como exposto na Figura 3, cada iteração terá suas rotinas de requisitos, análise, projeto, implementação testes e implantação. Como produto da iteração tem-se um executável ou ainda uma funcionalidade ou componente.

Com esta abordagem iterativa no projeto, pode-se responder mais rapidamente as mudanças de rumos, que são comuns em muitos projetos e com uma segurança ainda maior.

#### 4 METODOLOGIA DE DESENVOLVIMENTO

Todavia um processo de desenvolvimento de software ainda pode ser algo “rígido”, no que tange as suas características e implementação, contudo nos últimos anos o que se tem visto é a adoção cada vez maior de métodos, principalmente aqueles com características ágeis. Métodos ágeis têm foco na definição de um ambiente de desenvolvimento controlado e que disponibiliza soluções para inúmeras situações dentro deste ambiente.

Um método de desenvolvimento de software, engloba um conjunto de regras e atividades em fases ordenadas, que objetivam organizar o ambiente de desenvolvimento, focando em torna-lo adaptativo as mudanças. Em sua maioria possuem fases pequenas em curta duração no tempo.

Importante ressaltar que a adoção de métodos e processos estão ligadas diretamente as características de cada projeto LOBO (2008). Para exemplificar, projetos de sistemas grandes e complexos seriam mais bem conduzidos com a adoção de processos mais robustos como o PU, em casos de softwares menores ou sujeito a mudanças repentinas ou que a mudança seja algo intrínseca ao projeto, por vezes métodos iterativos como XP ou Scrum podem ser uma solução mais atrativa.

#### **4.1 ORIENTAÇÃO A OBJETOS**

De acordo com ARAÚJO (2006), a Orientação a Objetos, coloca o desenvolvimento de sistemas em um novo nível. A Orientação a Objetos é um mecanismo que possibilita a definição da estrutura da aplicação como uma abstração das características do mundo em que vivemos.

Dentre os padrões hoje utilizados para o desenvolvimento de software, o adotado na maioria dos projetos é a Orientação a Objetos (OO). Em Orientação a Objetos as classes são o cerne para a criação dos objetos que serão utilizados em todo o sistema. Os objetos têm a capacidade de comunicar-se entre si, além de executar ações na memória do computador.

Quando tratamos do desenvolvimento de aplicações sob o paradigma da Orientação a Objetos são criados modelos que visam representar de uma forma rápida e efetiva elementos de projeto. Geralmente tais modelos são criados antes da codificação do projeto e buscam detalhar todo o sistema pelo intermédio de diagramas, na imagem da Figura 4, é apresentado dois diagramas o de classes e o de objetos. Para o desenvolvimento destes diagramas a especificação utilizada atualmente é a da Linguagem de Modelagem Unificada (UML), a ser vista no tópico 4.2.



Figura 4– Definição de classe e objeto na UML.

## 4.2 MODELAGEM UNIFICADA (UML)

A Linguagem de modelagem unificada (UML) é uma linguagem visual que ajuda aos envolvidos em projetos de software especificar, construir e documentar artefatos de sistemas de um modo fácil e rápido para todos os integrantes da equipe de um determinado projeto.

“Ela pode ser utilizada no apoio às fases de análise, projeto e implementação, pois nos permite pensar antes de codificar. Trabalha com um conjunto de diagramas que permitem uma notação clara e consistente”. (GOÉS, 2014).

De acordo com LARMAN (2007), a palavra visual, na definição é um ponto chave, pois é um padrão para desenhar ou apresentar figuras relacionadas a software principalmente àqueles desenvolvidos sob o paradigma orientado a objetos.

Para DZENDZIK (2005), ao utilizar a UML para projetos Web não exploramos todos os recursos de classes, objetos, relacionamentos, fluxos, mensagens e outras entidades comuns da orientação a objetos, mas fazemos uso de recursos da UML comuns na busca de melhores projetos de Web site. Como recursos da UML muito utilizados em projetos podemos ver o uso de diagramas como, por exemplo, diagramas de seqüência, diagramas de caso de uso, diagramas de classes e demais diagramas que possam ser utilizados na modelagem de projetos Web.

O Diagrama de Sequencia da UML, Figura 5, é um artefato criado para ilustrar os eventos de entrada e saída relacionados com o sistema em discussão, LARMAN (2007). Este diagrama é um dos mais importantes por representar toda a troca de mensagens entre os objetos que compõe o sistema em um panorama de visão de ordem temporal.

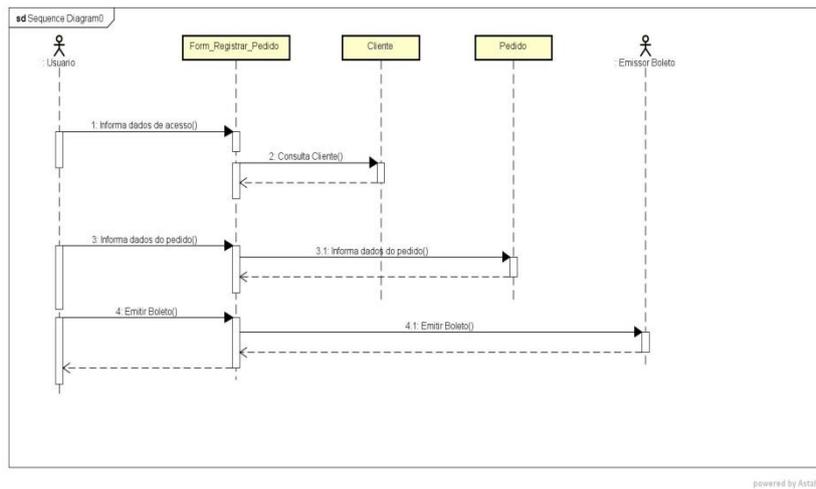


Figura 5 – Diagrama de sequência UML.

Diagrama de caso de uso, Figura 6, é um diagrama da UML composto por narrativas textuais, amplamente utilizadas para descobrir e registrar requisitos e podem influenciar muitos aspectos do projeto, LARMAN (2007).

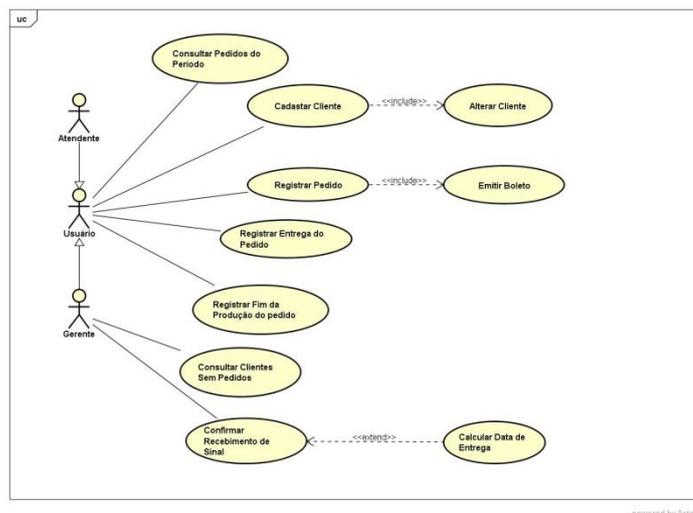


Figura 6 - Diagrama de caso de uso da UML.

Diagrama de classes, Figura 7, é o mais amplamente utilizado da UML, neste diagrama são representados as estruturas abstratas do software, ou seja, tudo a ser realizado pelo sistema a ser desenvolvido.

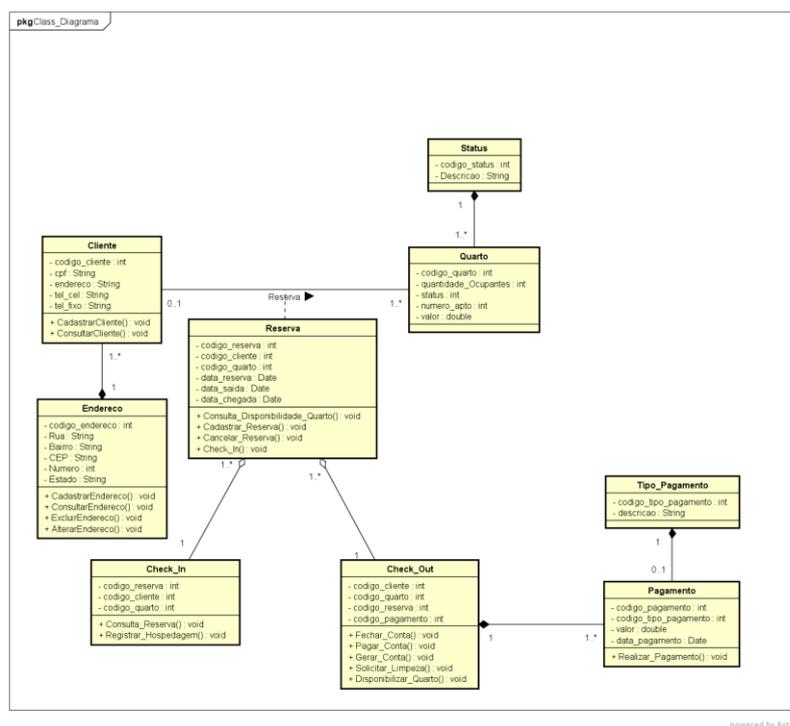


Figura 7 – Diagrama de classes da UML.

Ainda de acordo com DZENDZIK (2005), o objetivo é buscar na modelagem de interfaces Web com a UML, uma linguagem comum para todos os atores envolvidos no projeto como: desenvolvedores, administradores de sistemas, testadores, analistas, clientes.

## 5 QUALIDADE DE SOFTWARE

De acordo com PRESSMAN (2007), qualidade de software pode ser entendida como sendo uma gestão de qualidade efetiva e que deve ser aplicada para a criação de um produto útil e que produza um valor que seja mensurável para aqueles que o produzam e para aqueles que deles utiliza.

A qualidade de um produto tem um propósito: satisfazer o cliente. Esse objetivo implica tratar um domínio, em geral, bastante nebuloso. Para compreender o motivo, considere novamente o caso de uma pessoa que deve adquirir um produto comum no mercado. Ninguém compra uma camisa pensando nas propriedades mecânicas do tecido com o qual ela foi fabricada: "Que bela camisa! Pode resistir a 10 kg de tração!". Em vez disso, são fatores muito difíceis de medir que, em geral, terão maior peso na decisão. (KOSCIANSKI, SOARES 2007).

No desenvolvimento de sistemas, aplicativos ou websites o fator qualidade assume grande importância e deve ser uma das prioridades desde as fases iniciais de projeto.

Aplicar qualidade ao produto que está sendo desenvolvido permite a criação de soluções capazes de atender a necessidade de usuários da melhor maneira possível. Independente dos métodos ou processos aplicados no desenvolvimento será de suma importância aplicar alguns conceitos que ajudam a modelar características de qualidade como :

- **Funcionalidade:** característica de qualidade que garante que o software ou solução realiza funções que atende as reais necessidades de seus usuários.
- **Legalidade:** garantia que a solução implementada está de acordo com a legislação vigente.
- **Segurança:** Essencial em todo tipo de sistema, mas, sobretudo em aplicações WEB. É a garantia que um sistema de informação não permitirá o acesso de pessoas não autorizadas aos dados e informações sensíveis tratadas ou armazenadas na aplicação.
- **Manutenibilidade:** Garantia a evolução e correção da solução.
- **Suporte:** Um socorro permanente aos usuários da aplicação.
- **Intuitivo:** A aplicação deve permitir que seus usuários encontrem funcionalidades de maneira fácil e amigável.
- **Praticidade:** facilidade de operação do sistema.
- **Eficiência:** tempo de resposta e processamento levando em consideração todos os recursos utilizados no sistema.

- Backup: Todo software que prima por qualidade deve possuir alguma rotina de backup de seus dados.

Estas características de qualidade de software se aplicam a qualquer solução a ser desenvolvida. Quando estes requisitos forem alcançados teremos um software com qualidade satisfatória ao usuário.

## **6 METODOLOGIAS PARA DESENVOLVIMENTO DE WEBSITES E SISTEMAS WEB**

Dentro da UML foi criado um conjunto de modelos que visam dar suporte ao desenvolvimento de WebApps essa metodologia é chamada Engenharia Web Baseada em UML (UML-based Web Engineering – UWE), ela congrega uma linguagem de modelagem para a construção de artefatos e modelos. O UWE possui entre suas principais atividades um processo de modelagem que é composto por: análise de requisitos, projeto conceitual (modelo de classes), projeto de navegação, como apresentado na Figura 8, um projeto de apresentação, modelagem de tarefas e por fim implantação.

O UWE toma como base o desenvolvimento de WebApps concentrando especial atenção à sistematização, focando no código e como automatizar a geração do mesmo, nos casos de uso para a especificação de requisitos e no projeto conceitual para a definição de classes e suas associações.

No Projeto de Navegação, atividade de grande importância neste modelo de desenvolvimento é em geral é subdividida em duas etapas:

- *Espaço de navegação*: baseia-se no modelo conceitual e nos requisitos já definidos nos casos de uso do projeto e sua função é especificar quais das classes do modelo conceitual construído estarão visíveis ao usuário e como navegar até chegar a essas classes.
- *Estrutura de navegação*: um refinamento do que foi estipulado no espaço de navegação. Nesta fase de modelagem adicionamos outros elementos que ajudam a estruturar a navegação tais como índices e menus.

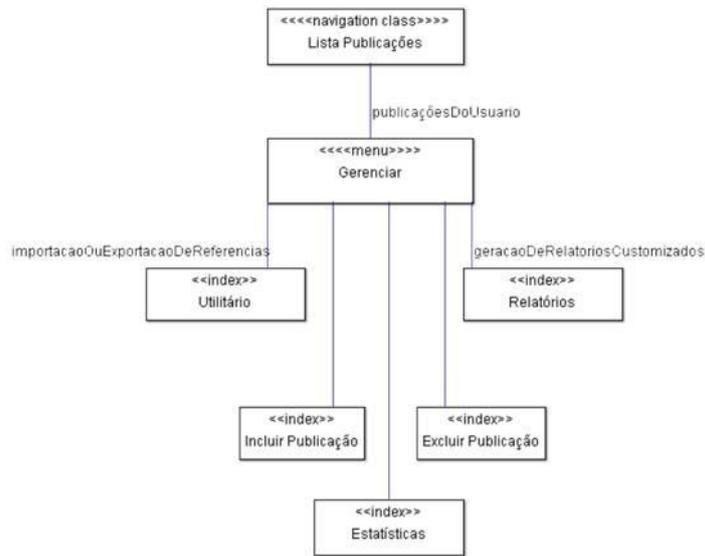


Figura 8– Exemplo de modelo de navegação. Fonte: Disponível em <<http://www.devmedia.com.br/engenharia-web-baseada-na-uml/31001>>. Acesso: 25 de Maio de 2016.

No modelo de apresentação, exposto na Figura 9, é proposto o layout e aparência das páginas da WebApps que será desenvolvida. De acordo com MEDEIROS (2016), o modelo de apresentação foca na organização estrutural da apresentação das páginas, como exemplo, a disposição dos textos e imagens, a utilização de formulários e etc.

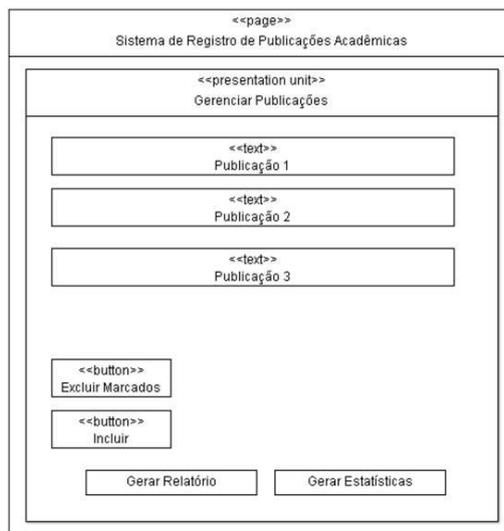


Figura 9 – Exemplo de modelo de apresentação. Fonte: Disponível em <<http://www.devmedia.com.br/engenharia-web-baseada-na-uml/31001>>. Acesso: 25 de Maio de 2016.

Dentro da UWE foi criada uma série de estereótipos, que são identificados ("<<" e ">>"), para uma melhor representação dentro da UML a fim de facilitar o entendimento e intuitividade na elaboração de diagramas com foco no desenvolvimento WEB. A essa extensão é dado o nome de WAE (Web Application Extensions), Extensão de Aplicações WEB, que congrega os meta-modelos da UML a fim de especificar características das aplicações WEB. A seguir, na tabela1 , com alguns estereótipos utilizados e na Figura 10, um diagrama de classes usando o estereótipos WAE.

Estereótipo	Significação
<<page>>	Representa uma página web.
<<server page>>	Representa uma página dinâmica com elementos que são processadas no lado do servidor.
<<cliente page>>	Página estática a ser renderizada no navegador cliente.
<<HTML form>>	Usado para a representação de um formulário de entrada de dados HTML.
<<link>>	Um link entre páginas, representado em HTML pela tag <a>.
<<anchor>>	Ancora de navegação HTML.
<<button>>	Botão HTML.
<<build>>	Representa que o resultado de um processamento no servidor resultará em uma <<client page>>
<<redirect>>	Representa um redirecionamento dentro de um diagrama.
<<forward>>	Delega uma página para outra. Diferente do <<redirect>> por manter o contexto da requisição feita à primeira página anterior.

<<text>>	Representação de um texto comum.
<<index>>	Página index da aplicação.

Tabela 1 – Exemplos de Esteriótipos WAE.

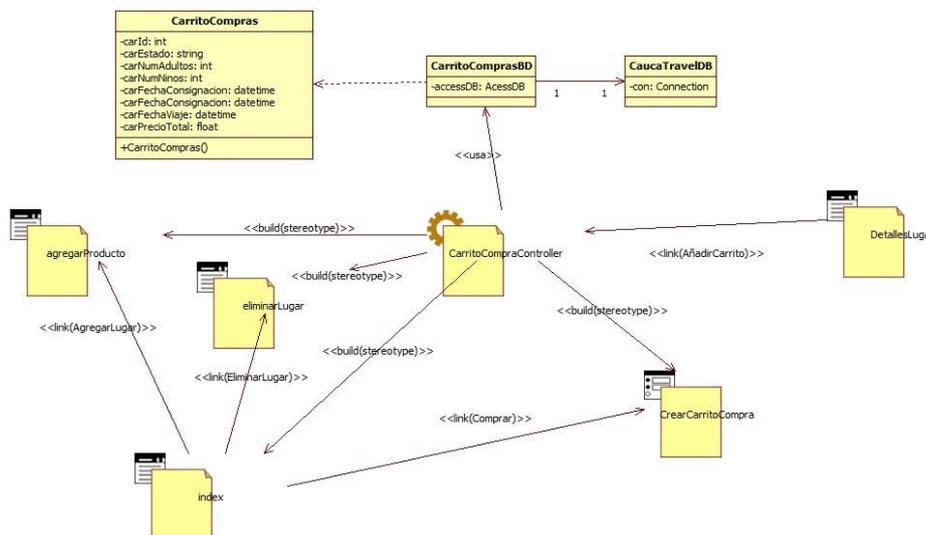


Figura 10 – Exemplo de diagrama de classes usando esteriotipos WAE. Fonte: Disponível em < <https://caucatruvel.wordpress.com/category/diagramas-wae/>>. Acesso: 25 de Maio de 2016.

## CONSIDERAÇÕES FINAIS

A atividade de desenvolvimento de software de um modo geral implica na adoção de melhores práticas de Engenharia de Software, daí a necessidade de se conhecer seus conceitos e definições, bem como de Engenharia Web que é a aplicação da Engenharia de Software tradicional no ambiente de desenvolvimento de Websites e aplicações WEB.

No artigo foram apresentados alguns conceitos chave no desenvolvimento de software como Processos e Métodos, que podem ajudar o desenvolvedor, a criar e planejar as atividades que irão compor seu trabalho de construção de um software que foque em qualidade e atenda as características exigidas pelos clientes. Soma-se a isso a utilização do paradigma orientado a objetos apoiado na linguagem de modelagem UML que ajuda a compor um ferramental para a construção de

diagramas e documentos que nortearão toda a equipe envolvida na elaboração da modelagem conceitual e estrutural do projeto.

Por fim apresenta-se a Engenharia Web Baseada em UML (UML-based Web Engineering – UWE), como uma ferramenta que pode ser utilizada amplamente para o desenvolvimento de aplicações baseadas em WEB, que possibilita estender uma série de funcionalidades e diagramas já utilizados pela Engenharia de Software tradicional. Este é um campo dentro da Engenharia de Software ainda pouco explorado pelos profissionais que trabalham desenvolvendo soluções de softwares para a WEB e se traduz numa oportunidade ímpar, de busca pela qualidade de aplicações tanto almejadas pela Engenharia de Software, agora traduzida em uma Engenharia WEB.

**ABSTRACT**

**SOFTWARE ENGINEERING PROCESSES IN THE DEVELOPMENT OF WEBSITE  
AND WEB SYSTEMS PROJECTS**

This article aims to present guiding concepts of Software Engineering applied to the WEB, methods and methodologies and other tools that can be used by professionals in the development of websites and systems for the internet, with special focus on WEB Engineering techniques in the process of creating solutions that give priority to quality in application development activities that run on the internet platform.

**KEYWORDS:** Software Engineering, WEB Engineering, WEB Systems.

## REFERÊNCIAS BIBLIOGRÁFICAS

ARAÚJO, Anderson Viçoso de, **Treinamento Avançado em .NET**, São Paulo: Digerati Books, 2006.

DZENDZIK, Isolete Teresinha, **PROCESSO DE DESENVOLVIMENTO DE WEB SITES COM RECURSOS DA UML (Mestrado em Computação Aplicada)**, Instituto Nacional de Pesquisas Espaciais(INPE), 2005.

GOÉS, Wilson Moraes ,**Aprenda UML por Meio de Estudos de Caso**, São Paulo: Novatec, 2014.

LOBO, E. J. R., **Curso de Engenharia de Software**, São Paulo: Digerati Books, 2008.

LARMAN, C. **Utilizando UML e Padrões**, 3. ed. Porto Alegre: Bookman, 2007.

KOSCIANSKI, André, SOARES, Michel dos Santos. **Qualidade de Software**, 2 ed. São Paulo: Novatec, 2007.

KLEEMANN, Jean Wagner, **Engenharia de Componentes - Parte 1**, Disponível em < <http://www.linhadecodigo.com.br/artigo/3119/engenharia-de-componentes-parte-1.aspx>>. Acesso em 1 Agosto, 2016.

PRESSMAN, R. S., **Engenharia de Software**, . ed. São Paulo: Mc. Graw Hill, 2011.

MEDEIROS, Higor, **Engenharia Web baseada na UML**. Disponível em < <http://www.devmedia.com.br/engenharia-web-baseada-na-uml/31001>>. Acesso: 25 de Maio de 2016.

TURQUETTE, D. F. **Engenharia de Software para WEB**. Barbacena, UNIPAC. Disponível em < <http://www.unipac.br/site/bb/tcc/tcc-5e3b4b161a0feb21fdac64954fefe51.pdf>>. Acesso: 28 Maio. 2016.