

PADRÕES DE PROJETO PARA WEB

PEREIRA, Alexandre¹, COSTA, Eduardo Isidoro da¹, MENDES, Filipe Moreira¹, PEREIRA, Renato de Ávila¹

REIS, José Cláudio de Sousa²

¹ Acadêmicos do 8º período do curso de Bacharelado em Ciência da Computação – UNIFENAS - Alfenas.

² Professor do curso de Bacharelado em Ciência da Computação, UNIFENAS - Alfenas.

RESUMO. *Padrões de projeto são soluções para problemas existentes encontrados no desenvolvimento do software. De tal modo, os padrões de projetos compreendem o vínculo entre um contexto, um problema e uma solução. Esse trabalho tem o objetivo de mostrar a eficiência na aplicação de padrões de projeto para web. No desenvolvimento pressupõe-se desenvolver duas aplicações, sendo uma com a utilização de padrões e outra sem a utilização de padrões. A implementação das aplicações foi feita com a linguagem C# .NET com orientação a objetos e programação em camadas conciliado com as tecnologias HTML5 e CSS3, usando a IDE Visual Studio Express 2012. Com a utilização de padrões, foi possível verificar um maior reaproveitamento de código e agilidade no desenvolvimento de software e em futuras manutenções acarretando em uma maior produtividade e qualidade nos sistemas. No que diz respeito aos usuários houve um grande benefício demonstrando uma maior usabilidade e acessibilidade do sistema. Com isso, os padrões de projeto podem gerar uma maior viabilidade para construção de software. Finalmente, o software demonstra a utilização de alguns padrões de apresentação que facilitam a interação do usuário com o sistema.*

Palavras-chaves: Desenvolvimento Web, Padrões de Projeto para Web, Usabilidade de Software

ABSTRACT. *Standards of design are solutions for existing problems found in software development. In such a way, the standards of design comprise the link between a context, a problem and a solution. This work has the objective of show the efficiency in the application of standards of design for web. In the*

development is assumed to develop two applications, being one with the utilization of standards and other without utilization of standards. The implementation of the applications was done with C#.NET language with orientation to objects and programming in layered conciliated with HTML5 and CSS3 technologies, using the IDE Visual Studio Express 2012. With the utilization of standards, it was possible verify a greater code reuse and agility in software development and future maintenance resulting in an increased productivity and quality in the systems. With respect to the users there was a great benefit demonstrating a superior usability and accessibility of the system. With that, the standards of design can generate greater viability for build of software. Finally, the software demonstrates utilization of some standards of presentation that facilitate user interaction with the system.

Keywords: Web Development, Standards of design for Web, Usability Software.

1 INTRODUÇÃO

Historicamente, o surgimento dos padrões de software se deu com Christopher Alexander, arquiteto e urbanista nascido em Viena, Áustria, “que reconheceu o fato de um conjunto de problemas recorrentes ser encontrado toda vez que um edifício era projetado” (PRESSMAN,2011, p. 316).

De tal modo, os padrões de projetos compreendem o vínculo entre um contexto, um problema e uma solução, para que se entenda melhor a situação em que o impasse em questão se encontra.

Busca-se com este estudo demonstrar as vantagens da utilização dos padrões de projetos de Web, além de também demonstrar sua influência no cotidiano das pessoas, gerando comodidade e melhoria na qualidade de vida destas e maior coerência dos projetos de *software*.

2 REFERENCIAL TEÓRICO

2.1 História do software

Um fator importante para o desenvolvimento de tecnologias tem sido o software, que tem facilitado e simplificado vários processos. Além disso, tornou-se uma característica fundamental que as organizações devem ter para que suas gestões se tornem mais eficientes.

2.2 Processos de software

Define-se processo de software como “um conjunto de atividades, ações e tarefas realizadas na criação de algum produto de trabalho (*work product*)” (PRESSMAN, 2011, p.40).

O processo de software permite integrar usuários e desenvolvedores, visando que a criação do software ocorra de forma planejada e gerenciada, haja vista que determina precisamente o procedimento a ser utilizado.

Dentre os processos de software pode-se destacar inúmeros modelos. Contudo, este estudo irá se ater aos dois processos, conforme passa a expor.

2.2.1 O modelo cascata

Este modelo é o mais antigo dentre os existentes, possui como principal característica a sequência de atividades.

2.2.2 O modelo espiral

O modelo em espiral foi proposto por Barry Boehm em 1988, o qual incorporou as melhores características dos modelos existentes à época a novos elementos, visando melhorar o desenvolvimento de softwares.

Os processos de desenvolvimento de software foram muito importantes, mas verificou-se a necessidade de melhorar ainda mais as técnicas de construção deste. Sendo assim, surgiu a necessidade da utilização de padrões.

2.3 Padrões de Desenvolvimento

A ideia de padrões foi apresentada por Alexander (1977) e possuía como base o contexto de arquitetura de prédios e cidades.

Cada padrão descrevia um problema que ocorria repetidamente no cotidiano das pessoas, havendo assim para cada problema, um padrão que continha a solução.

O padrão de projeto se caracteriza como uma regra de três partes que expressa uma relação entre um contexto, um problema e uma solução - por Alexander (1979).

O foco principal do uso dos padrões de projeto no desenvolvimento de software é o da orientação a objetos. Como os objetos são os elementos determinantes em projetos orientados a objetos, a parte mais difícil do projeto é a separação de um sistema em objetos. É complicado, pois, existem vários

fatores como encapsulamento, granularidade, dependência, flexibilidade, desempenho, evolução, reutilização, etc. que interferem na decomposição, geralmente de forma conflitante.

Com o uso de padrões de projeto é possível identificar aspectos comuns entre duas soluções diferentes para um mesmo problema usando o mesmo padrão. Conhecer esses aspectos nos permite desenvolver soluções cada vez mais eficientes e melhores que podem ser reutilizadas, fazendo com que o conhecimento humano progrida cada vez mais.

2.4 Padrões para Web

A Internet vem se popularizando cada dia mais e se tornado indispensável em todos os setores de uma sociedade. Porém a Web nem sempre foi assim como é atualmente.

A fase Pré Web é marcada pela tentativa de tornara Internet útil, mas os serviços de informação oferecidos eram em pequenos espaços e os usuários navegavam restritos a eles. Além disso, todas as soluções eram proprietárias e as plataformas não eram abertas.

Na Web 1.0 todas as páginas eram estáticas, ou seja, o conteúdo não podia ser alterado pelo usuário, sendo a página somente de leitura. O grande trunfo dessa geração era a grande quantidade de informação. A Web 1.0 foi a primeira geração comercial da Internet.

A Web 2.0 é conhecida como dinâmica e colaborativa, pois o usuário pode interagir e deixar de ser um mero espectador. Esse é o modelo usado atualmente.

A geração Web 3.0, também conhecida como Web Semântica, ainda não existe, mais se propõe a suceder a Web 2.0 e trazer novas formas de interação e de conexão de informações. (LAUDON, 2010)

Padrões de projetos Web consistem em soluções geralmente regulamentadas por algum órgão, como por exemplo, o W3C (*World Wide Web*).

2.4.2 Padrões W3C

A missão do W3C é levar a *World Wide Web* a um formato padrão, desenvolvendo protocolos e diretrizes que garantam seu crescimento e qualidade.

O W3C desenvolve especificações técnicas e orientações através de um processo projetado para maximizar a consenso sobre as recomendações, garantindo qualidades técnicas e editoriais, além de transparentemente alcançar apoio da comunidade de desenvolvedores, do consórcio e do público em geral. Desde sua fundação, o W3C publicou mais de 110 de padrões, denominado *Recomendações do W3C Web Standards – Padrões Web*.

2.4.3 Padrões de Interface

Os padrões de interface têm por objetivo de proporcionar uma experiência mais simples e fluída para o usuário, podendo melhorar a usabilidade e interação do software.

Segundo (Pressman, 2011) nos últimos anos foram propostos centenas de padrões de interface do usuário, mas a maior parte deles cai em dez categorias.

As dez categorias são: *TopLevelNavigation*, *CardStack*, *Fill-in-the-Blanks*, *SortableTable*, *BreadCrumbs*, *EditInPlace*, *SimpleSearch*, *Wizard*, *ShoppingCart* e *ProgressIndicator*.

2.4.3.1 Padrão *TopLevelNavegation*

Este padrão é usado quando o site ou uma aplicação implementa uma série de funções fundamentais. Apresenta um menu de alto nível, geralmente acoplado a um logo ou imagem identificadora, que permite a navegação direta para qualquer uma das principais funções do sistema.

2.4.3.2 Padrão *CardStack*

O Padrão *CardStack* é empregado quando uma série funções inferiores ou categorias de conteúdo específicas relacionadas com um recurso ou função deve ser selecionada em ordem aleatória. Compõe uma pilha de fichas indexadoras, cada uma delas selecionável com um clique de mouse e cada qual representando funções inferiores ou categorias de teor específico.

2.4.3.3 Padrão *Fill-in-the-Blanks*

Este padrão possibilita que dados alfanuméricos sejam inseridos em uma “caixa de texto”.

2.4.3.4 Padrão *SortableTable*

O padrão *SortableTable* demonstra uma extensa lista de registros que podem ser ordenadas por meio da seleção de um mecanismo comutador para qualquer rótulo de coluna.

2.4.3.5 Padrão *BreadCrumbs*

Este padrão fornece um trajeto de navegação completo quando o usuário está trabalhando com uma hierarquia de páginas ou telas complexa.

2.4.3.6 Padrão *EditInPlace*

Aprovisiona o recurso de edição simples para certos tipos de conteúdo no local em que é exibido. Nenhuma necessidade de o usuário inserir explicitamente uma função de ou modo de edição de texto.

2.4.3.7 Padrão *SimpleSearch*

Proporciona a aptidão de pesquisar em um site ou fonte de dados constantes para um dado simples descrito em uma *string* alfanumérica.

2.4.3.8 Padrão *Wizard*

Este padrão visa dirigir o usuário, passo a passo, por meio de uma tarefa complexa, orientando para que haja a finalização da tarefa por meio de uma série de telas de janelas simples.

2.4.3.9 Padrão *ShoppingCart*

O Padrão *ShoppingCart* provê uma lista de itens selecionados para compra, elencando informações de itens, quantidade, código de produto, disponibilidade (disponível/esgotado), preço, informações para entrega, custos de remessa e outras informações de compra relevantes. Além disso, oferece a capacidade de edição (por exemplo. Remover, modificar quantidade). (Welie, 2013)

2.4.3.10 Padrão *ProgressIndicator*

Proporciona uma indicação do processo quando uma operação leva mais que N segundos. Isso se dá pela representação em forma de ícone ou animação ou em uma caixa de mensagens contendo alguma indicação visual (por exemplo, um “poste de barbeiro” girante, um controle deslizante com um indicador da porcentagem completada) de que o processamento está em andamento. Também pode conter uma indicação de teor de texto do estado do processamento. (Tidwell, 2013)

3 RESULTADOS E DISCUSSÃO

Foi desenvolvido um sistema de e-commerce para demonstrar a usabilidade de todos os padrões. Desenvolveu-se o mesmo sistema sem a utilização de padrões.

O projeto foi desenvolvido em camadas, sendo que as de negócio e de dados se mantiveram para ambos os sistemas, uma vez que, os padrões utilizados são padrões de interface.

A análise comparativa entre os dois sistemas, demonstrou os seguintes resultados em relação a utilização dos padrões.

A aplicação do padrão *TopLevelNavigation* resultou em uma maior interação do usuário com o sistema, uma vez que o mesmo pode acessar os principais recursos do sistema sem complicações.

A utilização do padrão *CardStack* proporcionou maior compreensão do usuário, posto que, este padrão utiliza abas para conteúdo específico. Assim, o usuário terá acesso ao conteúdo de forma rápida.

Com o uso do padrão *Fill-in-the-Blanks* a comunicação do usuário com o sistema se torna interativa e prática, fazendo com que o usuário tenha total interação de seus dados com o sistema.

O padrão *SortableTable* proporcionou maior comodidade e facilidade para o usuário, haja vista que permite que o mesmo ordene seus dados da melhor forma possível.

O uso do padrão *BreadCrumbs* demonstrou que o usuário não fica perdido no sistema, uma vez que o padrão exibe toda hierarquia de páginas que foram navegadas.

A aplicação do padrão *EditInPlace* resultou em uma edição mais rápida e fácil, sendo que o usuário pode fazer uma alteração com apenas um duplo clique na região a ser editada.

O padrão *SimpleSearch* proporcionou um rápido acesso por um conteúdo específico no sistema evitando assim, a perda de tempo do usuário.

A utilização do padrão *Wizard* resultou em uma forma simplificada de tarefas complexas no sistema, evitando assim, o cansaço do usuário devido a atividades extensas.

O padrão *ShoppingCart* demonstrou a facilidade do usuário com relação a suas compras, uma vez que este pode visualizar todos os produtos que foram selecionados para a mesma, bem como alterá-los ou excluí-los.

O uso do padrão *ProgressIndicator* proporcionou ao usuário maior clareza sobre suas ações, uma vez que este mostra ao usuário uma estimativa do status do seu processo.

4 CONCLUSÃO

Foi possível verificar que a utilização de padrões de projeto possibilita um desenvolvimento de sistemas de melhor qualidade deixando o sistema com uma interface mais interativa e eficiente facilitado a usabilidade do usuário, gerando assim uma comodidade maior ao usuário.

Conhecer esses padrões e aplicá-los de forma correta facilita e minimiza o trabalho dos desenvolvedores de software.