**IDADE FEDERAL DO PIAUÍ - UFPI**

**Centro de Educação Aberta e a Distância – CEAD/UFPI**

**CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**MONOGRAFIA**

**Sistemas de informação para o combate ao desperdício de alimentos na empresa delta engenharia.**

**Nome do Aluno (sem abreviação)**

**GILBUES – PIAUÍ**

**2016**

**NOME DO ALUNO (sem abreviação)**

**SISTEMAS DE INFORMAÇÃO PARA O COMBATE AO DESPERDÍCIO DE ALIMENTOS NA EMPRESA DELTA ENGENHARIA.**

Monografia submetida ao Curso de Bacharelado de Sistemas de Informação como requisito parcial para obtenção de grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Xxxxxx

Tutor: Prof. Xxxxx (Caso possua tutor orientador no polo)

**GILBUES – PIAUÍ**

**2016**

**SISTEMAS DE INFORMAÇÃO PARA O COMBATE AO DESPERDÍCIO DE ALIMENTOS NA EMPRESA DELTA ENGENHARIA.**

Nome do Aluno (sem abreviação)

Monografia aprovada / reprovada / aprovada com ressalva como exigência parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

.

Cidade – PI, XX de XXXXXX de 2013

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof, XXXXXXXXXX

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof, XXXXXXXXXX

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof, XXXXXXXXXX

MODELO DE DEDICATÓRIA (opcional)

Aos meus pais Penilton e Bucetilda pelo eterno incentivo.

MODELO DE FOLHA DE AGRADECIMENTOS (opcional)

**AGRADECIMENTOS**

Ao nosso Orientador Prof. [NOME DO ORIENTADOR] pelo incentivo, simpatia e presteza no auxílio às atividades e discussões sobre o andamento e normatização desta Monografia de Conclusão de Curso.

Especialmente a Professora [NOME DA PROFESSORA]. pelo seu espírito inovador e empreendedor na tarefa de multiplicar seus conhecimentos, pela sua disciplina nos ensinando a importância do trabalho em grupo e pela oportunidade de participação em publicações, eventos e exercícios simulados.

Aos demais idealizadores, coordenadores e funcionários da [NOME DA UNIVERSIDADE] do [NOME DO ESTADO].

A todos os professores e seus convidados pelo carinho, dedicação e entusiasmo demonstrado ao longo do curso.

Particularmente ao Prof. [NOME DO PROFESSOR], por sua vocação inequívoca, por não poupar esforços como interlocutor dos alunos e por suprir eventuais falhas e lacunas.

Aos colegas de classe pela espontaneidade e alegria na troca de informações e materiais numa rara demonstração de amizade e solidariedade.

A nossa querida revisora [NOME DA REVISORA], sem a qual nossa Monografia não teria a mesma qualidade.

Às nossas famílias pela paciência em tolerar a nossa ausência.

E, finalmente, a DEUS pela oportunidade e pelo privilégio que nos foram dados em compartilhar tamanha experiência e, ao frequentar este curso, perceber e atentar para a relevância de temas que não faziam parte, em profundidade, das nossas vidas.

MODELO DE FOLHA DE EPÍGRAFE (opcional)

*“Ubi societas ubi comunicatio:*

*Onde existe sociedade, existirá*

*a comunicação humana.”*

*José Cretella Júnior*

Introdução

Sistemas de Informação[[1]](#footnote-1) é a expressão utilizada para descrever um Sistema seja ele automatizado (que pode ser denominado como Sistema Informacional Computadorizado), seja manual, que abrange pessoas, máquinas e/ou métodos organizados para coletar, processar, transmitir e dissemina dados que representam informação para o usuário e/ou cliente.

A literatura tem salientado a importância dos Sistemas de Informação (SI) e das Tecnologias de Informação e Comunicação (TIC), em geral, para a estruturação, funcionamento e competitividade das empresas, bem como para a definição dos seus produtos e serviços e dos seus modelos de negócio, nomeadamente do sector da hotelaria (Connolly e Haley, 2008).

No presente trabalho será abordado a aplicação de um sistema de informação – denominado Hotel5+ - em uma empresa do setor hoteleiro da cidade de Corrente – PI. O sistema é desenvolvido em linguagem php com o auxilio do framework CodeIgniter.

**DESENVOLVIMENTO DE SOFTWARE**

Este capítulo apresentará um pouco sobre as duas linguagens utilizadas para a construção do sistema, o HTML e o PHP. Além disso irá abordar sobre a utilização de CSS com o auxílio do framework *Zurb Foundation* 6.0[[2]](#footnote-2).

Será apresentado o Padrão de Projeto ou, *Desing Patterns*[[3]](#footnote-3), MVC junto ao conceito de framework web. Por fim um pouco sobre o MySQL, utilizado para o armazenamento de dados e a metodologia de desenvolvimento SCRUM, que possibilitou a gestão do projeto.

Ao longo das seções deste capítulo diversas referencias serão feitas, através de imagens do código do próprio sistema, de forma a exemplificar a utilização de cada linguagem, modelo ou banco de dados.

**O DESENVOLVIMENTO DE SOFTWARE**

Nos primeiros anos da Computação, o software era projetado sob medida  
para cada aplicação e desenvolvido pelo próprio programador ou organização. O  
programador que desenvolvia era o mesmo que colocava em funcionamento e  
consertava os erros quando ocorriam (PRESSMAN, 1995).

Com o avanço da tecnologia do hardware, softwares complexos e de  
grande porte passaram a serem desenvolvidos, entretanto vários projetos  
importantes que sofriam enormes atrasos, depois de implantados, eram difíceis  
mantê-los. Muitos não eram confiáveis, seus custos superavam as previsões e  
detinham um desempenho insatisfatório (SOMMERVILLE, 2007).

Iniciou-se, então, a crise do software que era consequência da indisciplina  
que dominava o mundo do desenvolvimento do software. Necessitava-se de que  
esse mundo fosse disciplinado, obtendo-se então a conclusão de que o processo de  
software precisava passar pelos processos da engenharia (PRESSMAN, 1995,  
2011).

Em 1968, a OTAN (*Organização do Tratado do Atlântico Norte*) realizou  
uma conferência sobre Engenharia de Software, a NATO (*North Atlantic Treaty  
Organization*) *Software Engineering Conference*, em Garmisch, conforme ilustrado  
na FIG.1, na Alemanha, com o principal objetivo de estabelecer melhores práticas  
para o desenvolvimento. Esse encontro é considerado como o nascimento da  
Engenharia de Software (LUIZ, 2010).

A engenharia de software é uma tecnologia em camadas, cujo foco na  
qualidade é o que sustenta a engenharia de software, sendo assim, a principal  
camada.

**ENGENHARIA DE SOFTWARE**

A Engenharia de Software é a área da computação que estabelece uma abordagem sistemática de desenvolvimento de software com qualidade envolvendo processos, técnicas e ferramentas apropriadas para uma ampla gama de aplicações, considerando prazos, restrições e recursos disponíveis (PRESSMAN, 2011).

A criação da Engenharia de Software surgiu no intuito de contornar a crise do software, dando um tratamento de engenharia ao desenvolvimento de sistemas complexos caracterizados por um conjunto de componentes abstratos (estrutura de dados e algoritmos) encapsulados na forma de procedimentos, funções, módulos, objetos ou agentes e interconectados entre si, compondo a arquitetura do software, devendo ser executados em sistemas computacionais (WIKILIVROS, 2016).

**ARQUITETURA DE SISTEMAS WEB**

Uma arquitetura bem elaborada auxilia no gerenciamento de complexidade do software trazendo diversos benefícios como: redução de tempo e custo de desenvolvimento e na manutenibilidade do mesmo, além de facilitar a combinação de abordagens e reutilização de código (Padrões de Projeto e Orientação a Objetos), estando diretamente relacionado a atributos de qualidade, enfatizando-se os requisitos não funcionais, relacionados à segurança, desempenho entre outros. A arquitetura está diretamente ligada ao sucesso de um sistema e conduz à evolução do software.

A arquitetura de um software consiste na definição de seus componentes, as propriedades externamente visíveis destes elementos e os relacionamentos entre eles, enfatizando a separação dos interesses (BASS, 2003).

**A LINGUAGEM PHP**

PHP é uma linguagem de script embutida no HTML. Muito da sua sintaxe é emprestada de C, Java e Perl com algumas características específicas do PHP juntas. O objetivo da linguagem é permitir que desenvolvedores web escrevam páginas geradas dinamicamente de forma rápida.



Figura 1 - Logotipo do php

O PHP como é conhecido hoje, é na verdade o sucessor para um produto chamado PHP/FI. Criado em 1994 por Rasmus Lerdof[[4]](#footnote-4), a primeira encarnação do PHP foi um simples conjunto de binários *Common Gateway Interface (CGI)*[[5]](#footnote-5) escrito em linguagem de programação C. Originalmente usado para acompanhamento de visitas para seu currículo online, ele nomeou o conjunto de scripts de "*Personal Home Page Tools*" mais frequentemente referenciado como "PHP Tools." Ao longo do tempo, mais funcionalidades foram desejadas, e Rasmus reescreveu o PHP Tools, produzindo uma maior e rica implementação. Este novo modelo foi capaz de interações com Banco de Dados e mais, fornecendo uma estrutura no qual os usuários poderiam desenvolver simples e dinâmicas aplicações web, como um livros de visitas. Em Junho de 1995, Rasmus liberou o código fonte do PHP Tools para o público, o que permitiu que desenvolvedores usassem da forma como desejassem. Isso permitiu - e encorajou - usuários a fornecerem correções para bugs no código, e em geral, aperfeiçoá-lo [4].

A versão 4.0 surgiu no ano de 199, a qual incluía uma reescrita do núcleo funcional da linguagem, desenvolvido pelos dois programadores israelenses e nomeada de *Zend Engine*. Dentre outros, o objetivo dessa reescrita foi o aumento da eficiência da linguagem.

A versão 5.0 foi lançada no ano de 2004 trazendo a segunda versão do Zend Engine com um novo modelo de objeto, chamado Zend Engine II.

Os desenvolvedores que mantêm a linguagem, decidiram pular do PHP 6 para o 7, pois muitas alterações presentes na versão 5.6 já representam o PHP 6, desta forma atualmente está disponível apenas a versão 7 Preview da linguagem. O php 7.0 foi lançado em 2015.

**SEGURANÇA NO PHP**

PHP é uma linguagem poderosa e um interpretador, seja incluído em um servidor web como um módulo ou executado separadamente como binário CGI, é possível acessar arquivos, executar comandos e abrir conexões de rede no servidor. Essas propriedades fazem qualquer coisa executando em um servidor web inseguras por padrão. PHP é desenhado especificamente para ser uma linguagem mais segura para escrever programas CGI que Perl ou C, e com a escolha correta de opções de configuração em tempo compilação ou de execução, e práticas corretas de programação, ela pode dar a combinação exata de liberdade e segurança que você precisa [5].

Os principais problemas de segurança em aplicações php são *SQL Injection, Cross Site Scripting (XSS) e Cross Site Request Forgeries (CSRF).* Aseguir faremos uma breve descrição sobre cada uma das vulnerabilidades:

* *SQL Injection* (injeção SQL): é um tipo de ameaça de segurança que se aproveita de falhas em sistemas que interagem com bases de dados via SQL. A injeção de SQL ocorre quando o atacante consegue inserir uma série de instruções SQL dentro de uma consulta (query) através da manipulação das entradas de dados de uma aplicação.
* *Cross Site Scripting:* é um tipo de vulnerabilidade do sistema de segurança de um computador, encontrado normalmente em aplicações web que activam ataques maliciosos ao injectarem client-side script dentro das páginas web vistas por outros usuários.
* *Cross Site Request Forgeries* é um tipo de ataque informático malicioso a um website no qual comandos não autorizados são transmitidos através de um utilizador em quem o website confia. Em contraste com o *cross-site scripting*, que explora a confiança de um utilizador para um site particular, o CSRF explora a confiança que um website tem do navegador do usuário.

**O DESENVOLVIMENTO DE APLICAÇÕES PRA WEB MODERNA**

Atualmente o desenvolvimento de aplicações para a web vem da necessidade de baixo acoplamento à sistemas que dependem de hardwares específicos para rodar, ou seja, atualmente dá-se ênfase a aplicações mobile e portáveis idealizadas para *cloud computing* e web. Tendo em vista que o ambiente corporativo demanda uma série de exigências que são diferentes das necessárias ao mundo dos usuários domésticos, tais como hardwares dedicados para armazenamento de grandes bases de dados, recursos de processamento específicos para determinadas tarefas, e – uma das mais importantes de todas – a necessidade de os recursos computacionais da empresa estarem disponíveis para o usuário independentemente de sua localização geográfica e o seu hardware, a computação em nuvem se refere essencialmente à idéia de utilização, independente da plataforma ou lugar, de variadas aplicações por meio da internet com a mesma facilidade de tê-las instaladas em nossos próprios computadores (ALECRIM, 2008).

A Computação em Nuvem - ou *Cloud Computing* - pode ser definida como “um conjunto de recursos como capacidade de processamento, armazenamento, conectividade, plataformas, aplicações e serviços disponibilizados na Internet”. (TAURION, 2009, p. 2).

O tipo de computação em nuvem aqui proposto é o SAAS. SaaS (Software como Serviço) - É um tipo de computação em nuvem onde o sistema/software é oferecido em forma de serviço ou prestação de serviços. O software roda remotamente em um servidor na web. Não é preciso instalar nada na maquina do cliente, basta conectá-lo pela internet. Neste tipo de serviço, paga-se um valor periódico como se fosse uma assinatura somente pelos recursos que utilizar e/ou pelo tempo de uso. Os serviços do tipo SaaS mais comuns no mercado são Google Docs, Gmail e SalesForce. Esse sistema se reflete na ideia de um sistema que rode totalmente na web, sem a obrigatoriedade de um hardware especifico para o mesmo.

Atualmente, ao falar-se em sistemas web, logo vem em mente a ideia de um sistema que possa tanto ser utilizado em computadores, tv’s, tablets e celulares, o que é na verdade complexo e que pode ser resolvido com o a utilização do designe responsivo.

O Web Design Responsivo (Responsive Web Design - RWD) permite sair das limitações de um browser desktop e seu tamanho previsível possibilitando construir páginas com flexibilidade que suportem todo tamanho de tela, qualquer tipo de resolução e interfaces com *touch screem*[[6]](#footnote-6) ou mouse.

O conceito foi criado em 2010 no artigo “Responsive Web Design” escrito por Ethan Marcotte (2010) no blog “A List Apart ”. O autor propõe que, em vez de desenvolver um design para cada dispositivo, deveria ser projetado um único código que adaptasse o layout para as diferentes telas, por meio de tecnologias padronizadas (HyperText Markup Language [HTML] e Cascading Style Sheets [CSS]) (SILVA, 2014).



Figura 3- Designe responsivo

Esse encargo de configuração fica a cargo do CSS. Uma das técnicas mais utilizadas na construção de sistemas web responsives é empregando a técnica do Mobile First (ou “Móvel Primeiro”) é uma metodologia/filosofia/estratégia criada por Luke Wroblewski que estabelece que, no desenvolvimento no qual o design responsivo é levado em conta, deve-se primeiro planejar para dispositivos móveis — e, somente depois, projetar, gradualmente, para dispositivos maiores. Em termos mais simples: do menor para o maior. (ZEMEL 2013).

No presentente tabalho foi empregado o framework Zurb Foundation, que é um framework css para criação de paginas responsivas. O foundation tem a vantagem de ser leve e fácil de ser integrado aos sistemas web. Sua inicialização vem com a linkagem de seus arquivos de configugaracão que são:

* Foundation.css é o arquivo completo com as configurações de css.
* App.css é o arquivo onde se estiliza o css à necessidade do projeto.

Dentro da pasta js há a pagina vendor, e nela esta localizado os arquivos de java script utilizadaos pelo foundation:

* foundation.js
* jquery.js
* what-input.js

O framework inicia sua estilização responsiva com um grid de 12 colunas (row) que pode ser dividida em diversas outras colunas à nescidade do projeto. Para que as mesmas fiquem responsivas é nescessario definir uma escala e um tamanho usando uma das terminologias

* small
* médium
* large

seguida do tamanho e a identificação de colunas como mostra a imagem a baixo:

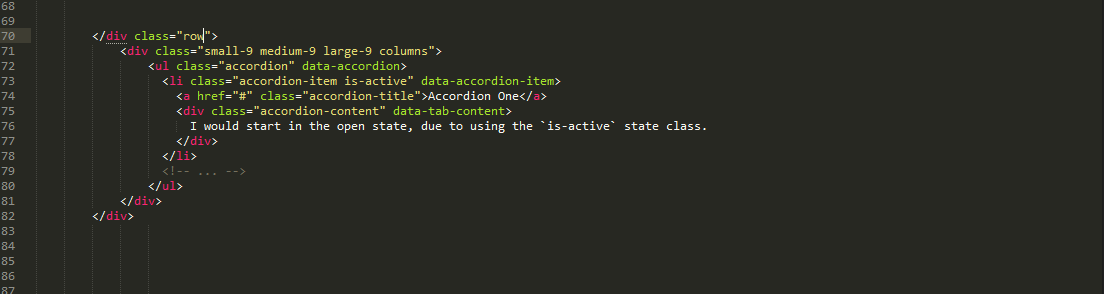


Figura 4 - Configuração de classes responsivas do foundation

O emprego de frameworks css possibilita uma grande economia de tempo no desenvolvimento front-end, pois diminui o estresse de criar gigantescos blocos de códigos para adaptação à dispositivos.

Um framework (ou arcabouço), em desenvolvimento de software, é uma abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica. Um framework pode atingir uma funcionalidade específica, por configuração, durante a programação de uma aplicação (SILVA, 2014).

No sistema desenvolvido, os dados das transações bem como dos funcionários ficam armazenados em uma base de dados MySQL.

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês *Structured Query Language*) como interface. É atualmente um dos bancos de dados mais populares, com mais de 10 milhões de instalações pelo mundo (SILVA, 2014).

Os Sistemas Gerenciadores de Bancos de Dados(SGBD) são usados em muitas aplicações, enquanto atravessando virtualmente a gama inteira de softwares de computador. O SGBD é o método preferido de armazenamento/recuperação de dados/informações para aplicações multiusuários grandes onde a coordenação entre muitos usuários é necessária. Até mesmo usuários individuais os acham conveniente, entretanto, muitos programas de correio eletrônico e organizadores pessoais estão baseados em tecnologia de banco de dados standard.

**CODEIGNITER (MVC)**

O CodeIgniter é um framework para PHP que segue o padrão MVC (model, view, controller) que atende perfeitamente as necessidades da orientação a objetos. Foi criado por Rick Ellis que PE fundador da empresa Ellislab a qual é responsável pelo desenvolvimento do framework.

Bibliografia

Connoly, D., & Haley, M. G. (2008). Information Technology Strategy in the Hospitality

Industry. In B. e. Brotherton, The SAGE HAndbook of Hospitality Management.

Thousand Oaks: SAGE.

1. São dados oferecidos de uma forma significativa e útil para os indivíduos. Dados são correntes de fatos brutos que importam eventos que estão ocorrendo nas organizações ou no ambiente físico, antes de terem sido organizados e arranjados de uma forma que as pessoas possam entendê-los e usá-los. Fonte: <https://pt.wikipedia.org/wiki/Sistema_de_informa%C3%A7%C3%A3o> [↑](#footnote-ref-1)
2. Framework web-based. [↑](#footnote-ref-2)
3. Padrão de projetos [↑](#footnote-ref-3)
4. É um programador canadiano-dinamarquês. Ele é o autor da primeira versão da linguagem de programação PHP. Rasmus foi co-autor das versões seguintes do PHP, juntamente com os israelitas fundadores da Zend Technologies, Andi Gutmans e Zeev Suraski. Fonte: <https://pt.wikipedia.org/wiki/Rasmus_Lerdorf> Acesso: 10/07/2016 [↑](#footnote-ref-4)
5. CGI (sigla em inglês para Common Gateway Interface), consiste numa importante tecnologia que permite gerar páginas dinâmicas, permitindo a um navegador passar parâmetros para um programa alojado num servidor web. Fonte: [https://pt.wikipedia.org/wiki/**CGI**](https://pt.wikipedia.org/wiki/CGI) Acesso: 10/07/2016 [↑](#footnote-ref-5)
6. A tela sensível ao toque é um display eletrônico visual que pode detectar a presença e localização de um toque dentro da área de exibição, por meio de pressão. O termo refere-se geralmente ao toque no visor do dispositivo com o dedo ou a mão, que também podem reconhecer objetos, como uma caneta. [↑](#footnote-ref-6)