

STANFORD UNIVERSITY
SISTEMAS DE INFORMAÇÃO

RELATÓRIO FINAL DE ESTÁGIO
SUPERVISIONADO

Linus Torvalds

Varginha – 2016



DECLARAÇÃO

Declaramos para os devidos fins, que Linus Torvalds, portador do CPF nº 085.726.371-88, RG nº MG-32.556.032, realizou estágio supervisionado no Setor de Desenvolvimento, no período de 01 de Dezembro de 2008 a 23 de Novembro de 2016, totalizando uma carga horária de 11.500 (Mil e quinhentas) horas.

Varginha – MG, 23 de Novembro de 2016.

Larry Ellison
ANALISTA DE SISTEMAS

RELATÓRIO DE ESTÁGIO

Aluno:	Linus Torvalds
Curso:	Sistemas de Informação
Ano de conclusão do estágio:	2016
Empresa:	ORACLE CONSULTORIA E ASSESSORIA
Local / Estado:	Varginha – MG
Setor Estagiado:	Desenvolvimento
Período do Estágio:	01 de Dezembro de 2008 a 23 de Novembro de 2016
Total de Horas de Estágio:	11.500 horas

APRESENTAÇÃO

O trabalho realizado durante o estágio foi de fundamental importância para a minha formação acadêmica, pois proporcionou uma excelente oportunidade para a aplicação prática das teorias estudadas durante o curso de Sistemas de Informação.

No Capítulo I é destacado a Web 2.0.

O capítulo II caracteriza a empresa ORACLE CONSULTORIA E ASSESSORIA, local onde o presente estágio foi realizado. Esta é uma empresa reconhecida na cidade no ramo de prestação de serviços de consultoria e assessoria, portanto foi importante vivenciar seus processos. A área escolhida foi o desenvolvimento de software para a web.

No Capítulo III, mais precisamente no item 3.1 é realizada uma revisão teórica dos fundamentos do desenvolvimento web. As fontes pesquisadas são atuais e tratam do assunto de forma clara e precisa. Em seguida, no item 3.2 são descritas as atividades que foram desenvolvidas no decorrer do estágio.

No Capítulo IV são apresentadas as conclusões relativas ao estágio de uma forma geral e são apresentadas também sugestões à Faculdade e à empresa que proporcionou o estágio.

No Capítulo V são apresentadas todas as referências bibliográficas consultadas na realização do trabalho.

E por fim, são apresentados os anexos no capítulo VI.

Nesse período, a principal limitação encontrada, foi a dificuldade de conciliar o estágio às aulas e trabalhos extraclasse, porém esta limitação pôde ser resolvida quando foi estabelecido um cronograma das prioridades diárias. Essa limitação considerada como um desafio mostrou que a organização e o planejamento melhoram o rendimento das atividades e possibilitam melhor aproveitamento do tempo proporcionando tranquilidade para desenvolver as tarefas que necessitam de maior raciocínio e concentração.

Finalizando a apresentação deste estágio, gostaria de agradecer primeiramente a Deus e Maria que sempre estão ao meu lado me dando forças para lutar. Agradeço principalmente aos meus pais e namorada que com muito amor,

paciência e apoio irrestrito me incentivaram e auxiliaram, não só durante o período de estágio, mas durante todos os anos que estou estudando.

Agradeço a todos os meus professores pela dedicação e paciência que tiveram comigo, terei sempre admiração por eles. Agradeço também aos meus amigos de sala, porque nesses anos pudemos conviver e aprender uns com os outros, transformando assim as dificuldades em desafios.

Contribuiu também para o sucesso deste trabalho o analista Larry Ellison que com disposição me recebeu e orientou. Seu apoio foi fundamental para concretização deste trabalho.

Por fim, agradeço ao meu orientador de estágio, o professor Hélio Lemes Costa Junior, principalmente pela paciência e profissionalismo com que me orientou.

SUMÁRIO

I – INTRODUÇÃO	7
II – A EMPRESA	6
2.1 – Histórico.....	6
2.2 – Missão, Visão e Valores.....	6
2.3 – Serviços Prestados.....	7
2.4 – Estrutura da Área Estagiada.....	8
2.4.1 – Organograma e Croqui	9
III – O ESTÁGIO	11
3.1 – Referencial teórico	11
3.1.1 – PHP	11
3.1.2 – MYSQL	12
3.1.3 – PHP e o MYSQL	13
3.1.4 – PHP e o Modelo Orientado a Objetos	13
3.1.5 – Javascript	14
3.1.5.1 – Javascript não é Java	15
3.1.6 – XML	15
3.1.7 – CSS	16
3.1.8 – AJAX.....	16
3.1.8.1 – Funcionamento do AJAX.....	16
3.1.8.2 – A Classe XMLHttpRequest	18
3.2 – Atividades desenvolvidas.....	19
3.2.1 – Biblioteca TTABLE	20
3.2.2 – Biblioteca TNAVIGATOR	24
3.2.3 – Implementação.....	25
IV – CONCLUSÕES E SUGESTÕES	27
4.1 – Conclusões	27
4.2 – Sugestões a Empresa.....	28
4.3 – Sugestões a STANFORD.....	28
V – REFERÊNCIAS BIBLIOGRÁFICAS	29
VI – ANEXOS	30
6.1 – Avaliação do estágio pelo aluno	30
6.2 – Avaliação do estagiário pelo Supervisor na Empresa.....	32

I – INTRODUÇÃO

Na Web 2.0 os aplicativos funcionam pela Internet, não somente instalados no computador, de forma que vários programas, aplicativos estejam instalados e configurados em uma maquina servidora e o usuário possa acessar de qualquer maquina. Os programas online estão cada vez melhores, indispensáveis e populares como editores de texto, publicadores, comunicadores, organizadores e muitos outros.

Com a web 2.0 mudou-se a forma de fazer softwares. Para o melhor funcionamento na Internet, foi necessária a união de várias tecnologias como o Ajax por exemplo que proporcionou para a web uma comunicação cliente/servidor transparente para o usuário com interfaces rápidas e muito fáceis de usar, hoje podemos destacar duas grandes empresas que investem muito em Ajax, o Yahoo e a Google.

O conteúdo dos websites também sofreu uma enorme mudança, a utilização de blogs dando ao usuário a possibilidade de participar do conteúdo de websites sem ter conhecimento em html. Mesmo quando o conteúdo não é gerado pelos usuários, este pode ser enriquecido com comentários e avaliação.

Algumas aplicações Web permitem ao usuário a configuração de conteúdo sob forma de página pessoal, permitindo a ele a filtragem de informação que ele considera relevante.

Até o momento não existe definição sobre o que exatamente é a Web 2.0, e as definições variam de acordo com o entendimento de cada especialista. Esta indefinição também se deve ao fato de a Web 2.0 não ser um objeto, um produto, tampouco uma marca.

Existe uma tendência integrante da Web 2.0 chamada Computação em nuvem, esta tende a levar todas as informações, sistemas operacionais que o usuário utiliza em sua maquina para servidores online de forma que o usuário não tenha nada instalado e armazenado em sua maquina.

A Web 2.0 tem afetado diretamente as empresas, pois é um ótimo diferencial apresentar serviços online aos seus clientes proporcionando agilidade e acesso a informações em tempo real.

II – A EMPRESA

2.1 – Histórico

A ORACLE CONSULTORIA teve início em 2004, quando Amélia Oliveira, vislumbrou novos horizontes na sua vida profissional, convidou 2 amigos o Psicólogo José Donizete da Silva e o Analista de Sistemas Larry Ellison e então iniciaram oferecendo consultorias de Recursos Humanos e de Tecnologia da Informação desenvolvendo soluções para as empresas.

Em outubro de 2005 a ORACLE percebendo que poderia crescer ainda mais oferecendo desenvolvimento em informática e cursos profissionalizantes para adolescentes de níveis sociais menos favorecidos fez parceria com a EDUCARE, participando desta empresa com 70% do capital.

A EDUCARE, formação básica constituída desde 2003 com foco no desenvolvimento de competências e habilidades dentro da área técnico-educacional. E no sentido de agregar valor aos serviços prestados uniram suas forças para atender com objetividade seus clientes. No final de 2007, a ORACLE assumiu por completo as atividades da parte de cursos da empresa EDUCARE, fundando a NETWAY Cursos, para expansão do atendimento em treinamentos e cursos em diversas áreas de tecnologia e profissionalizantes, para a cidade e região. As duas empresas hoje, conta com profissionais altamente competentes para desenvolver o seu negócio.

A ORACLE iniciou pequena, meio tímida, mas hoje, presta serviços para dezenas de empresas no Brasil, tendo seu reconhecimento e respeito no mercado. O crescimento foi deslumbrante e gratificante, pela total responsabilidade de seus empreendedores.

2.2 – Missão, Visão e Valores

Missão: Disponibilizar às empresas e as pessoas um serviço de excelência em consultoria de gestão eficiente em tecnologia, conhecimento e inovação, conquistando a confiança do mercado, trazendo vantagem competitiva para nossos

clientes e o bem-estar de nossos empregados, colaboradores e da comunidade onde atuamos.

Visão: Ser reconhecida como uma empresa de referência na área em que atual e que presta seus serviços de forma diferenciada pela base de conhecimento que dispõe e reutiliza.

Valores:

- Compromisso com a entrega,
- Qualidade na prestação de Serviço,
- Inovação,
- Engajamento,
- Ética,
- Transparência,
- Respeito,
- Foco no desenvolvimento,
- Responsabilidade Social e Ambiental.

2.3 – Serviços Prestados

Constatamos a necessidade de desenvolver soluções em Recursos Humanos, Gestão de Pessoas, Controles informatizados através da Consultoria Estratégica voltada para o dia a dia das organizações, com projetos de curto e médio prazo, no intuito de identificar ferramentas e procedimentos seguros para o gerenciamento na Gestão de Recursos Humanos e os recursos e/ou investimentos em Tecnologia, bem como controles estratégicos gerenciais e alinhados à cultura e ao planejamento das empresas.

Desenvolvemos treinamentos direcionados a necessidade da empresa, atendendo os aspectos: comportamentais, motivacionais, qualidade, conhecimentos técnicos específicos, reengenharia funcional, segurança do trabalho e outros. Para o pronto atendimento temos profissionais qualificados e outras parcerias que promovem ORACLE em todos os seus processos.

Consultoria e suporte em Tecnologia da informação em conformidade com a estratégia da área de TI da empresa, principalmente vindo a otimizar soluções para empresas contratantes de soluções de terceiros.

Aconselhamento/acompanhamento de negociações com fornecedores de soluções de TI.

Suporte e acompanhamento de projetos de TI, principalmente softwares de Gestão e projetos de infra-estrutura.

Acompanhamento e suporte em soluções de problemas de sistemas contratados de terceiros (ERP), assessorando em abertura de chamados (fichas de ocorrências), acompanhamento de atualizações de novas versões, bem como necessidades específicas do negócio da empresa.

Consultoria para aplicação de atualizações de novas versões (pacotes) do ERP e conversões de bases.

Desenvolvimento de customizações específicas de programa para sistemas ERP, para extração de informações, geração de relatórios e planilhas conforme necessidade da empresa, inclusive conversão automática em planilha Excel, diretamente dos relatórios padrões do sistema ERP, facilitando o tratamento da informação (contas à pagar, contas à receber, estoques, etc) e eliminando a impressão em papel, provendo dados para tomada de decisão mais eficiente.

Também desenvolvemos soluções completas Desktop/Web integradas ao ERP para processos especiais, conforme necessidade da empresa (Progress, Webspeed, Delphi., PHP, COBOL, etc).

Temos buscado parceiros no mercado de forma a prover uma solução completa em TI, desta forma A ORACLE consultoria, também atua em projetos completos de infra-estrutura de dados, telefonia de qualquer porte, inclusive com venda dos equipamentos para o projeto.

A ORACLE oferece para empresas e comunidade em sua Escola Netway Cursos, treinamentos e cursos de Informática, profissionalizantes e idiomas.

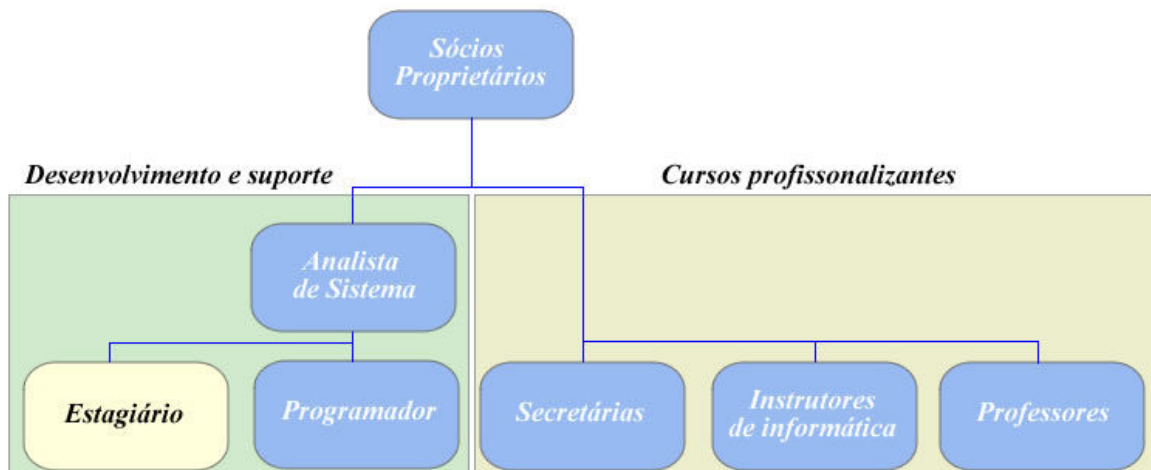
2.4 – Estrutura da Área Estagiada

A empresa é composta de:

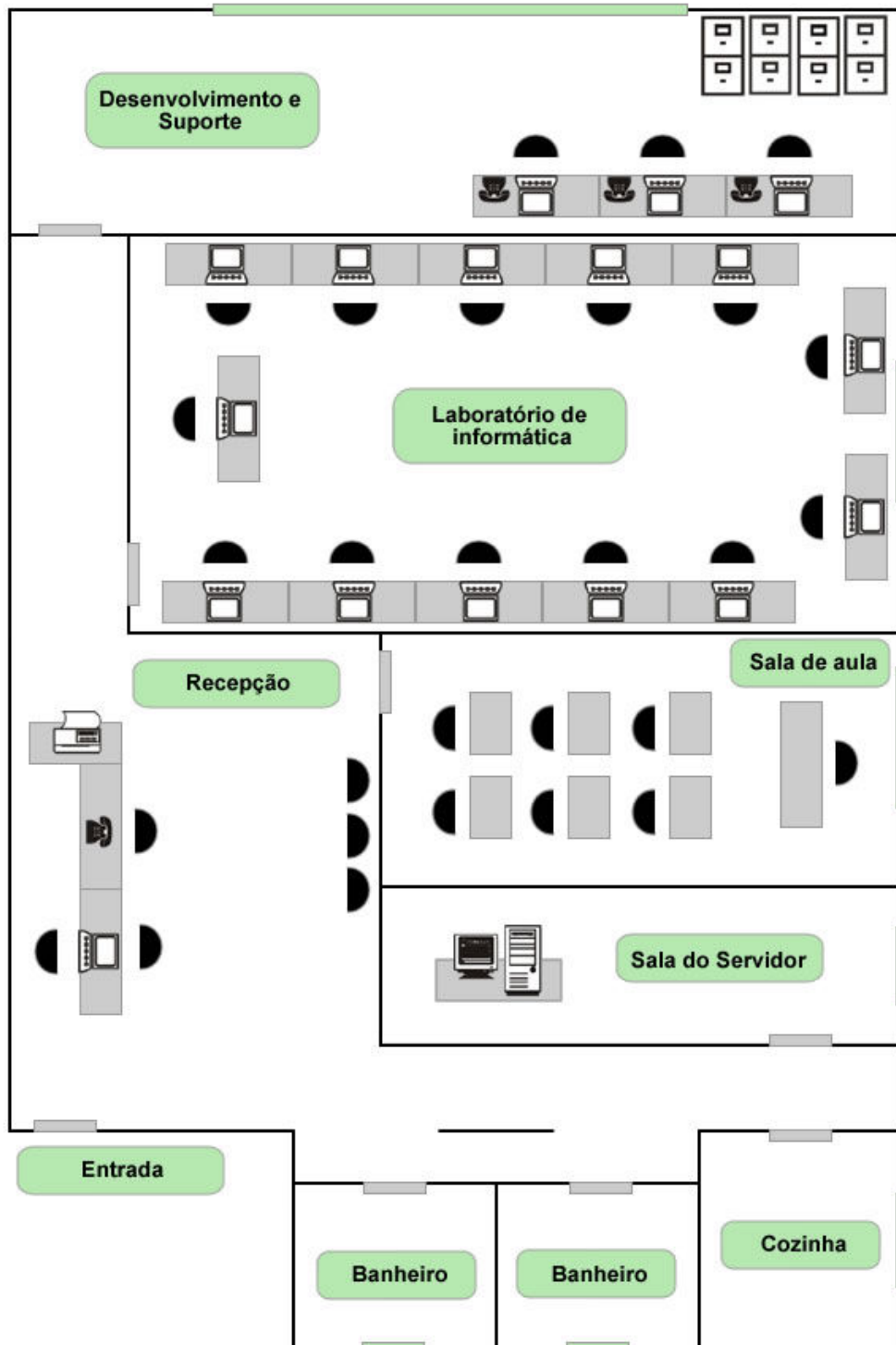
- 3 Sócios Proprietários
- 1 Analista
- 2 Programadores
- 1 Estagiário
- 2 Secretárias
- 2 instrutores
- 1 Professora Parceira de Inglês
- 18 Computadores
- 1 impressora copiadora
- 26 mesas, 29 cadeiras e 2 arquivos.
- 2 linhas de telefone
- 2 banheiros
- 1 cozinha

2.4.1 – Organograma e Croqui

- **Organograma**



- Croqui



III – O ESTÁGIO

3.1 – Referencial teórico

3.1.1 – PHP

PHP (*Hipertext PreProcessor*) é uma linguagem de programação utilizada para criar sites dinâmicos. Aplicações em PHP são geradas em excelente performance, e automaticamente, pelo servidor. O usuário não tem acesso ao código PHP. Esse tipo de detalhe é muito importante quando estamos lidando, por exemplo, com senhas e acesso a banco de dados.

Eduardo Moraz (2005, p.14) diz que:

Atualmente na versão 5.0.5, o PHP foi desenvolvido em 1994 por Rasmus Lerdorf. As primeiras versões foram aplicadas em sua homepage pessoal, com o intuito de saber quem estava consultando seu currículo. A primeira versão a ser utilizada por outras pessoas foi disponibilizada em meados de 1995 e era conhecida como Personal Home Page Tools (Ferramenta para Homepages Pessoais)

O PHP é uma linguagem de script, executada no lado do servidor. Portanto, podemos coletar dados de formulários, gerar páginas com conteúdo dinâmico, enviar e receber dados, gerar xml a partir de dados coletados em banco de dados entre outras funções.

- Scripts no lado do servidor (*server-side*) é a principal área de atuação do PHP. Para implementá-lo, precisamos apenas de um interpretador PHP, um servidor Web e um browser. Basta então rodar o servidor Web conectado a um PHP instalado. É possível acessar os resultados do programa PHP com um browser, visualizando a página PHP por meio de um servidor Web.

O PHP pode ser utilizado na maioria dos sistemas operacionais, como Linux, variantes Unix (Solaris, OpenBSD), Microsoft Windows, etc. Ele também é suportado pela maioria dos servidores Web atuais, incluindo Apache, Microsoft Internet Information Server (IIS), Personal Web Server e muitos outros. (Eduardo Moraz, 2005, p.15).

Com o PHP não estamos limitados a gerar HTML. As funcionalidades do PHP incluem gerações de imagem, arquivos PDF e animações em Flash criadas dinamicamente. Podemos, ainda, gerar padrões de texto, como XML, e enviá-los via HTTP para o usuário. Com o PHP é possível se conectar numa variedade de banco de

dados. Entre os bancos suportados pelo PHP, destacam-se MySQL, Oracle (OCI7 e OCI8), InterBase e PostgreSQL.

Para Luke Welling e Laura Thomson (2005, p.368) “O Protocolo HTTP não tem nenhuma maneira predefinida de manter o estado entre duas transações”, ou seja, o HTTP não fornece uma maneira de dizer se as duas solicitações vêm do mesmo usuário. Com isso a segurança da informação fica comprometida, podendo disponibilizar dados indevidos a um determinado usuário do sistema. Para resolver o problema de segurança na transação cliente/servidor o PHP utiliza o controle de sessão, onde é gerado um ID único de sessão, um número criptograficamente aleatório. Esse ID de sessão é gerado pelo PHP e armazenado no lado cliente. A idéia do controle de sessão é monitorar um usuário durante uma única sessão em um Web site. Podendo facilmente suportar o login de um usuário e mostrar o conteúdo de acordo com seu nível de autorização.

3.1.2 – MYSQL

O MySQL é um sistema de gerenciamento de banco de dados relacional (*relational database management system -RDBMS*) poderoso e muito rápido. Um banco de dados permite armazenar, pesquisar, classificar e recuperar dados de forma eficiente. O servidor do MySQL controla o acesso aos dados para assegurar que vários usuários possam trabalhar com os dados ao mesmo tempo, fornecer acesso rápido aos dados e assegurar que somente usuários autorizados obtenham acesso. Portanto, o MySQL é um servidor multiusuário e multithreadado (*ou multithreaded*). Ele utiliza SQL (Structured Query Language), a linguagem de consulta padrão de banco de dados em todo o mundo.

Para Luke Welling e Laura Thomson (2005) “O MySQL está publicamente disponível desde 1996, mas tem uma historia de desenvolvimento que remonta a 1979. O MySQL ganhou o prêmio Journal Readers` Choice Award Linux em várias ocasiões”.

MySQL está disponível sob a licença Open Source, contanto que cumpra os termos de licença.

3.1.3 – PHP e o MYSQL

Uma das maiores qualidades do PHP e do MySQL é o fato de eles funcionarem com qualquer dos maiores sistemas operacionais e muitos dos menores:

- Com o Linux utilizando o servidor Web Apache.
- Com o Microsoft Windows XP utilizando o Microsoft Internet Information Server (IIS).

3.1.4 – PHP e o Modelo Orientado a Objetos

A linguagem de programação PHP disponibiliza recursos para que a criação de scripts possa ser realizada baseada nos conceitos do modelo orientado a objetos ou escolher entre programação estrutural ou, ainda, uma mistura de ambas.

Em junho de 2004 foi lançada a versão 5 do PHP. O novo modelo de orientação a objeto em PHP, reformulou os construtores e adicionou destrutores. O tratamento de objetos do PHP foi completamente reescrito, permitindo um desempenho melhor e mais vantagens. Enquanto na versão anterior era preciso muito esforço para atender à orientação a objetos, o PHP 5 veio para sanar essa deficiência. A visibilidade de uma propriedade ou método pode ser definida com os seguintes modificadores de acesso: `public`, `protected` ou `private`. Itens declarados como públicos podem ser acessados pelo objeto (*instância da classe*). Membros protegidos estão acessíveis às classes filhas (*herdadas*). A visibilidade privada limita a apenas a classe que define o atributo ou método.

Alguns conceitos básicos do modelo orientado a objetos, segundo Sérgio Luiz Tonsig (2007);

Classes: É um modelo estrutural que possui características semelhantes, que podem ser modeladas e especificadas em uma estrutura.

Tomando como exemplo uma classe “funcionários”, esta terá as características gerais de um funcionário. Cada funcionário é uma instância (objeto) da classe “funcionários”.

Objetos: Objeto é criado (*instanciado*) a partir de uma classe, para que seja possível utilizar-se dos atributos e métodos especificados na classe que representa o objeto.

Atributos: São as características próprias de um determinado objeto. Considere o objeto “pessoa”. Pessoa pode ter nome, endereço, telefone, altura, idade, religião, cor dos olhos etc. Todas estas características específicas de pessoa são chamadas de atributos.

Métodos: Funcionalidades e ações que o objeto pode oferecer. No caso de um objeto “funcionário”, por exemplo, algumas possíveis funcionalidades seriam: admissão, promoção, suspensão, alteração, demissão, afastamento etc.

Herança: Pode chamar de classe filha aquela que herda as características de outra (classe mãe). Herança é uma forma de reutilização de códigos. Por exemplo, que se tenha uma classe “pessoa” e também uma classe “funcionários”. Na classe “pessoa” se define alguns atributos como nome, endereço, telefone e alguns métodos como, por exemplo, cadastrar e consultar. Se a classe “funcionários” tiver como característica tudo que já foi definido para “pessoa”, então, a classe “funcionários” é uma sub-classe de “pessoa” (“pessoa” é a classe mãe). “Herança, implica dizer que é tudo aquilo que foi definido na classe mãe está também disponível (se faz presente) na classe filha”. (Sérgio Luiz Tonsig, 2007).

Encapsulamento: Conceito que abrange à liberação ou restrição de acesso aos atributos e métodos de uma classe, utilizando a propriedade de visibilidade que emprega os chamados “modificadores de acesso” (private, project e public).

Polimorfismo: Segundo Sérgio Luiz Tonsig (2007) “é a Capacidade de respostas diferentes a uma mesma forma de chamada do método. Fato que pode ocorrer pela sobreposição do método”, isto é, a classe mãe possui uma definição e procedimento de um método que, na classe filha o mesmo não é utilizado tal como foi herdado, em função de seus procedimentos terem sido redefinidos.

3.1.5 – Javascript

Javascript é uma linguagem de programação, interpretada e com recursos de orientação a objetos. “O núcleo de uso geral da linguagem foi incorporado no

Netscape, Internet Explore e em outros navegadores Web”. (David Flanagan, 2004). O javascript roda lado cliente e permite que o conteúdo executável seja incluído em paginas Web. Significa que uma pagina Web não precisa mais de HTML estático, mas pode conter programas que interagem com o usuário, controlam o navegador e criam conteúdo HTML dinamicamente.

Sintaticamente, o JavaScript assemelha-se às linguagens C, C++ e Java, instrução IF, o laço while e o operador &&. Entretanto, o JavaScript é uma linguagem não-tipada, ou seja, as variáveis não precisam ter um tipo especificado.

3.1.5.1 – Javascript não é Java

Um dos equívocos mais comuns que se comete sobre JavaScript é sugerir que seja o Java, a linguagem de programação da Sun Microsystems. Exceto por alguma semelhança sintática e pelo fato de que tanto Java como JavaScript podem fornecer conteúdo executável em navegadores Web, as duas linguagens não tem nenhuma relação. Segundo David Flanagan (2004) “A semelhança de nomes é puramente uma manobra de *marketing* (no inicio a linguagem foi chamada de LiveScript; seu nome mudou para Javascript no ultimo minuto)”.

JavaScript e Java formam, porém, uma boa equipe. As duas linguagens possuem diferentes recursos. JavaScript consegue controlar o comportamento e o conteúdo de um navegador, mas não consegue desempenhar operações de rede. Java não tem nenhum controle sobre o navegador como um todo, mas oferece recursos gráficos e recursos de rede. Segundo David Flanagan (2004) “JavaScript do lado cliente consegue interagir com applets Java embutidos em uma pagina Web e as controlar”. Nesse sentido, JavaScript pode realmente empregar Java em Scripts.

3.1.6 – XML

XML (*Extensible Markup Language*) é uma linguagem que veio com a intenção de organizar e formatar conjuntos de códigos aplicados em textos ou a dados. Segundo Ryan Asleson e Nathaniel T. Schutta (2006) “O XML é um formato para a criação de documentos com dados organizados de forma hierárquica, a estrutura de tags do XML é parecida como a do HTML porem ela dá ao programador a capacidade de

criar suas próprias tags”. Um banco de dados pode, através de uma aplicação, escrever em um arquivo XML, e um outro banco distinto pode ler então estes mesmos dados. Hoje temos como exemplo o sistema de nota fiscal eletrônica que troca informações utilizando XML.

3.1.7 – CSS

CSS (*Cascading Style Sheets*) é uma linguagem de estilo utilizada para definir a apresentação de documentos. Ao invés de colocar a formatação dentro do documento, o desenvolvedor cria um link para um arquivo que contém os estilos, procedendo de forma idêntica para todas as páginas do site. Quando o programador deseja alterar a aparência do site basta modificar apenas o arquivo linkado. Segundo Edgard B. Damini (2006) “O CSS veio para botar cada coisa no seu lugar por ser uma linguagem própria de formatação”.

3.1.8 – AJAX

AJAX é um acrônimo para Asynchronous Java Script e XHTML (JavaScript e XHTML Assíncrono). Trata-se de um recurso existente há muito tempo na linguagem JavaScript sendo muito utilizada por algumas empresas como a Google, por exemplo.

3.1.8.1 – Funcionamento do AJAX

Em geral, uma aplicação Desktop apresenta facilidades operacionais e grande velocidade no retorno das informações (*tempo de resposta*), estes aspectos sem a tecnologia AJAX estavam fora do alcance das aplicações desenvolvidas para Web.

A utilização do AJAX torna possível que às aplicações desenvolvidas para Web tenham o mesmo contexto das aplicações desenvolvidas para Desktop, desta forma, a Web deixa de ser uma provedora de conteúdos e passa a ser uma plataforma de aplicativos, o que tem permitido atribuir à Web uma nova categorização (Web 2.0), embora ela própria (estrutura) nada tenha mudado até o momento.

Com o AJAX é possível conseguir em aplicações, dinamismo, recuperação assíncrona de dados, utilizando a classe XMLHttpRequest.

A figura 1.0 a seguir apresenta aspectos dos dois modelos de aplicações em ambientes Web: o modelo clássico e o modelo com AJAX.

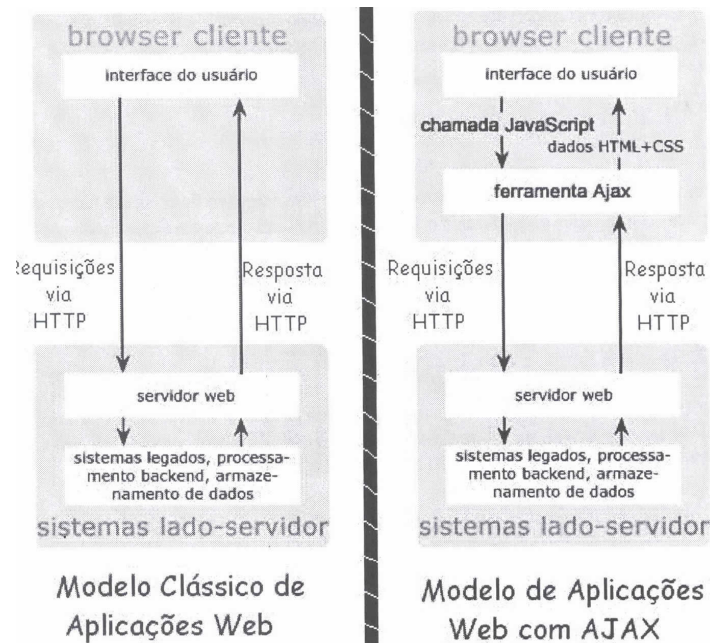


Figura 1 – Comparativo do modelo web clássico e modelo web em Ajax
 Fonte: TONSIG (2007, p.222)

No modelo clássico de aplicações para Web, uma aplicação funciona da seguinte forma:

- Um evento (com o clique em um link) na interface HTML dispara uma solicitação http para um servidor.
- O servidor recebe e verifica o que fazer com a solicitação (faz algum cálculo, acessa o banco de dados) retornando uma resposta em HTML que é recebida pelo browser, que realiza um carregamento (*reload*) da página.

No modelo clássico, após a solicitação ter sido disparada, o usuário fica aguardando um retorno do servidor, durante o tempo em que o servidor processa a requisição, o usuário nada pode fazer na interface. A cada retorno do servidor, toda a interface (página HTML) é recarregada.

No modelo com AJAX as aplicações podem ser constituídas de maneira que a comunicação com o servidor seja transparente ao usuário e também que não tenha que

ficar aguardando retornos que interrompam o funcionamento local do aplicativo, tal como acontece em uma aplicação desktop. O AJAX utiliza uma nova camada de software no lado cliente (engine AJAX) que é escrita em JavaScript, a qual executa processamentos assíncronos de comunicação com o servidor (trocando dados) de forma que um único browser possa realizar mais de uma solicitação ao mesmo tempo ao servidor

A figura 2.0 a seguir apresenta o funcionamento dos dois modelos de aplicações em ambientes Web: o modelo síncrono e o modelo assíncrono.

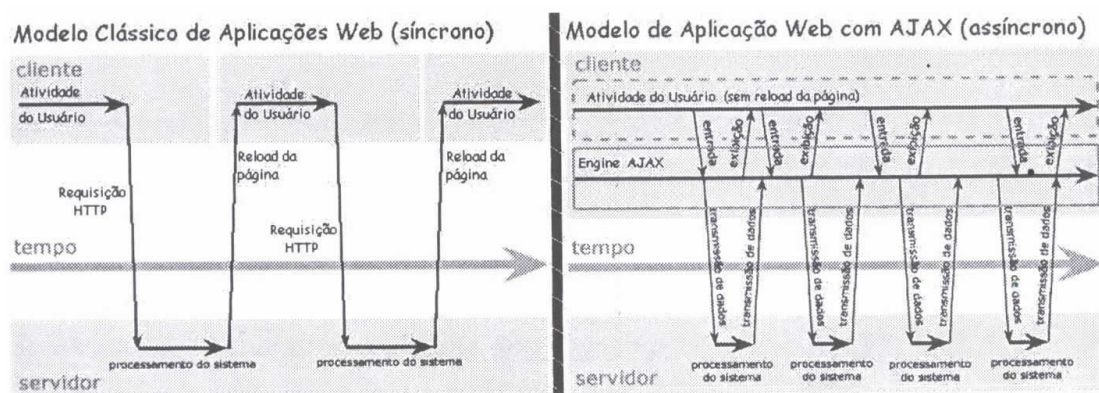


Figura 2 – modelo de comunicação síncrono e assíncrono

Fonte: TONSIG (2007, p223)

Um processamento do tipo síncrono aguarda o término da execução de um processo para só depois iniciar um outro, enquanto um processamento assíncrono permite que outros processos sejam acionados ao mesmo tempo da execução do primeiro.

3.1.8.2 – A Classe XMLHttpRequest

O XMLHttpRequest é uma API (Application Programming Interface), que disponibiliza as funcionalidades HTTP (Hiper Text Transfer Protocol – Protocolo de Transferência de Hiper-Texto) permitindo a troca de dados entre cliente e servidor.

Tosing (2007) diz que:

Inicialmente foi disponibilizada pelo Internet Explore 5.0 (Julho/2000) e depois pelos demais navegadores. Porém, enquanto os demais navegadores seguiram as especificações da W3C, o Internet Explore disponibilizou o

recurso na forma de um ActiveX (Conjunto de tecnologias criado pela Microsoft para facilitar a integração de aplicação).

Portanto, para que o recurso possa funcionar em todas as versões de browsers é necessário observar alguns procedimentos no momento de instanciar a classe XMLHttpRequest.

3.2 – Atividades desenvolvidas

Durante o período de estágio na empresa ORACLE um de seus clientes aqui da cidade de Varginha, solicitou um desenvolvimento de um sistema Web para a gestão de sua empresa. Uma das principais solicitações do cliente era que o sistema pudesse ser acessado de qualquer maquina sem a necessidade de instalar aplicativos locais, ou seja, a única requisição para acesso ao sistema era ter um navegador instalado na maquina. A partir deste ponto juntamente à empresa ORACLE foram estudadas as melhores ferramentas para o desenvolvimento deste sistema. Decidiu-se optar por uma combinação de JavaScript, MySQL e PHP, com essa combinação tinha-se o javaScript atuando lado cliente com total domínio sobre os eventos e interações com o navegador, proporcionando ao usuário um ambiente interativo fazendo com que o sistema ganhe agilidade evitando o excesso de solicitação de dados ao servidor. Com o PHP rodando lado servidor tinha-se aplicações poderosas com uma excelente comunicação com o banco de dados Mysql capazes de gerar javaScripts, Xml, PDF, etc. Além de serem ferramentas open Source elas tem total portabilidade para sistemas Linux, Windows e os principais navegadores como: FireFox, Google Chrome, Internet Explore, Opera.

Após optar por essas três linguagens surgiu um pequeno problema: a agilidade da programação, pois não há boas ferramentas de desenvolvimento de sistemas voltados a PHP e javaScript. Foi aí dedicado o período de estagio: a construção de bibliotecas e classes PHP capazes de gerar rotinas que agilizam a programação reduzindo linhas de códigos e implementadas nestas bibliotecas o conceito de Ajax para tornar transparente a comunicação cliente/servidor.

A seguir tem-se um detalhamento maior sobre as duas principais bibliotecas desenvolvidas: a biblioteca TTable e a TNavigator que juntas tem a capacidade de gerar

telas, criar rotinas lado cliente que interagem com o usuário, manipulam dados locais e rotinas lado servidor que manipulam informações no banco de dados.

3.2.1 – Biblioteca TTABLE

A biblioteca TTABLE é uma ferramenta geradora de código desenvolvida na linguagem PHP, ela auxilia na construção de aplicativos reduzindo grandes linhas códigos em pequenas passagens de parâmetros a partir de técnicas de orientação a objeto. A função da biblioteca é controlar a comunicação da interface do usuário com o banco de dados gerando scripts capazes de atuar lado cliente e scripts pré programados que agem lado servidor.

Lado cliente

Para a comunicação lado cliente são geradas rotinas em javascripts para manipulação da interface com o usuário, criação de eventos (onClick(), OnMouseOver(), etc.), manipulação dinâmica de folhas de estilo css, são gerados scripts em Ajax para a comunicação direta com o servidor fazendo requisições HTTP, nesta comunicação o Ajax faz solicitações de dados ao servidor onde o mesmo retorna os dados no formato XML para que de forma assíncrona o javascript possa atualizar os dados na tela do usuário.

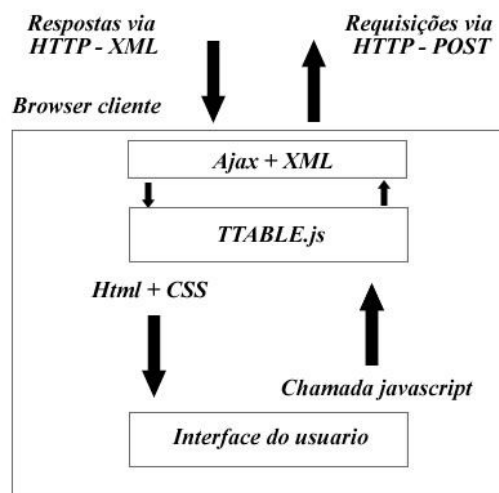


Figura 3 – Comunicação lado cliente

Lado Servidor

Para a comunicação lado Servidor são gerados scripts PHP pré programados capazes de receber dados via HTTP enviados pelo Ajax (lado cliente) e comunicar com o banco de dados fazendo consultas, inclusões, alterações, exclusões, etc. Após atingir um resultado esperado o script PHP cria dinamicamente um arquivo XML e envia para o Ajax lado cliente.

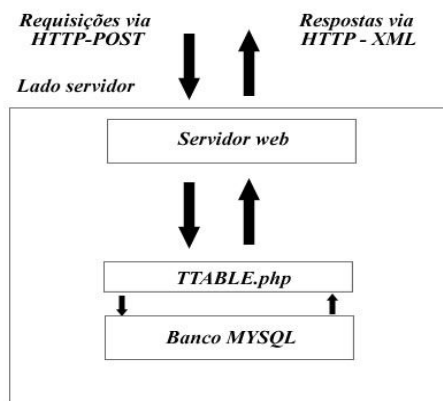


Figura 4 - Comunicação lado servidor

Segurança

Para manter segura a comunicação cliente/servidor a biblioteca utiliza-se de um recurso de sessão que testa a autenticidade do browser lado cliente com o servidor, além disso são criadas variáveis de sessão que contem as configurações e informações da aplicação que foi gerada e passada para o cliente, isto permite com que a cada solicitação do usuário a biblioteca possa validar a autenticidade da informação.

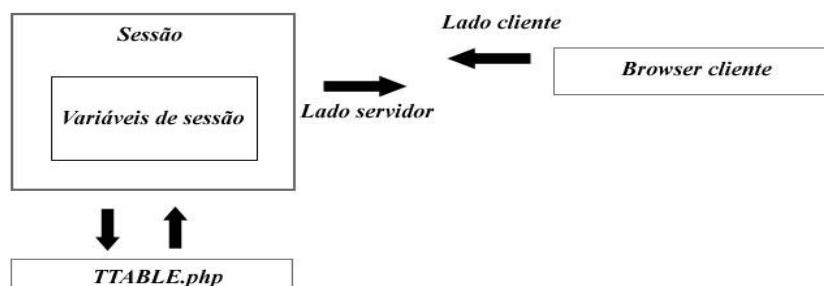


Figura 5 – Controle de segurança baseado em sessão

Funcionalidades

A biblioteca possui diversas funcionalidades como: telas com funções de filtro de dados, telas com funções de pesquisa, telas com funções de localização de registro pela chave primaria, telas com funções de uploads de arquivos, tratamento de erros, exibição de imagens e arquivos pdf.

Tela de Filtragem

A tela de filtragem tem como objetivo disponibilizar ao usuário a capacidade de definir quais os registros da tabela em questão ele ira visualizar e trabalhar, tomando como exemplo um cadastro de “clientes” o usuário poderá escolher trabalhar com apenas clientes da cidade de Varginha. Ao clicar em OK o sistema se encarregara de trazer do servidor os primeiros 50 registros filtrados no formato XML para a maquina do cliente facilitando sua forma de consulta.

Filtrar	
Contem	Intervalo
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>

Ordenar

Código: ☐ Nome: ☐

Figura 6 - Tela de Filtragem

Tela de Pesquisa

A tela de pesquisa tem como objetivo disponibilizar ao usuário a capacidade de listar pequenos grupos de registros. A partir de um pequeno clique sobre um registro listado o sistema carregara o mesmo na tela.

Pesquisar		
Contem		Intervalo
Código	<input type="checkbox"/>	<input type="checkbox"/> <input type="text"/>
Nome	<input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="text" value="silva"/>
Cidade	<input type="checkbox"/>	<input type="checkbox"/> <input type="text"/>

Lista			
I	Código	Nome	Cidade
<input checked="" type="checkbox"/>	fulano	Fulano Silva	Varginha

Ok Cancelar

Figura 7 - Tela de Pesquisa

Tela Ir Para

A tela de Ir Para tem como objetivo disponibilizar ao usuário a capacidade de localizar um registro pela chave primaria e carregar na tela.

Ir Para

Código
10

Ok Cancelar

Figura 8 - Tela Ir Para

Tela de Upload

A tela de upload tem como objetivo disponibilizar ao usuário a capacidade de carregar arquivos que estão em sua maquina para um diretório específico no servidor e registrar no banco de dados o nome e diretório onde o arquivo se encontra, os arquivos permitidos são apenas PDF e JPEG.

Upload

Selecione uma imagem JPG - tam.(190 X 65)Px.

Procurar...

Ok Cancelar

Figura 9 - Tela de Upload

Tratamento de erros - A biblioteca também é capaz de tratar alguns erros de entrada de dados como: validação de CPF, CNPJ, campos que não podem estar em branco, comparação de senhas e validação de data.

Exibição de imagens e arquivos pdf - A biblioteca é capaz de buscar uma imagem ou arquivo pdf que foi carregado anteriormente pela rotina de upload e exibi-los no browser.

Para o correto funcionamento da biblioteca é necessário que se agregue a sua configuração a utilização de uma outra biblioteca chamada TNavigator para que assim o usuário possa ter acesso visual aos botões de navegação e através destes possa disparar eventos.

3.2.2 – Biblioteca TNAVIGATOR

A biblioteca TNavigator é essencial para o funcionamento da biblioteca TTable, sendo responsável pela parte visual dos botões de navegação e pelo direcionamento de eventos que serão disparados pelo usuário do sistema

Régua de Navegação

A régua de navegação é gerada dinamicamente pela biblioteca TNavigator onde são gerados folhas de estilos css para estilizar o visual dos botões e javascripts para controlar o bloqueio e visibilidade dos mesmos. É possível optar pela geração de botões pré-estilizados ou por imagens geradas pelo programador.



Figura 10 – Régua de Navegação

No exemplo acima, demonstrado na Figura10. Foi criada uma imagem específica para cada botão da régua, estas imagens estão em uma pasta, basta o programador configurar o diretório, nome e extensão da imagem, a partir daí a biblioteca se encarrega de realizar o vínculo com a TTable a qual contém toda a programação dos eventos onClick disparados pela régua.

3.2.3 – Implementação

A Figura 11 abaixo demonstra uma tela de cadastro de usuários, onde foi usada as bibliotecas TTable e a TNavigator.

Figura 11 – Tela de Cadastro de Usuários

A capacidade de criação destas bibliotecas se deu ao fato do PHP ser orientado a objeto, nessas foram implementados os conceitos de classes, objetos, herança e polimorfismo assim como descritos abaixo.

Tanto a biblioteca TTable quanto a TNavigator possuem uma classe principal com seus respectivos nomes, esta classe é responsável por todo o processo de utilização e configuração a partir de acesso a métodos e propriedades da classe. A classe TTable possui diversas classes filhas cujo a função é tratar eventos lado servidor. A biblioteca permite ao programador implementar cálculos e eventos fora da classe (com eventos que tratam dados antes de uma gravação no banco de dados) isto se dá ao fato da biblioteca seguir o conceito de polimorfismo, onde ocorre a sobreposição de métodos da classe por métodos programados pelo programador.

Algumas configurações básicas da classe TTable:

- Include("libs/componentes/TTable/TTable.php"); // Fazendo o uso da biblioteca TTABLE na pagina.

- `$tabela1= new TTable('libs/');` // Criando uma instancia da classe TTABLE. O parâmetro “libs” identifica o diretório raiz da localização das bibliotecas no servidor.

- `$tabela1->nome=”cadastroDeUsuarios”;` // Definindo um nome para identificação do objeto. Esta propriedade ajuda na identificação via javascript com php.

- `$tabela1->nomeTabela=”cad_usuario”;` // Vinculando o objeto a uma tabela do banco de dados, basta informar o nome da tabela

- `$tabela1->idCone=1;` // Informando o id de conexão com o banco de dados. O id “1” indica um respectivo nome de banco de dados, usuário e senha de conexão que se encontra em um arquivo de configuração.

- `$tabela1->campos[1]='codigo,,';`

`$tabela1->campos[2]='nome,,';` // Configurando campos da tabela. Esta propriedade é um array onde podemos informar quais os campos o objeto vai tratar e reconhecer.

- `$tabela1->carregar();` // Implementando as configurações. Este método tem a função de implementar todas as configurações realizadas anteriormente, incluindo a geração dos programas que atuarão no lado cliente.

Algumas configurações básicas da classe TNavigator:

- `require_once("libs/componentes/TNavigator/TNavigator.php");`//

Fazendo o uso da biblioteca TNavigator na pagina.

- `$obj=new TNavigator;` // Criando uma instancia da classe TNavigator.

- `$obj->nome='navegador1';`// Definindo um nome para identificação do objeto. Esta propriedade ajuda na identificação via javascript com php.

- `$obj->nomeTabela='cadastroDeUsuarios ';` // Vinculando o objeto TNavigator ao objeto TTABLE, basta informar o valor da propriedade “nome” do Objeto TTABLE.

- `$obj->setRegua(“1,1,1,1,1,1,1,0”);` // Informando quais botões de eventos estarão visíveis. Existem nove botões de eventos, neste exemplo somente o ultimo estará invisível, o parâmetro “1” indica visível e ”0” invisível.

- `$obj->setNavegador(“1,1,1,1”)` // Informando quais botões de navegação estarão visíveis. Existem quatro botões de navegação, neste exemplo todos estarão visíveis, o parâmetro “1” indica visível e ”0” invisível.

\$obj->carregar(); // Implementando as configurações. Este método tem a função de implementar todas as configurações realizadas anteriormente, incluindo a geração das réguas de botões.

IV – CONCLUSÕES E SUGESTÕES

4.1 – Conclusões

Com toda base adquirida durante o curso de Sistemas de informação, principalmente nas aulas de programação e com muitas horas de leituras de apostilas, sites e de livros adquiridos nestes períodos, ao término deste estágio chega-se à conclusão de que ele foi de grande valia para concretizar todo este aprendizado. O principal ponto ao desenvolver estas bibliotecas, foi adequar todo o funcionamento da biblioteca aos conceitos da web 2.0, o que é fundamental para a vida útil de um sistema web. Com este desenvolvimento adquire-se maiores conhecimentos em banco de dados MySql, PHP, JavaScript, Ajax, Xml, Css e na construção de programas que atuam lado cliente/servidor. Com o auxílio do supervisor (Larry Ellison) foi possível acessar conceitos de programas desenvolvidos pela empresa ORACLE e de ERPs consagrados. Através deste conhecimento foi possível além de acertar na definição do modelo de dados, implementar nas bibliotecas interfaces com usuário bem definidas, ergonomicamente corretas e procedimentos seguros de transações com banco de dados, mantendo a integridade e segurança da informação a todo momento e sob quaisquer circunstâncias. Um dos principais benefícios da criação destas bibliotecas foi a redução do tempo de desenvolvimento de novos programas, com elas foi possível reduzir 96% do tempo de desenvolvimento, ou seja, sem a utilização das bibliotecas um programador web que tenha conhecimento em php e javaScript, gastaria em média 25 horas ou mais para o desenvolvimento de um cadastro de clientes com funções de inclusão, alteração, exclusão, filtragem, pesquisa, upload de arquivos e criação de programas lado cliente/servidor. Com a utilização da biblioteca foi possível criar um cadastro de clientes com todos estes recursos em apenas 1 hora e sendo necessários apenas conhecimentos básicos em PHP. Foi um desenvolvimento complexo, completo e bem

trabalhado, porém o retorno deste trabalho virá agora com o desenvolvimento de novos sistemas.

Um desafio futuro é facilitar a implementação destas bibliotecas e outras que venham a ser desenvolvidas, através de telas mais amigáveis e um gerador automático de programas auto-documentados.

4.2 – Sugestões a Empresa

Criar um programa de treinamento de futuros programadores, porque por mais que a faculdade ensine, é na prática que o aprendizado se consolida. É muito importante o futuro profissional já sair da faculdade com este conhecimento da tecnologia aplicada nas empresas.

Outra sugestão é identificar e indicar alguns projetos que poderiam ser executados por grupos de alunos da faculdade. No tempo estagiado, percebi que algumas solicitações de orçamentos, devido a sobrecarga de trabalhos para clientes maiores, não eram possíveis de ser atendidas.

4.3 – Sugestões a STANFORD

Sugere-se a Stanford University criar um horário para simulações de estágio onde os alunos possam apresentar prévias da pasta de estágio, e com isso entender melhor como funciona a avaliação da banca examinadora. Dessa forma os alunos poderiam se sentir mais seguros na hora de apresentar as suas pastas de estágio. É importante também intensificar as aulas que dão um embasamento melhor ao aluno a respeito de práticas cotidianas que possam otimizar a relação entre empresas e clientes.

As orientações poderiam ter um horário pré-estabelecido em alguns dias da semana, para que os professores pudessem ter mais tempo para orientar os alunos. Durante os intervalos das aulas o tempo é muito curto para um orientador dar a devida orientação para vários alunos ao mesmo tempo. A Biblioteca, por sua vez, poderia criar uma lista de livros separados por assunto, que os alunos pudessem consultar, assim ficaria mais fácil encontrar o melhor livro e agilizaria o processo de escolha.

V – REFERÊNCIAS BIBLIOGRÁFICAS

ASLESON, RYAN; SCHUTTA, NATHANIEL T.. **FUNDAMENTOS DO AJAX**. RIO DE JANEIRO: ALTA BOOKS, 2006.

DAMIANI, EDGARD. **JAVASCRIPT: O GUIA DE CONSULTA RÁPIDA**. SÃO PAULO: NOVATEC, 2006.

FLANAGAN, DAVID. **JAVASCRIPT: O GUIA DEFINITIVO**. PORTO ALEGRE: BOOKMAN, 2004.

MORAZ, EDUARDO. **TREINAMENTO PRÁTICO EM PHP**. SÃO PAULO: DIGERATI BOOKS, 2005.

OFICINADANET. **O QUE É XML?**. DISPONÍVEL EM: <[HTTP://WWW.OFICINADANET.COM.BR/ARTIGO/437/O_QUE_E_XML](http://www.oficinadanet.com.br/artigo/437/o_que_e_xml)>. ACESSO EM: 19 OUT. 2016.

TONSIG, SÉRGIO LUIZ. **PHP COM AJAX NA WEB 2.0**. RIO DE JANEIRO: CIÊNCIA MODERNA, 2007.

WELLING, LUKE; THOMSON, LAURA. **PHP E MySQL DESENVOLVIMENTO WEB**. RIO DE JANEIRO: ELSEVIER, 2005.

WIKIPEDIA. **WEB 2.0**. DISPONÍVEL EM: <[HTTP://PT.WIKIPEDIA.ORG/WIKI/WEB_2.0](http://pt.wikipedia.org/wiki/Web_2.0)>. ACESSO EM: 16 OUT. 2016.

WIKIPEDIA. **CASCADING STYLE SHEETS**. DISPONÍVEL EM: <[HTTP://PT.WIKIPEDIA.ORG/WIKI/CASCADING_STYLE_SHEETS](http://pt.wikipedia.org/wiki/Cascading_Style_Sheets)>. ACESSO EM: 19 OUT. 2016.

VI – ANEXOS

6.1 – Avaliação do estágio pelo aluno

Nome	Linus Torvalds				
Curso	Sistemas de Informação	Período	8º	Data de Término do Estágio	23/11/16
Empresa	ORACLE CONSULTORIA E ASSESSORIA				
CNPJ	06.306.877/0001-91				

O estágio tem atendido suas expectativas com relação aos objetivos de sua atuação profissional? Sim.			
A empresa onde está atuando possui estrutura suficiente que possibilite seu desempenho profissional? Sim.			
Seu estágio tem sido orientado, supervisionado de forma adequada? Sim.			
Qual é o cargo de seu Supervisor do estágio na Empresa? Sócio proprietário, Analista de sistemas.			
Seu orientador é solícito quando você pede a sua ajuda para solução de determinado problema?			
EMPRESA	Sim (x)	Não ()	Às vezes ()
STANFORD	Sim (x)	Não ()	Às vezes ()
De modo geral, a orientação recebida foi suficiente?			
EMPRESA	Sim (x)	Não ()	Às vezes ()
STANFORD	Sim (x)	Não ()	Às vezes ()
Como é seu relacionamento pessoal com o Supervisor de Estágio?			
EMPRESA	Ótimo (x)	Bom ()	Regular ()
STANFORD	Ótimo (x)	Bom ()	Regular ()
O estágio que está realizando possibilitou aplicação prática dos conhecimentos adquiridos no curso? (x) SIM () NÃO () EM PARTE OBS: _____			
Seu curso ofereceu embasamento teórico e prático suficiente para a realização das atividades desenvolvidas durante o estágio? () SIM () NÃO (x) EM PARTE OBS: Falta de embasamento teórico para Linguagens Web.			
O estágio permitiu conhecer a organização da Empresa, transmitindo experiências úteis para o futuro do exercício profissional? (x) SIM () NÃO () EM PARTE OBS: _____			
A Empresa proporcionou condições para o desenvolvimento e aproveitamento do estágio? (x) SIM () NÃO () EM PARTE OBS: _____			
A atuação por parte do supervisor de seu estágio na Empresa contribuiu para o bom andamento e aproveitamento do estágio?			

(<input checked="" type="checkbox"/>) SIM (<input type="checkbox"/>) NÃO (<input type="checkbox"/>) EM PARTE OBS: _____
As orientações e/ou informações fornecidas pela STANFORD foram adequadas para a realização do seu estágio? (<input checked="" type="checkbox"/>) SIM (<input type="checkbox"/>) NÃO (<input type="checkbox"/>) EM PARTE OBS: _____
Durante o estágio, você participou de algum treinamento ou curso dentro da Empresa? Se afirmativo, quais? Não.
De 0 a 5, que nota você daria para: (0 – Mínimo 5 – Máximo) 04 Sua dedicação pessoal ao estágio 04 Aprendizagem adquirida no estágio 05 Relacionamento com outros funcionários 05 Relacionamento com o Supervisor de Estágios da STANFORD 05 Relacionamento com o Supervisor da Empresa onde ocorreu o estágio OBS: _____
Você recebe orientação na Empresa antes de executar tarefas específicas? (<input checked="" type="checkbox"/>) SIM (<input type="checkbox"/>) NÃO (<input type="checkbox"/>) EM PARTE OBS: _____
De modo geral, como você avalia os conteúdos aprendidos e apresentados em seu curso e aproveitamento no estágio? (<input type="checkbox"/>) ÓTIMO (<input checked="" type="checkbox"/>) BOM (<input type="checkbox"/>) RUIM (<input type="checkbox"/>) MÉDIO OBS: _____
De modo geral, como você avalia o estágio realizado? (<input type="checkbox"/>) ÓTIMO (<input checked="" type="checkbox"/>) BOM (<input type="checkbox"/>) RUIM (<input type="checkbox"/>) MÉDIO OBS: _____

6.2 – Avaliação do estagiário pelo Supervisor na Empresa

Estágio Supervisionado: _____ Período: _____ Semestre Letivo: ____/____

Aluno	Linus Torvalds				
Curso	Sistemas de Informação	Período	8º	Data de Término do Estágio	23/11/16
Empresa	ORACLE CONSULTORIA E ASSESSORIA				
CNPJ	06.306.877/0001-91				
Supervisor (Empresa)	Larry Ellison				
Cargo	Analista de sistemas				

PARECER DO SUPERVISOR (Empresa/Instituição)		A – 90 a 100 C – 70 a 79 E – 0 a 50	B – 80 a 89 D – 50 a 69
ASPECTOS PROFISSIONAIS			
Itens a serem avaliados:		Conceitos (A, B, C, D ou E)	
1. Qualidade do trabalho: cuidado no manejo de equipamentos, capricho, destreza.		A	
2. Espírito inquisitivo: disposição para aprender, curiosidade científica, capacidade investigativa.		A	
3. Iniciativa e determinação: capacidade para realizar tarefas de estágio sem influências externas, capacidade para solucionar problemas, segurança na execução de tarefas.		A	
4. Conhecimentos: preparo técnico demonstrado no desenvolvimento das atividades.		A	
ASPECTOS PESSOAIS			
Itens a serem avaliados:		Conceitos (A, B, C, D ou E)	
1. Assiduidade: ausências de faltas injustificadas e cumprimento do horário estabelecido.		A	
2. Disciplina e responsabilidade: observância das normas internas relativas ao local de estágio, discrição quanto a assuntos sigilosos, zelo pelo patrimônio, postura adequada.		A	
3. Relacionamento Interpessoal: facilidade de integração com os colegas, integração ao ambiente de trabalho e relação com outros profissionais afins.		A	
4. Cooperação: disposição para colaborar com os colegas e atender prontamente às solicitações das partes envolvidas.		A	
5. Interesse: envolvimento e comprometimento demonstrados para com as Tarefas a serem realizadas e o ambiente de estágio.		A	
Carga horária cumprida: 1.500 Horas			
Observações:			
Supervisor		Carimbo da Empresa	
Local e data:			

STANFORD UNIVERSITY
Bacharelado em sistemas de Informação
CADASTRO DE ESTAGIÁRIO

Estagiário

Nome: Linus Torvalds

Endereço: Rua Eustáquio Júnior, nº 36

Telefone Residencial: (35) 555-1890

Telefone Celular: (35) 555-2086

Curso: Sistemas de Informação

Período: 8º

E-mail: linus@yahoo.com.br

Instituição Provedora do Estágio (Empresa/Instituição)

Nome: ORACLE CONSULTORIA E ASSESSORIA

Endereço: Av. Major Venâncio, nº417 – Sala 2

Telefones: (35) 555-7441

E-mail: larry@oracle.com.br

Profissional Supervisor do Estágio (Empresa/Instituição)

Nome: Larry Ellison

Cargo ou Função: Sócio proprietário, Analista de Sistemas

Telefones: (35) 555-6580

E-mail: larry@oracle.com.br

Área do Estágio: Desenvolvimento

Professor Orientador do Estágio:

Nome: Prof. Hélio Lemes Costa Junior, PhD

Aceite: _____ Data: ____/____/____