**Referencial Teórico**

A Internet surgiu no período da Guerra Fria, cenário em que os Estados Unidos da América e Rússia viviam uma forte tensão política e em constante iminência de embate militar. Em meio a essa crise, havia uma preocupação da cúpula governamental dos EUA de que ocorresse vazamento de informações secretas, situação que deixaria a potência totalmente vulnerável. A fim de impedir tal instabilidade na guerra, os EUA investiram e idealizaram um sistema que permitisse descentralizar as informações a partir de troca e compartilhamento de dados, este modelo concretizou-se na década de 1960; ainda que rudimentar, este pode ser considerado a primeira versão da Internet. Na década seguinte o termo “Internet” foi criado e difundido na comunidade científica, ainda nesse período surgiram os protocolos de comunicação TCP/IP. Em meados de 1994 a internet expandiu por diversos setores (empresas, universidades, domicílios), tornando-se um dos maiores meios de comunicação da atualidade.

Durante o ano de 1992 nasceu a primeira versão da Web, essa versão teve como precursor o cientista Tim Berners-Lee que propôs a criação do World Wide Web, mais conhecido como o “www”, sendo utilizado ainda hoje antes das URLs dos sites. O World Wide Web consistia basicamente em um sistema que disponibilizava documentos no formato de hipertexto, com o intuito de disponibilizar o acesso por diversos usuários simultaneamente. Essa versão foi apenas o primeiro passo para a criação de diversas funcionalidades na Internet, a exemplo das linguagens HTML e CSS.

-

No ano 2000, aproximadamente, a empresa O'Reilly Media desenvolveu o termo Web 2.0, ele faz alusão a forma como a Web é vista e utilizada pelos usuário e desenvolvedores, não se referindo a uma atualização na parte técnica da Web. Concomitante com o advento da Web 2.0 nasceu o entendimento que a Web deve ser desenvolvida da mesma forma de uma plataforma, isto é, a ideia é utiliza-la de forma semelhante a um aplicativo desktop, assim sua velocidade aumentaria e tornaria o sistema mais dinâmica. Por causa dos conceitos trazidos pela Web 2.0, hoje ela tem seu foco direcionado a sites como blogs, wikis, redes sociais, entre outros; essas páginas Web contêm uma grande quantidade de multimídia, além de conteúdo completamente dinâmico. Diante dessas inovações, a Internet que antes era encarada como um ambiente em que se encontrava apenas um acervo de informações, agora é vista como local de interação e participação de diversos usuários simultaneamente.

A próxima versão da Web será 3.0, que atualmente existe apenas em conceito, esse modelo propõe implementar uma rede inteligente de computadores, de forma que os usuários poderão fazer perguntas complexas a máquina e ela será capaz de responder. O projeto visa utilizar redes semânticas por meio de conceitos de inteligência artificial, sendo este um modo de representar o conhecimento.

Com a evolução da Internet surgiram também recomendações para o melhor desenvolvimento de uma aplicação Web, elas têm por objetivo padronizar a forma de programar um site. Salienta-se que apesar de não serem obrigatórias as recomendações, elas são extremamente indicadas para uma boa prática de desenvolvimento.

Nesse capítulo iniciaremos com o conceito de padrões Web, seguindo de uma explicação de forma resumida sobre as tecnologias DOM, HTML, JavaScript, CSS, XML, JQuery e AJAX. Após essas tecnologias serem apresentadas, será exposta a arquitetura escolhida para o desenvolvimento do portal acadêmico implementado neste trabalho. Posteriormente retratar-se-á o conceito de regras de negócios e banco de dados. O último ponto a ser tratado no capítulo são as ferramentas utilizadas no desenvolvimento desse trabalho.

**1.1 Padrões**

Os padrões Web são algumas recomendações do W3C (World Wide Web Consortium), tendo por objetivo criar uma plataforma Web com um amplo potencial de extensão e que possa ser acessada a partir de qualquer dispositivo de forma segura e confiável. Essa plataforma também propõe um ambiente interativo, com numerosa navegação de dados, de forma estável e rápida, além de definir boas práticas de desenvolvimento Web. Na atualidade, apesar do grande potencial de expansão de vários elementos dos padrões Web, os líderes da indústria citam o HTML5 como elemento decisivo nessa plataforma.

**1.1.1 Web Design e aplicações**

Para construção de uma boa aplicação Web é necessário ter conhecimento de normas e tecnologias de desenvolvimento como DOM, HTML, CSS, JavaScript, JQuery, XML e AJAX. Essa tecnologias, permitem a criação de páginas com conteúdo diversificado e dinâmicas

Segue abaixo uma breve descrição das principais.

**1.1.1.1 Document Object Model**

O Modelo de Objeto de Documento (DOM) é uma interface entre a linguagem JavaScript e os objetos do HTML. Essa tecnologia é organizada numa estrutura de árvore de objetos, em que cada nó pode ser manipulado utilizando os métodos desse elemento. Uma das grandes vantagens do DOM é a multiplataforma (compatibilidade com diversas plataformas), ou seja, ele permite que um mesmo documento seja executado em diferentes browsers. Ademais, a manipulação dos objetos da árvore proporciona ao programador criar funções e métodos no JavaScript, estes manuseiam eventos como “click do mouse” ou “pressionar teclado”, gerando, deste modo, um documento altamente dinâmico.

**1.1.1.2 Hyper Text Markup Language**

O HTML é uma linguagem de marcação criada por Tim Berners-Lee, em 1991. Originalmente ela era apenas uma coleção de ferramentas para resolver um problema de compartilhamento de documentos entre instituições. Essa tecnologia surgiu da união de dois padrões HyTime e SGML; quanto ao HyTime (HyperText Markup Language), ele é um conjunto de eventos concorrentes conectados por webs ou hiperlinks; em relação ao SGML (Standard Generalized Markup Language), trata-se de um padrão de formatação de textos.

Após a primeira versão, foi publicada outras versões, como o HTML 3.2 que incorporou tabelas e textos flutuantes ao redor da imagem. No momento atual o HTML possui semântica de apresentação, pois descreve como o dado será estruturado na página. Essa linguagem é utilizada para desenvolver documentos que são páginas Web e podem ser interpretados por navegadores como Chrome, Firefox, Internet Explore e Safari. A última versão lançada foi a do HTML 5, ela possibilita a utilização de recursos que anteriormente somente eram possíveis por meio de outras tecnologias. Além disso, no HTML 5 também foi adicionado novas tags como <audio>, <canvas> e <video>, segue exemplo abaixo. Destaca-se que conquanto a referida versão da linguagem está em fase de implantação pelos navegadores, ela já aparece como uma das principais tecnologias essenciais para os padrões Web.

**1.1.1.3 JavaScript**

JavaScript é uma linguagem de programação criada por Brendan Eich, essa linguagem é executada no lado do cliente da aplicação, ou seja, pelo próprio navegador. Além do mais, ela é uma linguagem baseada em scripts e orientada a objeto. O JS tornou-se bastante popular por conseguir interagir com o DOM das páginas, promovendo sites mais dinâmicos. Algumas características positivas do JavaScript são sua compatibilidade com diferentes dispositivos e browser e a possibilidade de oferecer acessibilidade a deficientes físicos. Em contra partida, um ponto negativo é a falta de segurança, visto que seu código fica acessível a usuários maliciosos.

**1.1.1.4 Cascading Style Sheets**

CSS é uma linguagem de folhas de estilo, sua primeira versão foi criada em 1995 pela W3C para auxiliar no front-end dos sites. Por meio dele é possível estilizar as páginas Web utilizando diferentes cores, fontes e tamanhos, como resultado o layout fica mais atraente. O CSS abre diversas possibilidade de formatação da página, no entanto se utilizado apenas com o HTML ele pode ser limitado e pouco dinâmico. Em contrapartida, se unido a frameworks (como bootstrap) e bibliotecas (a exemplo JQuery) ele se torna extremamente poderoso. Para demostrar o diferencial do CSS em um arquivo HTML foi exposto um exemplo abaixo:

**1.1.1.6 Extensible Markup Language**

Extensible Markup Language é uma linguagem de marcação utilizada para criação de documentos com conteúdo, tal qual o do banco de dados. Por meio do XML é possível organizar a informação utilizando infinitas tags, função que auxilia na leitura e compressão dos dados tanto para humanos, quanto para máquinas. Uma grande vantagem do XML é sua portabilidade, como seu conteúdo é independente de plataformas, ele pode ser criado em um banco e ser lido em outro banco sem nenhum problema de compatibilidade. Logo, o XML é uma forma organizada e fácil de armazenar os dados. Segue exemplo de um XML abaixo:

**1.1.1.7 jQuery**

JQuery é uma biblioteca JavaScript que é compatível com a maioria dos browsers e foi desenvolvido para simplificar a navegação do documento HTML. Com ele é possível manipular eventos, criar animações e desenvolver aplicações AJAX sem a complexidade que antes era necessário. Pelo potencial e facilidade do JQuery, ele se popularizou muito rápido e hoje é utilizada na maioria das páginas Web. As principais vantagens do jQuery são a compatibilidade com a maioria dos navegadores e a facilitação na elaboração dos códigos – alcança-se adequada funcionalidade com códigos compactos, pois a biblioteca implementa grande parte do código e trabalhar com AJAX e DOM.

**1.1.1.8 Asynchronous Javascript and XML**

AJAX, como o próprio nome já sugere, é a união de tecnologias, a mais usual é a união da JavaScript com a XML. Uma das características do AJAX é a possibilidade de executar blocos de código de forma assíncrona. Por meio dessa função as páginas ficam mais interativas e rápidas, e os dados podem ser processados sem necessariamente ter que recarregar toda a página (evita-se, deste modo, processamentos desnecessários de scripts extras). Apesar do XML no nome, não é obrigatório sua utilização, ele pode ser substituído por tecnologias como JSON ou outras linguagens de marcação.

**2.2 Arquitetura da Web**

Para desenvolver uma boa aplicação Web é necessário se preocupar com a arquitetura que será utilizada na construção do sistema, para isso é fundamental definir a quantidade e a funcionalidade das camadas que serão utilizadas. Existem diversos padrões arquiteturais tais como MVC, Pipeline e N-tier, nesse tópico iremos forcar no MVC pois esse será o padrão implementado no portal acadêmico.

O Model-View-Controller (MVC) é atualmente um dos modelos mais utilizados. O seu fluxo de dados acontece da forma que se segue: o usuário clica em um link e é redirecionado ao controller; este executará a lógica daquela chamada e utilizará o model para inseri, consultar e atualizar os dados do banco; após esse processamento o resultado será enviado para a view; nessa camada os dados serão organizados para serem mostrados na tela, a organização acontece por meio de um documento HTML unido a outras tecnologias como o CSS.

A Figura 1.1 abaixo ilustra quais as camadas da arquitetura MVC e quais as conexões existem entre elas:

Verifica-se abaixo as principais características de cada camada da arquitetura MVC:

**a)Modelo (Model)**

Essa camada gerencia os dados da aplicação, ela é responsável por armazenar, excluir e editar os dados que são persistidos no banco de dados. A mesma também é responsável por informar a camada de visão sobre alterações no banco.

**b)Visão (View)**

A camada de visão tem como papel renderizar os dados retornados da camada de modelo, ou seja, ela define a forma que o dado será organizado e visualizado pelo usuário.

**c)Controlador (Controller)**

Na camada Controller fica toda a lógica, regras de negócios e funções do sistema. Essa parte do modelo também é responsável por manipular as requisições feitas pelo usuário da aplicação.

**1.1.2.1 Regras de negócios**

Regras de negócio definem a estrutura e restrições de determinada empresa para fazer negócio. Citando, especificamente, no âmbito de desenvolvimento de software, as regras de negócios tem por objetivo detalhar as funcionalidade e limitações particulares ao software, as quais a aplicação precisa atender. Essas regras auxiliam o programador a desenvolver métodos de tratamento de exceções, particularidade do sistema e limitar ações fora das especificações do projeto. As regras de negócios não são imutáveis, elas dependem das normas da empresa, se uma companhia, por exemplo, alterar suas normas, o sistema, igualmente, deve ajustar suas regras de negócios.

Alguns exemplos de regras de negócio são: a quantidade de disciplinas que um professor pode lecionar em uma universidade, quantos caracteres devem ter na senha e valor que um cliente pode sacar no caixa do banco

**1.1.2.2 Banco de Dados**

Bancos de dados são coleções de dados que estão organizadas e conectados de uma forma logicamente coerente, cada base representa o Universo de Discurso (UD) de uma determinada aplicação. A depender do universo e da aplicação é necessário escolher qual modelo de banco será utilizado, o modelo mais usado é o relacional, esse baseia-se nos conceitos de entidade e relacionamento.

Para administrar esses dados é utilizado um Sistema Gerenciador de Banco de Dados (SGBD), que consiste em um conjunto de programas de computador que tem como objetivo gerenciar, manipular e organizar os dados. Alcança-se tal finalidade utilizando a linguagem Structured Query Language (SQL), alguns exemplos de SQBD são PostgreSQL, Firebird, MySQL, Oracle, SQL-Server, JADE e Microsoft Visual.

Destaca-se que é necessário considerar as propriedades atomicidade, consistência, isolamento e durabilidade (juntas são conhecidas pela sigla “ACID”) com o intuito de executar de forma íntegra uma transação no banco de dados. Abaixo estão descrições de cada propriedade:

* **Atomicidade**: essa propriedade define uma transação como uma unidade atómica, isto é, todas as operações dessa transação devem ser executadas em caso de sucesso, ou nenhuma deve ser executada em caso de falha. O mecanismo garante que quando houver falha não existirá inconsistência no banco por salvar apenas uma parte dos dados enviado. O responsável pela atomicidade é o subsistema de restauração do SGBD.
* **Consistência**: essa propriedade tem por objetivo assegurar que em todos os momentos da transação os dados mantenham sua consistência, sem ferir nenhuma regra de negócio, lógica ou conceito do banco de dados, a exemplo da chave única. Para que essa propriedade seja respeitada é necessário a existência de validações tanto o SGBD, como a aplicação.
* **Isolamento:** o objetivo do isolamento é que cada transação ocorra sem sofrer nenhuma interferência externa de outras transações concorrentes. Objetivando evitar essa influência, não é permitido pelo Sistema de Gerenciamento de Banco de Dados a realização de transações em paralelo. Essa propriedade é importante, pois se existem duas transações a serem executadas a partir da aplicação no SGBD (sendo uma operação de persistir o dado e outra de consultar), então a ordem de execução influenciará no resultado final, fazendo com que a aplicação fique inconsistente. Essa garantia de isolamento é realizada pelo SGBD.
* **Durabilidade:** refere-se a garantia de que o dado está persistido no banco após uma operação de salvar. Dessa forma, há uma certeza de que o conteúdo está armazenado e pode ser consultado posteriormente, sem que desapareça ou seja corrompido. Essa garantia deve ser mantida independente de fatores externos à aplicação, como quedas de energia, travamento ou qualquer outro erro. Por fim, registra-se que uma forma de garantir a durabilidade do dado é através de logs e backups.

**2.3 Ferramentas**

No desenvolvimento de uma aplicação Web são utilizadas diversas ferramentas para a facilitação da criação do projeto. Alguns exemplos de ferramentas utilizadas são: IDE (Integrated Development Environment), servidor Web, frameworks, SGBD e software de diagramas e arquitetura. Ao decorrer desse tópico apresentar-se-á as principais ferramentas utilizadas nesse projeto.

**2.2.1 Ambiente Integrado de Desenvolvimento**

O Ambiente Integrado de Desenvolvimento (IDE) é um software que apoia o programador na construção de aplicações, ele reúne uma série de características e ferramentas que auxiliam o desenvolvedor em diversos aspectos. As principais características e ferramentas encontradas nos IDEs são: 1) editor de texto, o qual suporta diferentes linguagens, tais como Java, Php e Delphi; 2) compilador, ele tem por função transformar a linguagem do programa em linguagem de máquina; 3) e depurador, este auxilia o desenvolvedor encontrar e corrigir os erros do código-fonte. Observa-se que todas essas ferramentas facilitam na construção de programas, transformando o desenvolvimento de aplicações um processo muito mais rápido e eficiente. Alguns exemplos de IDE são: NetBeans, Eclipse e Visual Studio.

**2.2.1 Moqups**

Moqups é uma aplicação Web desenvolvida em HTML5, essa aplicação tem por objetivo criar protótipos de diversos projetos, em diferentes plataformas. Algumas das vantagens da sua utilização são a simplicidade e rapidez na criação de protótipo e a possibilidade de planejar a aparência do site antes de começar a criar seu código fonte. Além disso, por ele rodar diretamente no navegador, não é necessário a instalação de nenhuma ferramenta adicional.

**2.2.2 XAMPP**

O XAMPP é um servidor livre e independente de plataforma que tem seu código fonte aberto. O ‘X’ da sua sigla refere-se a possibilidade de ser utilizado em diferentes softwares e o ‘AMPP’ é uma abreviação de Apache MySQL PHP Perl. Enfim, registra-se que a instalação e utilização do XAMPP é extremamente descomplicada.

**2.2.3 Astah**

O Astah é um software para modelagem de UML (Linguagem de Modelagem Unificada), mas inicialmente seu nome era JUDE, que é uma abreviação para Java and UML Developers Environment (Ambiente para Desenvolvedores UML e Java). Com esse programa é possível criar diagramas de classe, sequência, caso de uso, dentre outros. Esses diagramas são necessários para a unificar o plano de trabalho dos desenvolvedores da aplicação, além de gerar uma documentação com os principais artefatos do projeto.

**2.2.4 PhpMyAdmin**

O PhpMyAdmin é um sistema de gerenciamento de banco de dados desenvolvido em PHP e tem por função administrar o MySQL. Essa ferramenta é gratuita e após sua instalação pode ser acessada através de um browser. A partir do PhpMyAdmin é possível executar diversos comandos no banco, veja-se a seguir alguns deles: criar e remover bases de dados; criar, remover e alterar tabelas; inserir, remover e editar registros das tabelas, executar querys; e importar e exportar as bases.

**2.2.5 Apache**

Apache é um servidor Web compatível com o protocolo HTTP, criado em 1995 por Rob McCool. Esse servidor é responsável por disponibilizar páginas para serem acessadas a partir da Internet. Algumas das possíveis funcionalidade que o servidor permite realizar são: envio de e-mails, compras online e chats. O Apache é distribuído pela licença GNU, ou seja, sua utilização é gratuita e seu código é aberto.

**2.4 Sumário**

Neste capítulo foi falado de forma breve sobre o surgimento da internet e dos padrões e tecnologias que regem ainda hoje a Web. Seguidamente foi apresentado os padrões e tecnologias que serão utilizados na construção do portal acadêmico proposto neste trabalho. Após essa secção falamos da arquitetura utilizada, das regra de negocio, do banco de dados e por fim das ferramentas que serão necessárias durante o desenvolvimento desse trabalho.