

UNIVERSIDADE FEDERAL DE VIÇOSA
CAMPUS DE RIO PARANAÍBA
SISTEMAS DE INFORMAÇÃO

GABRIEL ALVES RESENDE ROCHA

**SISTEMA DE GERENCIAMENTO DE DADOS WEB
VOLTADO PARA INSTITUIÇÕES DE ENSINO**

RIO PARANAÍBA

2022

GABRIEL ALVES RESENDE ROCHA

SISTEMA DE GERENCIAMENTO DE DADOS WEB VOLTADO
PARA INSTITUIÇÕES DE ENSINO

Monografia apresentada à Universidade Federal de Viçosa como parte das exigências para a aprovação na disciplina Trabalho de Conclusão de Curso I

Orientador: Professor Guilherme de Castro Pena

RIO PARANAÍBA

2022

Resumo

Com o surgimento da tecnologia e grandes avanços tecnológicos acontecendo a todo instante, uma tarefa difícil se tornou algo rápido e prático nos dias atuais. Antes uma busca por um documento poderia durar horas, hoje com a implementação de sistemas de gerenciamento de dados, essa mesma busca pode ser realizada em questão de segundos. Nas instituições de ensino isso não é diferente, um sistema de gerenciamento consegue gerar apoio a todas as entidades que precisam buscar, guardar ou alterar informações de maneira rápida, prática e segura. Este trabalho propõe a criação de um sistema de gerenciamento de dados acadêmicos Web, esse sistema permitirá o gerenciamento de todos os dados da instituição e de suas entidades, como atividades avaliativas dos alunos, materiais complementares para as disciplinas, lista de presença, dados de turmas, professores, alunos e outras informações. Com o desenvolvimento e uso desta aplicação em instituições de ensino, espera-se gerar apoio aos professores, alunos e à instituição, almejando também uma melhora no desempenho acadêmico dos alunos.

Palavras-chaves: Gerenciamento de dados, SGBD, Desenvolvimento Web.

Abstract

With the arise of technology and major technological advancements happening, a difficult task became something easier and practical nowadays. A search for a document could last for hours, today with the implementation of data management systems, this same search can be done in a matter of seconds. In educational institutions, this is not different, a management system generates support for all entities which need to seek, save or change information quickly, conveniently and safely. This work proposes the creation of an online academic data management system, this system will allow the management of all data of the institution and its entities, like assessment activities, supplementary materials for disciplines, attendance lists, classes, teachers and students data and other information. With the development and use of this application in educational institutions, it is expected to provide support for teachers, students and the institution, also aiming at an improvement in the academic performance of students.

Key-words: Data Management, DBMS, Web Development.

Lista de ilustrações

Figura 1 – HTML5	11
Figura 2 – CSS3	12
Figura 3 – JavaScript	13
Figura 4 – jQuery	13
Figura 5 – PHP	14
Figura 6 – phpMyAdmin	15
Figura 7 – SGBDR	16
Figura 8 – Apache	16
Figura 9 – XAMPP	17
Figura 10 – Casos De Uso	20
Figura 11 – Plano de Execução	21

Lista de tabelas

Tabela 1 – Cronograma Normal	24
--	----

Lista de abreviaturas e siglas

HTML	Linguagem de Marcação de Hipertexto
CSS	Folha de Estilo em Cascatas
ID	Identificador
PX	Pixel
DOM	Modelo de Objeto de Documento
PHP	Pré-processador de hipertexto
SQL	Linguagem de Consulta Estruturada
SGBD	Sistema de gerenciamento de dados
SGBDR	Sistema de gerenciamento de dados relacional

Sumário

1	Introdução	8
1.1	Objetivo Geral	9
1.1.1	Objetivos Específicos	9
1.2	Organização do documento	9
2	Referencial Teórico	10
2.1	Desenvolvimento Web	10
2.2	HTML5	10
2.3	CSS3	11
2.4	JavaScript	12
2.5	jQuery	13
2.6	PHP	14
2.7	MySQL	14
2.8	phpMyAdmin	14
2.9	Sistema de gerenciamento de banco de dados	15
2.10	Apache	16
2.11	XAMPP	17
3	Trabalhos Relacionados	18
4	Metodologia	20
4.1	Descobrimento das principais funcionalidades	20
4.2	Criação da estrutura e estilização da aplicação <i>Web</i>	21
4.3	Modelagem e criação do banco de dados	21
4.4	Codificação das funcionalidades e comunicação com o banco	22
4.5	Testes	23
5	Cronograma	24
6	Resultados Esperados	25
	Referências	26

1 Introdução

Com as inovações tecnológicas, surgem diversos avanços em todas as áreas que conhecemos hoje. Um desses grandes avanços é a questão do armazenamento, gerenciamento e visualização dos dados. A pouco tempo atrás quase todas as informações que existiam eram guardadas em papéis e similares, porém, com a chegada da tecnologia, diversos softwares que trabalham com dados foram implementados, o que era uma tarefa difícil como a busca de um documento em uma estante ou prateleira, hoje é feita e questão de segundos em um software [C 57c]

Ainda assim, existem diversas instituições que não aderiram ao uso de software para o gerenciamento de informações e continuam utilizando métodos arcaicos, um exemplo disso são escolas, e isso acaba tornando uma simples busca de um documento em um processo demorado e cansativo. [C 11f]

Um sistema de gerenciamento de dados online pode acarretar diversos benefícios, tanto para alunos, professores e outros possíveis responsáveis [Com um sistema assim, é possível acessar as informações acadêmicas em questão de segundos e de qualquer lugar, além de facilitar o gerenciamento dessas informações, outro benefício seria a disponibilização de materiais acadêmicos para complementar os estudos em casa ou até mesmo realizar um revisão para algum possível teste, com esse sistema o responsável do aluno também poderia acompanhar de perto tudo que está acontecendo na vida acadêmica do mesmo.] Em resumo, um sistema desse modelo para um ambiente acadêmico tem a oferecer diversos benefícios para todos que possam utiliza-lo (COELHO; FILHO, 2016).

No desenvolvimento dessa aplicação, será seguida a seguinte ordem metodológica, o primeiro passo é realizar o descobrimento das principais funcionalidades que devem estar contidas no sistema, após isso, espera-se começar a implementação da ferramenta, o segundo passo é a criação da estrutura do site com base nas funcionalidades descobertas e em seguida é feita a estilização do site. [Com a estrutura e estilização do site concluídos, pretende-se seguir para o passo três que é responsável pela modelagem e criação do banco de dado, com essa etapa finalizada, propõe-se o passo quatro onde é realizada a codificação das funcionalidades e comunicação com o banco de dados e por fim será iniciado o passo cinco onde será feita uma série de testes na aplicação para verificar a integridade da mesma e se necessário, será realizada correções no sistema.] longo

Espera-se nesse trabalho, conseguir desenvolver uma aplicação para o gerenciamento de dados acadêmicos [eficaz e eficiente], que possa gerar valor para alunos, professores e instituições de ensino. É esperado que a ferramenta consiga dar apoio a todas as entidades que tenham contato com a mesma, de modo que ambos se beneficiem com a

segundo QUAL BENCHMARK?

aplicação.

1.1 Objetivo Geral

Este trabalho tem como objetivo geral, desenvolver um sistema de gerenciamento de dados acadêmicos Web, que possa ser acessado por qualquer dispositivo que tenha acesso a internet, facilitando o armazenamento e visualização dos dados acadêmicos pelos professores, alunos e responsáveis.

1.1.1 Objetivos Específicos

A aplicação Web desenvolvida neste trabalho, terá os seguintes objetivos específicos:

- Um sistema para o gerenciamento de alunos, professores, disciplinas, turmas e horário de aula que deverá ser acessado somente por um responsável específico;
- Um sistema para que os professores possam gerenciar as faltas e notas de todas as atividades desenvolvidas no decorrer do ano pelos alunos; *E O SAPIENS?*
- Um sistema de armazenamento de materiais acadêmicos das disciplinas;
- Um mural para que os professores possam colocar avisos para os alunos, como provas, feriados, trabalhos, e outros tópicos;
- Uma função para que todos que utilizem a ferramenta possam alterar seus dados como endereço, foto, telefone e outros dados;
- Uma verificação de *login* para acessos em contas relacionadas a professores e responsáveis de gerenciamento.

Requisito FUNCIONAL ≠ OBJETIVO

1.2 Organização do documento

Este trabalho possui seis capítulos que são eles: Introdução, Referencial Teórico, Trabalhos relacionados, Metodologia, Cronograma e Resultados Esperados. No Capítulo 1 (Introdução) é apresentado uma breve indução ao cenário abordado e ao trabalho, além de enfatizar os principais objetivos do referido e como é feita a organização do documento, no capítulo 2 (Referencial Teórico) são apresentadas todas as ferramentas que serão utilizadas para o desenvolvimento da aplicação, no capítulo 3 (Trabalhos Relacionados) são apresentados o estudo de quatro trabalhos semelhantes a esse, no capítulo 4 (Metodologia) mostra como será o desenvolvimento da aplicação, já no capítulo 5 (Cronograma) será apresentado o cronograma de desenvolvimento deste trabalho e por fim, o capítulo 6 (Resultados Esperados) serão apresentados os resultados esperados para este trabalho.

2 Referencial Teórico

Este capítulo aborda todas as ferramentas e metodologias que serão utilizadas para a implementação da aplicação abordada neste trabalho.

2.1 Desenvolvimento Web

Desenvolvimento *Web* é uma área da tecnologia que tem o foco a construção de aplicações voltadas para *Web* e para navegadores. As aplicações *Web* são processadas e acessadas nos navegadores como *Google Chrome*, *Internet Explorer* e todos os demais, a partir de requisições feitas pelo usuário, essas requisições são processadas no servidor trazendo como resposta as páginas *Web*, um exemplo disso seria realizar uma pesquisa no *Google*, a pesquisa seria enviada como uma requisição ao servidor que retornaria como resposta uma página *Web* (FAIN et al., 2014).

Para o desenvolvimento *Web* é necessário a utilização de diversas ferramentas e linguagens que serão citadas nos próximos tópicos deste capítulo.

2.2 HTML5

O *HTML* ou *HyperText Markup Language* (linguagem para marcação de hipertexto), promove a comunicação do desenvolvedor com o dispositivo do usuário. O *HTML* tem suas regras de sintaxe e escrita, e um arquivo que respeite esses fatores é um arquivo de marcação *HTML*, gerando um arquivo HTML final com extensão ".html"(As extensões .htm, .php e .aps também são válidas para arquivos de marcação) (SILVA, 2015). Em geral, o *HTML* é responsável por estruturar todo o site (SILVA, 2014).

No HTML a estrutura de marcação é criada por meio de *tags*. Uma *tag* é constituída por uma ou mais letras e até números, e devem ser escritas entre os sinais '<' e '>' (menor e maior) , para fechar a *tag* é similar a abertura da mesma, é necessário apenas inserir o símbolo '/' após o símbolo de menor. É possível observar um exemplo mais claro na figura 1, onde no lado esquerdo é disponibilizado o código em *HTML* de um botão onde é utilizado a *tag* de abertura '<button>' e a tag de fechamento '</button>' e entre as duas é adicionado o texto que deverá aparecer no botão. Já no lado direito fica visível o resultado gerado por esse código em uma página Web.

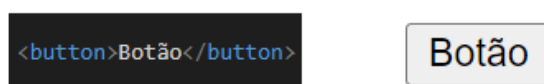


Figura 1 – Exemplo de código HTML5

Fonte: Próprio Autor

2.3 CSS3

CSS ou *Cascading Style Sheet* (folha de estilo em cascata) é uma linguagem utilizada para estilizar as páginas *Web*. [CITE]

Como o *HTML* foi criado exclusivamente para ser uma linguagem de marcação, não cabe a ele estilizar a página, logo é preciso utilizar o *CSS* para realizar as alterações relacionadas a estilo, por isso o *HTML* e *CSS* são extremamente unidos. Enquanto o *HTML* estrutura toda a página, o *CSS* prepara a apresentação dos elementos com diferentes estilos. É possível criar classes, ID ou até mesmo criar o *CSS* dentro do *HTML* (SILVA, 2015).

A regra de estilo do *CSS* é composta por quatro partes: Seletor (é o elemento da marcação *HTML* que no qual a regra *CSS* será aplicada), Declaração (São os parâmetros de estilização), Propriedade (Características que serão estilizadas) e Valor (Quantificação ou qualificação da estilização da propriedade) (SILVA, 2015).

Na figura 2 é apresentando um exemplo de aplicação do *CSS3* no *HTML5*. No lado esquerdo foi disponibilizado o código *CSS* e *HTML*, onde nosso seletor é um ID chamado '#botao', nossa declaração é todo o conteúdo que está entre as chaves, foram utilizadas as propriedades: *width* (largura), *height* (altura), ... e os valores estão localizados entre os símbolos ':' e ';' (Como exemplo o valor da propriedade *width* corresponde a 200px confirme a imagem). Já no lado direito da imagem mostra o resultado gerado por esse código no navegador.

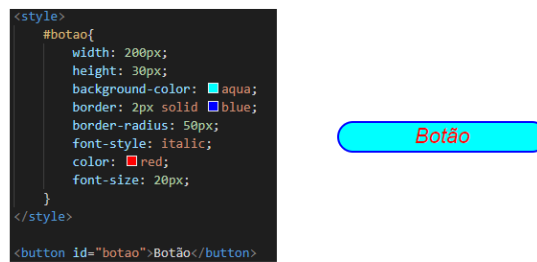


Figura 2 – Exemplo de código HTML5 e CSS3

Fonte: Próprio Autor

2.4 JavaScript

O *JavaScript* é uma linguagem de programação *Web*, onde a maioria dos sites utilizam essa ferramenta e todos os navegadores tem interpretadores *JavaScript*. É uma linguagem de programação de alto nível, dinâmica, interpretada e não tipada (FLANAGAN, 2015).

Com o surgimento do *JavaScript* que é uma linguagem que executa do lado cliente, logo todas as ações são processadas na máquina do usuário fornecendo respostas imediatas, a interação entre o sistema e o usuário final passou a ser mais dinâmica e interativa, com respostas instantâneas e páginas responsivas com efeitos visuais dinâmicos, situação que era impossível de ser reproduzida apenas com *HTML* e *CSS* (FERREIRA; ZUCHI, 2018).

Na figura 3 apresenta um exemplo de aplicação utilizando *JavaScript* e *HTML5* que envia uma resposta imediata após um evento (Nesse caso, o evento seria o clique no botão) ao usuário, sem precisar do recarregamento da página. No lado esquerdo da imagem tem-se a codificação em *JavaScript* que fica localizado entre as tags '`<script>`' e '`</script>`'. Já no lado direito da figura, traz o resultado gerado pelo código em um navegador. Assim que o usuário clicar no botão a frase 'O botão foi clicado' aparece abaixo do próprio botão.

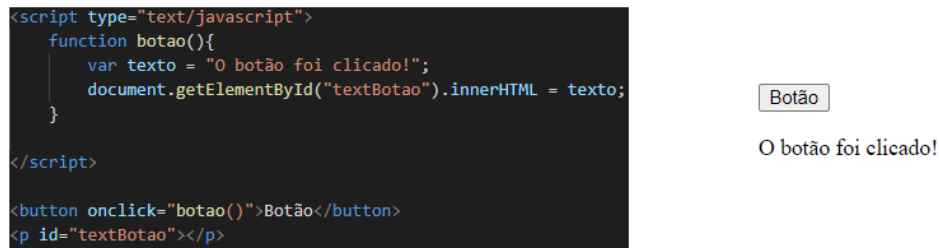


Figura 3 – Exemplo de código JavaScript

Fonte: Próprio Autor

2.5 jQuery

A *jQuery* é uma biblioteca de funções *JavaScript* que nos permite um fácil acesso ao *DOM* (modelo de documento por objeto) *HTML*. Com o *jQuery*, tarefas como: percorrer e manipular documentos *HTML*, tratamento de eventos, animações e outros pontos, ficam muito mais fáceis do que se estivesse utilizando *JavaScript* puro (SILVA, 2013).

Os principais recursos do *jQuery* são: seleção de elementos usando uma sintaxe semelhante à do *CSS*, capacidade de percorrer elementos, manipulação de elementos, tratamento de eventos, efeitos, animações, diversas funções utilitárias e muito mais. Assim como *JavaScript*, o *jQuery* também executa na máquina do cliente, trazendo respostas imediatas sem a necessidade de recarregamento da página.

A figura 4 apresenta um exemplo de aplicação usando *jQuery*, essa aplicação tem a mesma funcionalidade que a aplicação apresentada na figura 3, assim que o evento acontecer, a frase 'O botao foi clicado!' aparecerá abaixo do próprio botão. No lado esquerdo da imagem tem-se a codificação em *jQuery*, já no lado direito, é possível observar o resultado gerado por esse código em um navegador.

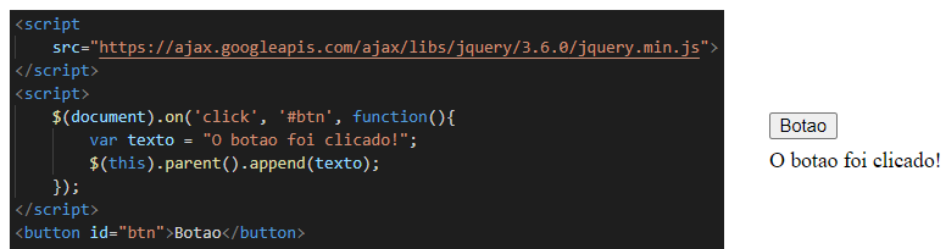


Figura 4 – Exemplo de código jQuery

Fonte: Próprio Autor

2.6 PHP

PHP ou *Hypertext Preprocessor* (Pré-processador de hipertexto) é uma linguagem *back-end* de desenvolvimento Web.

O *PHP* é uma linguagem de programação que executa do lado servidor e pode ser incorporada em *HTML*. A maior parte do que o *PHP* realizada não pode ser vista pelo usuário final, já que após sua execução no lado servidor, o resultado do *PHP* é *HTML*. Todo o acesso ao bando de dados, como conexão, busca, alteração, deletar ou incluir informações, podem ser feitos com o *PHP* (CONVERSE; PARK, 2003).

Na figura 5, é apresentado um exemplo de codificação em *PHP*. No lado esquerdo da figura é possível observar a codificação em *PHP*, no centro tem-se a representação do código em um navegador e a direita da imagem é disponibilizado o código que chega ao navegador do cliente. Com isso é possível notar que o *PHP* é executado no servidor, que envia ao cliente apenas a resposta (Nesse caso foi enviado ao cliente uma *tag* em *HTML* com o conteúdo 'Olá mundo!').

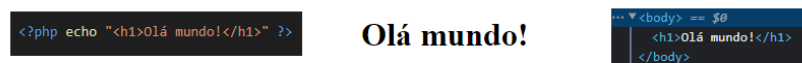


Figura 5 – Exemplo de código em PHP.

Fonte: Próprio Autor

2.7 MySQL

O *MySQL* é um servidor gerenciador de banco de dados relacional, ele atende a todos os tipos e tamanhos de aplicações, ele é código aberto e consegue competir com programas similares de código fechado (MILANI, 2007).

Existem várias vantagens no *MySQL*, fácil portabilidade entre os diferentes sistemas operacionais, suporte completo para integridade referencial, chaves estrangeiras, compatibilidade total com linguagens de programação e diversas outras vantagens (FERRARI, 2010).

2.8 phpMyAdmin

phpMyAdmin é uma ferramenta livre criada para a administração de um banco de dados *MySQL* ou MariaDB. Com o *phpMyAdmin* é possível criar e excluir banco de dados, excluir, alterar, criar e inserir tabelas, editar e adicionar campos, executar

instruções em *SQL*, gerenciar chaves, privilégios e exportar dados em diferentes formatos (CARBONELL, 2018).

Na figura 6, tem-se um banco de dados criado com a ferramenta *phpMyAdmin*, na imagem é possível notar um banco de dados com nome 'novobanco', com apenas uma tabela 'tabela1' onde a mesma contém 4 campos inteiros (campo1, campo2, campo3, campo4). Essa ferramenta disponibiliza uma boa interface que facilita as alterações (exclusão, inserção, alteração e busca) a serem realizadas.



Figura 6 – Interface da ferramenta phpMyAdmin

Fonte: Próprio Autor

2.9 Sistema de gerenciamento de banco de dados

Existem diversos tipos de sistemas de gerenciamento de banco de dados (*SGBD*), mas nesse projeto será utilizado um *SGBDRs*, ou seja, um sistema de gerenciamento de banco de dados relacionais (em um modelo de dados relacional, todos os dados são armazenados em tabelas, onde existem um tipo de relação entre os dados). Um *SGBD* é responsável pelo gerenciamento de uma base de dados, ele é responsável por toda a parte de acesso, manipulação e organização dos dados. Existem diversas vantagens em utilizar um *SBGD*, como: independência de dados (O *SGBD* dispõe uma visão abstrata dos dados ao aplicativo, ocultando detalhes), acesso eficiente (em um *SBGD* existem diversas técnicas sofisticadas para armazenar e recuperar dados de maneira eficiente), administração de dados, acesso de diversos usuário ao mesmo tempo, sistema de recuperação de falhas e diversas outras vantagens (RAMAKRISHNAN; GEHRKE, 2008).

Na figura 7, é apresentado um exemplo de modelagem de banco de dados relacional com duas tabelas. Nessa modelagem é possível notar uma relação entre duas tabelas, que no qual, a tabela 'user' tem uma relação de 1 para 1 com a tabela *address*, logo, um 'user' pode e dever ter apenas um 'address' e vice-versa.

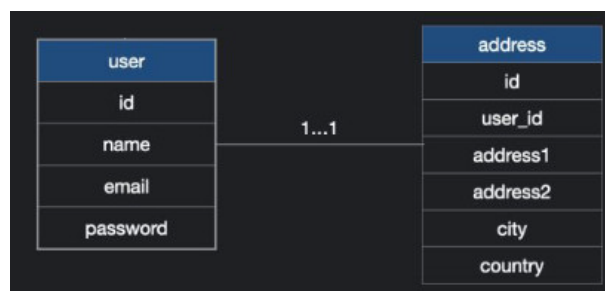


Figura 7 – Exemplo de modelagem de dados relacional

Fonte: Próprio Autor

2.10 Apache (SERVIDOR WEB)

O *Apache* é um servidor de código aberto, ele permite que os conteúdos da internet se mantenham no ar (caso seja um servidor *Apache*), e que possam ser acessados por diversos usuários. O *Apache* atua entre o servidor e as máquinas dos clientes, ele solicita o conteúdo do servidor em cada requisição do cliente, e entrega as repostas obtidas para o navegador. o *Apache* é um software que é executado em um servidor, sua principal obrigação é manter uma conexão entre o servidor e os navegadores como *Google Chrome*, *Internet Explorer*, *Firefox* e os demais navegadores, além de entregar as requisições feitas pelos clientes aos navegadores (LAURIE; LAURIE, 2003).

A figura 8, nos apresenta o funcionamento de uma arquitetura cliente servidor, onde o cliente envia uma requisição ao servidor, essa requisição é processada pelo servidor e por fim o servidor devolve uma resposta ao cliente. Em uma conexão entre clientes por exemplo, essa conexão nunca vai ser direta, sempre ocorrerá uma mediação feita pelo servidor.

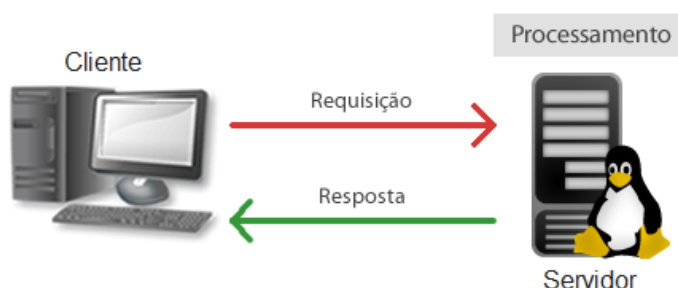


Figura 8 – Arquitetura cliente servidor

Fonte: Próprio Autor

2.11 XAMPP (TECNOLOGIA)

O *XAMPP* é uma ferramenta gratuita e multiplataforma onde contém os principais servidores de código aberto. Ele oferece suporte a diferentes sistemas operacionais (*Windows*, *Mac*, *Linux* e *Solaris*), oferece diferentes tipos de servidores (*Apache HTTP Server*, *FileZilla FTP Server*, *MERCURY Mail Server* e *Tomcat*). Com o *XAMPP* é possível criar facilmente um servidor web, gerando um excelente ambiente para desenvolvimento e testes de aplicações web (FRIENDS, 2019; VALADE; BALLAD, 2011).

A figura 9 apresenta a interface da ferramenta *XAMPP*, que no qual, nos dá a possibilidade de ativar diversas ferramentas e também realizar toda a parte de configuração da mesma.

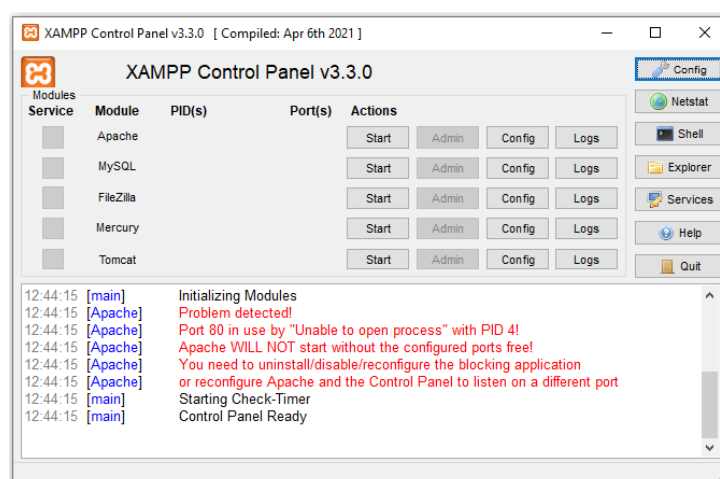


Figura 9 – Interface da ferramenta XAMPP

Fonte: Próprio Autor

3 Trabalhos Relacionados

Este capítulo tem como objetivo identificar os principais trabalhos relacionados com o referido, analisa-los e fazer uma comparação entre eles, destacando seus pontos similares e distintos.

Coelho e Filho (2016) propõe em seu trabalho a criação de um sistema padronizado e de fácil utilização, para conseguir atingir as necessidades de comunicação entre professores e acadêmicos. Essa aplicação em específico tem diversas funcionalidades como cadastro de acadêmicos, matérias, avisos, notas, projetos e trabalhos, todos feitos por professores, enquanto os alunos podem visualizar os dados e configurar sua conta. Diferente desse trabalho, o projeto abordado nesse trabalho será uma aplicação Web e terá uma página específica para cadastro de alunos, professores, matérias e horários, que serão feitas por um responsável específico aos invés de poder ser realizado por qualquer professor, para evitar a criação de divergências nos dados, as outras funcionalidades são bastante semelhantes. ✓

Silva, Neto e Silva (2011) propõem em seu trabalho, a criação de um aplicativo Web para o gerenciamento dos dados de alunos e funcionários de uma escola de música. Nesse sistema os usuários podem visualizar os horários de aulas e outras rotinas existentes no sistema, o funcionários podem matricular os alunos, marcar horário de aulas e outras funções. O projeto abordado neste trabalho tem alguns diferenciais como, um local específico para cadastro de alunos, professores e matérias feitas por um funcionário específico para evitar divergência nos dados, sistema para gerenciar notas e faltas de alunos, e um sistema de mural para avisos dos professores, as outras funcionalidades são semelhantes.

LEITE et al. (2015) propõem em seu trabalho a criação de um aplicativo integrado ao ambiente virtual *Moodle*, para o gerenciamento dos dados acadêmicos. Neste projeto todas as funcionalidades são bastante parecidas com as abordadas neste trabalho, um dos grandes diferenciais notados é a integração com o ambiente virtual *Moodle* apresentado neste trabalho relacionado, entretanto todas as outras funcionalidades (cadastro, alteração e visualização das informações) são semelhantes. ✓

Leal e Lustosa (2013) propõem no trabalho a implementação de uma aplicação web para a gestão dos dados acadêmicos, que tem como principal objetivo armazenar e manusear as informações. Nesse trabalho relacionado quase todas as funcionalidades são parecidas, entretanto existe algumas diferenças entre os dois trabalhos, neste trabalho citado existe uma autenticação para login (que pode ser levada em conta para o desenvolver deste trabalho), já na aplicação abordada neste trabalho existem dois adicionais, existe uma funcionalidade que atua como um mural de avisos onde os professores cria-

ram notícias para que os alunos possam acompanhar e terá uma parte destinada para a disponibilização de materiais complementares as aulas para os alunos, o restante das funcionalidades são basicamente iguais. ✓

px

4 Metodologia

FALTA PREÂMBULO

4.1 Descobrimento das principais funcionalidades

Com o estudo aplicado ao tema, revisões bibliográficas e análise dos trabalhos relacionados, foi possível descobrir os principais pontos a serem desenvolvidos nesta aplicação.

Após os estudos, foi possível perceber que algumas funcionalidades são de extrema importância para o sistema de gerenciamento acadêmico, algumas dessas funcionalidades são: uma funcionalidade para realizar o gerenciamento de alunos, professores, disciplinas, turmas e horários de aula, que devem ser gerenciados por profissionais específicos; uma funcionalidade onde os professores consigam gerenciar as notas e faltas de cada aluno; um sistema de mural onde os professores consigam deixar avisos para os alunos; uma funcionalidade onde os professores consigam armazenar materiais acadêmicos complementares para os alunos; o sistema deverá permitir a configuração das contas feita pelo usuário, e por fim, deverá existir uma verificação para login de professores ou responsáveis de gerenciamento.

A Figura 10, representa o diagrama de caso de uso das principais funcionalidades da aplicação. Nesse diagrama é possível perceber as funções dos atores, tal que: o ator 'Gerenciador Geral' tem como função a gerência de alunos, professores, disciplinas, turmas e horário de aula; o ator 'Professor' tem como função a gerência de materiais acadêmicos complementares, notas das atividades avaliativas, faltas, avisos no mural e também pode visualizar as informações que ele gerencia; já o ator 'Aluno' pode visualizar todas as informações a respeito dele que o ator 'Professor' gerencia; e todos podem configurar a conta.

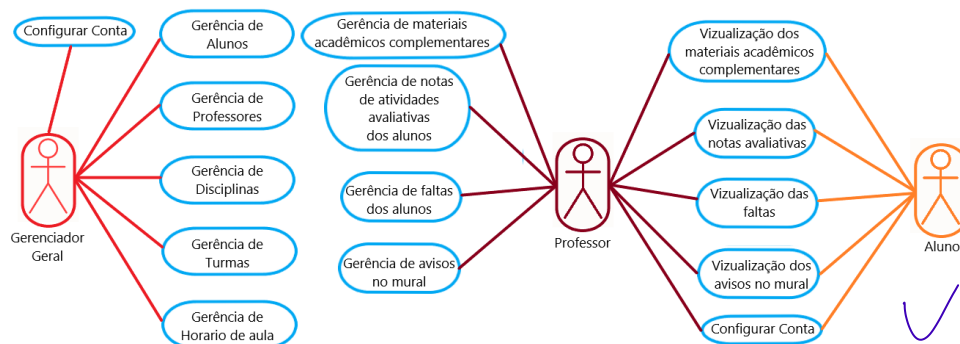


Figura 10 – Caso de uso das principais funcionalidades da aplicação

Fonte: Próprio Autor

Também foi elaborado um plano de execução. A figura 11 representa o plano de execução a ser seguido no desenvolvimento da aplicação, nele contém todas as etapas descritas de forma simplificada para o desenvolvimento da aplicação.

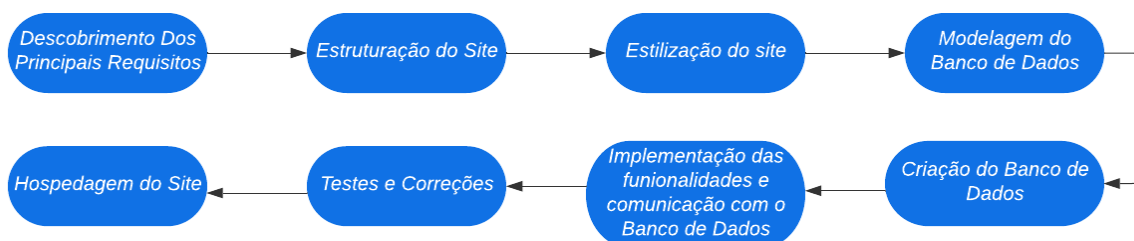


Figura 11 – Plano de execução da Aplicação

Fonte: Próprio Autor

4.2 Criação da estrutura e estilização da aplicação Web

Com base nas principais funcionalidades descobertas, é feita a criação da estrutura da aplicação Web. Nesse processo, o objetivo é estruturar a aplicação, deixando organizado e com uma boa hierarquia para que todos os próximos passos ocorram como planejado. Neste ponto, será utilizada a linguagem de marcação chamada *HTML5* que foi mencionada no Capítulo 2, logo, tem-se toda a marcação e divisão das páginas da aplicação Web.

Com a estrutura da aplicação Web feita, é aplicada uma estilização em todas as páginas da aplicação. Esse processo faz com que a aplicação fique mais apresentável e agradável de lidar, além de organizar um pouco mais os elementos da aplicação. Nesta etapa será utilizada a linguagem de estilos *CSS3*, *JavaScript* e *jQuery* que foram mencionados no Capítulo 2, com isso é gerada uma aplicação bem estruturada, agradável e mais organizada.

4.3 Modelagem e criação do banco de dados

A modelagem do banco de dados é uma parte muito importante, nela é possível estruturar todas as tabelas do nosso banco, criar os relacionamentos entre as tabelas, criar as chaves de cada tabela.

Para que tudo ocorra bem pretende-se seguir o seguinte passo a passo da criação de um diagrama de banco de dados:

1. Descobrimto de todas as tabelas necessárias;

2. Criação de todos os campos de cada tabela;
3. Definição de chaves primárias de cada tabela;
4. Definição do relacionamento entre as tabelas;
5. Definição das chaves estrangeiras com base no relacionamento entre as tabelas.

Para a modelagem do banco de dados, será utilizado a ferramenta LucidChart (LUCIDCHART, 2022) que é gratuita, pode ser acessada pelo navegador e conta com inúmeras funcionalidades para modelagem de banco de dados. Nessa ferramenta, será realizada a criação do diagrama visual de uma maneira fácil e rápida.

Com a finalização da modelagem do banco de dados, pretende-se criar o banco em nosso *SGBD*. Para a criação do banco será utilizada a linguagem de programação *mysql* e o *SGBD phpMyAdmin*, ambos citados no Capítulo 2.

Para que a criação do banco ocorra de acordo com o diagrama elaborado, planeja-se respeitar o seguinte passo a passo:

1. Realizar a criação do banco de dados;
2. Criar as tabelas que não possuem dependência de outras tabelas primeiro;
3. Na criação das tabelas sem dependências, definir também os atributos e chaves primárias;
4. Criar o restante das tabelas seguindo a ordem correta de dependências para que não ocorra erros;
5. Lembrar de definir os atributos, chaves primárias e chaves estrangeiras na criação das tabelas restantes;
6. Após isso, futuramente pode ser criado *Functions* e *Stored Procedures* para acelerar a utilização do banco.

4.4 Codificação das funcionalidades e comunicação com o banco

Com a estrutura do site feita e a criação do banco de dados finalizado, será iniciada a implementação das funcionalidades e comunicação da página com o banco de dados. Neste ponto, a aplicação começará a ter páginas dinâmicas devido as informações que são buscadas no banco e carregadas na página quando requisitadas.

A implementação das funcionalidades e comunicação com o banco de dados serão desenvolvidas com as linguagens de Programação *PHP* e *mysql* e com as ferramentas *XAMP* e *APACHE* que foram mencionadas do Capítulo 2. Com isso tem-se a finalização da implementação da aplicação.

4.5 Testes

Com a implementação da aplicação concluída, chega-se a fase de testes. Nessa etapa é necessário validar todos os pontos do site, tanto implementação de funcionalidades como implementação de estilização ou marcação.

Caso algum cenário de validação apresente erro ou retorne uma resposta inesperada, é necessário solucionar este ponto para que a aplicação tenha um código limpo e seguro. Durante os testes, também podem ser apontados detalhes de implementação que podem agregar ainda mais ao trabalho.

Se tudo estiver funcionando corretamente, será iniciado um dos últimos processos conhecido como minificação de *CSS*, *JavaScript* e *jQuery*, que tem a finalidade de deixar esses códigos mais leves deixando a navegação do site fluir com mais rapidez.

EXPLORAR OS TESTES CONCEITUALMENTE
COMO O TESTE UNITÁRIO, INTEGRAÇÃO, FUNCIONAL
DESEMPENHO E ETC.

5 Cronograma

Neste capítulo será apresentado o cronograma do desenvolvimento da aplicação para os próximos seis meses, onde é descrito o mês em que cada atividade será realizada.

A tabela 1, apresenta o cronograma do desenvolvimento da aplicação. O cronograma é organizado da seguinte maneira: o mês de Julho será alocado para a definição das funcionalidades, o mês de Agosto será utilizado para a criação da estrutura da aplicação e estilização da mesma, em Setembro será realizada a modelagem do banco de dados e será iniciada a criação do banco de dados, em Outubro será terminada a de criação do banco de dados e iniciada a codificação das funcionalidades, em Novembro ocorrerá o término da codificação das funcionalidades e início dos testes da aplicação e por fim, em Dezembro será realizado o desfecho dos testes inferidos na aplicação.

Tabela 1 – Cronograma do desenvolvimento da aplicação

Atividade	Jul	Ago	Set	Out	Nov	Dez
Definição de Funcionalidades	•					
Criação da estrutura da aplicação		•				
Estilização da aplicação		•				
Modelagem do banco de dados		•	•			
Criação do banco de dados			•	•		
Codificação das funcionalidades				•	•	
Testes e correções					•	•

ESCRITA
DEFESA

Fonte: Próprio Autor

6 Resultados Esperados

Com o desenvolvimento do trabalho, espera-se conseguir uma aplicação Web para gerenciamento acadêmico eficiente e eficaz, que possa facilitar e ajudar não só alunos, mas também professores e instituições de ensino, almejando também uma melhora no aprendizado e desenvolvimento dos discentes.

Através dessa aplicação em instituições de ensino, é esperado que a mesma facilite a interação do aluno com os materiais complementares, datas e horários de eventos importantes como avaliações ou trabalhos em geral e com a visualização de seus dados acadêmicos, como lista de presença ou resultados avaliativos por qualquer dispositivo com acesso a internet.

Espera-se também que esta aplicação gere um apoio aos professores, como um ambiente seguro para armazenamento de resultados avaliativos ou lista de presença dos alunos, um local para a disponibilização de materiais complementares e um mural para avisos de possíveis eventos aos alunos de cada classe ou turma.

Referências

CARBONELL, M. M. Phpmyadmin. Recuperado de <http://personales.upv.es/moimacar/download/servidores/phpmyadmin.pdf>, el, v. 1, 2018.

COELHO, J. C. B.; FILHO, D. R. B. Sistema de gerenciamento acadêmico. 2016. Disponível em: <https://www.comp.uems.br/~PFC/PFC/%20177.pdf>.

CONVERSE, T.; PARK, J. **PHP: a bíblia**. [S.l.]: Gulf Professional Publishing, 2003.

FAIN, Y. et al. **Enterprise Web Development: Building HTML5 Applications: From Desktop to Mobile**. O'Reilly Media, 2014. ISBN 9781449357061. Disponível em: https://books.google.com.br/books?id=xF_xAwAAQBAJ.

FERRARI, F. **Criação de banco de dados com MySQL**. Universo dos Livros Editora, 2010. ISBN 9788578731113. Disponível em: <https://books.google.com.br/books?id=0j-xNtXHTlgC>.

FERREIRA, H. K.; ZUCHI, J. D. Análise comparativa entre frameworks frontend baseados em javascript para aplicações web. **Revista Interface Tecnológica**, v. 15, n. 2, p. 111–123, dez. 2018. Disponível em: <https://revista.fatectq.edu.br/interfacetecnologica/article/view/502>.

FLANAGAN, D. **JavaScript: O Guia Definitivo**. Bookman Editora, 2015. ISBN 9788565837484. Disponível em: <https://books.google.com.br/books?id=zWNYDgAAQBAJ>.

FRIENDS, A. Xampp apache+ mariadb+ php+ perl. **Retrieved april**, v. 5, 2019.

LAURIE, B.; LAURIE, P. **Apache: The Definitive Guide**. O'Reilly Media, Incorporated, 2003. (Definitive Guide Series). ISBN 9780596002039. Disponível em: <https://books.google.com.br/books?id=1z6QfgsnpKsC>.

LEAL, A. B.; LUSTOSA, E. C. Sistema web para gestão acadêmica: modelagem e prototipação. 2013. Disponível em: http://repositorio.unifesspa.edu.br/bitstream/123456789/153/1/TCC_%20Sistema%20web%20para%20gest%c3%a3o%20acad%c3%aamica.pdf.

LEITE, L. et al. Sistema de gerenciamento de dados acadêmicos integrado ao ambiente virtual de aprendizagem sav/moodle. **Palmas, TO**, 2015.

LUCIDCHART. **Lucidchart Web**. 2022. Disponível em: <https://www.lucidchart.com/pages/>. Acesso em: 04 jul. 2022.

MILANI, A. **MySQL-guia do programador**. [S.l.]: Novatec Editora, 2007.

RAMAKRISHNAN, R.; GEHRKE, J. **Sistemas de gerenciamento de banco de dados-3**. [S.l.]: AMGH Editora, 2008.

SILVA, M. **HTML5 – 2ª Edição: A linguagem de marcação que revolucionou a web**. Novatec Editora, 2014. ISBN 9788575224038. Disponível em: <https://books.google.com.br/books?id=f7JVBQAAQBAJ>.

SILVA, M. **Fundamentos de HTML5 e CSS3**. Novatec Editora, 2015. ISBN 9788575224380. Disponível em: <<https://books.google.com.br/books?id=2iPYCQAAQBAJ>>.

SILVA, M. A. Pereira da; NETO, N. M.; SILVA, W. R. Pereira da. **Sistema web de gerenciamento acadêmico da escola de música Cover's**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2011.

SILVA, M. S. **jQuery-A Biblioteca do Programador JavaScript-3ª Edição: Aprenda a criar efeitos de alto impacto em seu site com a biblioteca JavaScript mais utilizada pelos desenvolvedores web**. [S.l.]: Novatec Editora, 2013.

VALADE; BALLAD, B. **PHP and MySQL Web Development All-in-One Desk Reference For Dummies**. Wiley, 2011. ISBN 9781118051757. Disponível em: <<https://books.google.com.br/books?id=ZtOHNw6jA7kC>>.