

GESTÃO DE WEB CONSULTA ESTRUTURADA*Helikson Maurilio¹; Ellen Morganna²; Roberto Assis³**^{1, 2, 3} Universidade de Uberaba**heliksonmaurilio@live.com¹, roberto.assis@uniube.br³***Resumo**

Empresas de médio e grande porte possuem uma maior necessidade de proteção das informações, já que o acesso aos dados deverá ser restrito apenas para colaboradores com níveis específicos de permissões. Estes dados, normalmente, são visualizados a partir de *softwares* relacionados com o banco de dados utilizado, havendo a necessidade de possuí-los em sua máquina para utilização, muitas das vezes os usuários não possuem experiência em utilizar estes *softwares* sendo necessário realizar alguns treinamentos para o aperfeiçoamento dos colaboradores. Embasando nestas informações, o presente projeto tem como a principal regra de negócio facilitar o acesso às informações disponíveis no banco de dados, também sendo possível realizar a inserção, a remoção e a atualização destas, tudo isso de forma *on-line* e que possa ser acessada por qualquer dispositivo, em qualquer lugar, devendo apenas ter o nível hierárquico necessário para realização das mesmas. Foi utilizado no desenvolvimento o banco de dados *Oracle* e as linguagens que melhor se encaixaram para o desenvolvimento do sistema, foram: *Javascript*, PHP, CSS e HTML. Com isso foi desenvolvido uma aplicação totalmente funcional que facilita e auxilia em buscas rápidas de informações sem a necessidade de possuir programas gerenciadores de banco de dados, possuindo apenas a necessidade de um prévio cadastro de uma SQL que poderá ser executada diversas vezes com valores dinâmicos.

Palavras-chave: Banco de dados. Informação. SQL.

1 Introdução

Pode-se afirmar que a segurança da informação possui três pilares cruciais para qualquer

empresa. Este projeto abordará um de seus princípios: a confidencialidade da informação. Neste, foi definido níveis de permissões para cada pessoa, ou seja, usuários com permissões mais amplas de acessos conseguirão visualizar mais informações do que os usuários em níveis mais restritos. Com base nesse princípio foi realizada a implementação de permissões para uma maior flexibilidade na administração e controle de acessos para cada usuário.

O objetivo principal é a otimização do tempo gasto na criação dos relatórios, a sua homogeneização e facilitar o acesso à informação para usuários com pouco ou nenhum conhecimento em banco de dados. Sendo assim, todos os relatórios serão executados na mesma página, dispensando a criação de sistemas adicionais e retrabalho para ensinar os usuários sobre locais ou como usá-los. Já para os desenvolvedores ele dará maior velocidade de criação dos relatórios, pois será necessário apenas criar a SQL e adicioná-la no sistema junto com as permissões de quem terá acesso a ela.

2 Materiais e Métodos

Em meio a inúmeras linguagens de programação para internet, o PHP foi escolhido como linguagem principal para o *back-end*, pois ele conta com mais de 20 anos de trajetória se tornando uma das linguagens mais utilizadas no mundo para desenvolvimento *web*. Neste projeto foi utilizada a versão 5.6 pela compatibilidade com os sistemas já utilizados dentro da empresa na qual foi implementado. O PHP possibilitou a realização da conexão com o banco de dados e a manipulação da informação para a execução dos comandos no sistema.

Assim como a versão do PHP, o banco de dados da *Oracle* foi utilizado levando em consideração os sistemas já em uso na empresa.

12º ENTEC – Encontro de Tecnologia: 16 de outubro a 29 de novembro de 2018

Trata-se de um banco relacional robusto, seguro e versátil, podendo ser utilizado tanto nos sistemas operacionais *Linux* quanto *Windows*.

As linguagens de marcações são utilizadas para a base estrutural de conteúdo em diversos sistemas *web*, pois possibilita a manipulação através de *ids* e classes para os elementos, podendo até hierarquizá-los sendo um grande auxiliador para o *front-end*.

O HTML está na sua versão 5 que foi publicado em 2008 pelo W3C, dentre as linguagens de marcação ela é a mais utilizada na maioria dos sistemas *web*, por este motivo foi escolhido para projeto.

Foi optado pela utilização de um *framework* visando a otimização da criação dos estilos desde o princípio, obtendo como resultado a facilidade e a redução no tempo de desenvolvimento do *front-end*.

O *framework* utilizado na aplicação foi o *Bootstrap*, criado à base de *CSS3* e *Javascript*, que por sua vez ajudou nas manipulações de cores e outros aspectos visuais do sistema, como o posicionamento de elementos através do recurso de 12 colunas que a ferramenta disponibiliza.

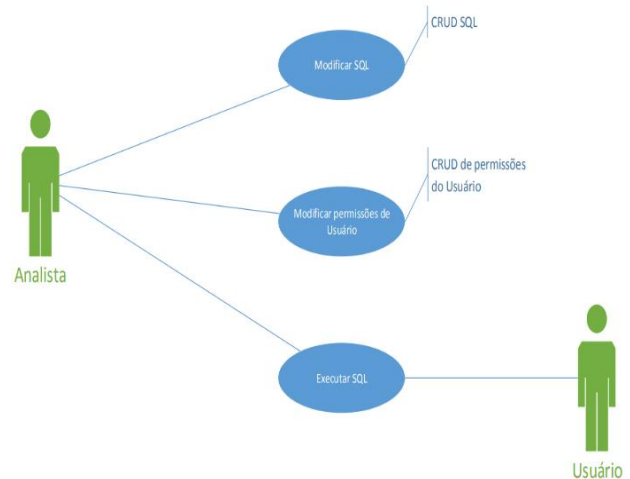
Em 1995, o *Javascript* apareceu pela primeira vez com apenas um objetivo, a manipulação de algumas validações de entrada. Hoje a linguagem não está restrita a apenas a validações simples, visto que a mesma ganhou recursos que interagem com quase todos os elementos disponíveis na janela do navegador. Com os recursos que ele contém, foi implementado para a realização de alterações no HTML através do DOM, possibilitando a criação de objetos que fazem referência diretas a *tags* do HTML possibilitando a manipulação dos elementos e facilitando a conexão do *front-end* com o *back-end*.

O planejamento é um dos principais fatores responsáveis pelo sucesso de um *software*, pois mostra de forma tangível os passos que devem ser seguidos para um bom desenvolvimento. Foi usufruído de três metodologias para o planejamento inicial do sistema.

O diagrama de Caso de Uso, que faz parte da documentação inicial de qualquer sistema que

utilize uma linguagem de programação orientada a objeto, mostrando de forma simples como o sistema deve se comportar. Na Figura 1 temos o diagrama de Caso de Uso do sistema desenvolvido.

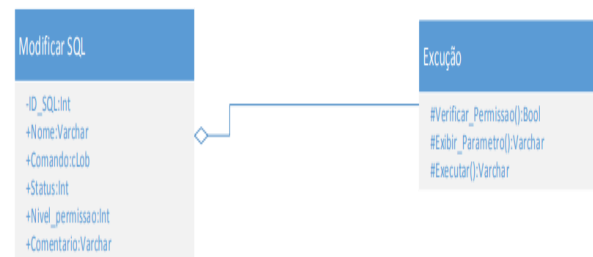
Figura 1: Diagrama de Caso de Uso



Fonte: Autor e coautor.

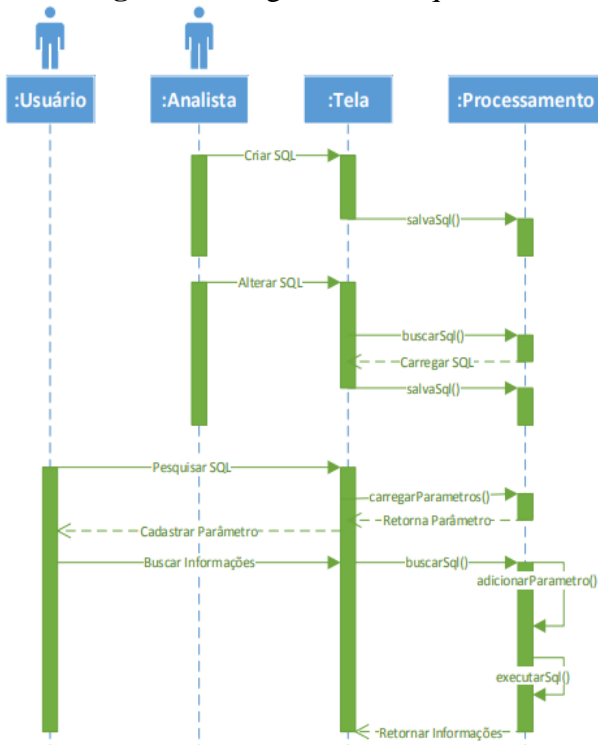
Com o diagrama de Classes, é possível definir todas as classes necessárias a serem criadas na fase desenvolvimento. O diagrama auxiliou na otimização da quantidade de classes e também ajudou a evitar duplicidades, ele pode ser visualizado na Figura 2.

Figura 2: Diagrama de Classes



Fonte: Autor e coautor.

Para o entendimento técnico do sistema o diagrama de Sequência mostra de forma detalhada cada requisição e respostas desejadas do sistema conforme disposto na Figura 3.

12º ENTEC – Encontro de Tecnologia: 16 de outubro a 29 de novembro de 2018
Figura 3: Diagrama de Sequência


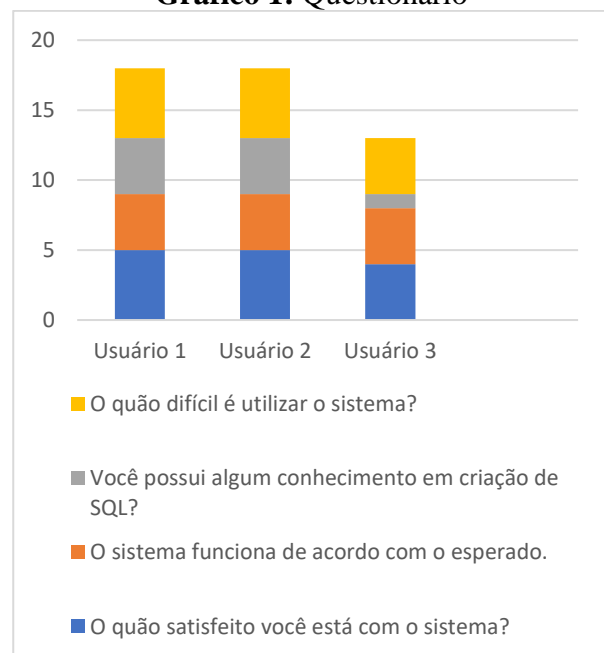
Fonte: Autor e coautor.

Após os primeiros passos do planejamento e entendimento do sistema, iniciou-se a fase do desenvolvimento. Logo no início foi definida a tabela para receber estes dados. Ela foi definida contendo: id como chave primária da tabela, nome da SQL, descrição, comando como sendo a SQL em si, status para definir se estará ativa ou inativa e o nível execução. Após estas definições, foi iniciado a distinção dos parâmetros do restante da SQL, visto que esses seriam dinâmicos. Para isso foi optado por criar uma nova especificação para delimitar as parametrizações dentro do sistema, que foi definido como: `::p_parametro::` onde “:” é o delimitador para ser identificado como parâmetro do sistema e “p_” definido para especificar o início do parâmetro, este, posteriormente, foi definido para distinguir as tipagens de dados que o sistema validará, ou seja, caso for adicionado “i_” o sistema validará como sendo um número inteiro, “dm_” será validado como decimal, “t_” como texto, “d_” como data e “dh_” para data e hora. Foi desenvolvido as funções para realizar a gravação das SQLs com os parâmetros e logo

após elas estarem salvas, as configurações de leitura, que consiste em remover os parâmetros da SQL e adicionar como entradas para o usuário, retirando os campos iguais e utilizando a validação anteriormente mencionada. No próximo passo, o que seria a execução do processo, foi realizado a mesma quebra para a exibição do usuário, só que com a diferença de que em vez de mostrá-lo foi realizada a alteração dos parâmetros que estavam contidos na SQL pelos valores digitados pelo usuário e logo após a troca dos parâmetros foi executada a SQL, onde caso seja um *select* ele captura os campos dinamicamente colocando como título da tabela e adicionando os valores retornados, caso seja um *insert*, *update* ou *delete*, ele apenas informa que foi realizado o propósito da SQL.

3 Resultados

Esse sistema pode ser usado em diversas áreas de uma empresa, mas para este, em específico, foi utilizado junto ao sistema de atendimento. Para a validação dos resultados criamos um questionário pontuando a utilização do sistema e níveis de melhorias que ele trouxe para os atendentes. Este questionário poderá ser visualizado na Gráfico 1.

Gráfico 1: Questionário


Fonte: Autor e coautor.

12º ENTEC – Encontro de Tecnologia: 16 de outubro a 29 de novembro de 2018

Foram selecionados apenas os valores 1, 2, 4 e 5 que significam 1 – Muito Ruim, 2 – Ruim, 4 – Bom, 5 – Muito bom. Estes valores foram selecionados com base na escala de Likert e foi retirado o valor 3, que significaria razoável, para evitar a neutralidade das pessoas ao questioná-las sobre a funcionalidade do sistema.

4 Discussão

Escolher a melhor forma para a apresentação dos dados não é fácil. É necessário entender o que o usuário final busca e como será a usabilidade do sistema. Este processo foi o que demandou uma quantidade maior de tempo, pois foi necessário um estudo sobre como seria realizado a apresentação dos dados, já que o mesmo seria dinâmico.

5 Conclusão

Com este projeto podemos concluir que existem diversas formas de visualização dos dados, sendo necessário apenas definir como visualizá-los.

Este projeto alcançou com êxito os objetivos definidos, melhorando consideravelmente a

velocidade em encontrar as informações e diminuindo o grau de complexidade para encontrá-las.

Referências

SILVA, Fernando. **História do PHP – Curiosidades por trás dos 23 anos da linguagem**. 2018. Disponível em:

<<https://king.host/blog/2018/06/historia-do-php>>. Acesso em: 07 out. 2018.

FLATSCHART, Fábio. **HTML5 Embarque imediato**. 2. ed. São Paulo: Brasport Livros e Multimídia Ltda, 2011.

OSMANI, Addy. **Learning JavaScript Design Patterns: A JavaScript and jQuery Developer's Guide**. 1. Ed. 2012.

MEDEIROS, Emani. **Desenvolvendo Software com UML Definitivo 2.0**. 6. ed. São Paulo: Person Education do Brasil, 2010.

ANNASTASI, A. **Psychological testing**. New-York: MacMillan. 1990.