

DOCUMENTO DE REQUISITOS
Sistema Gerenciador de Produção de Cerâmica

Aluno: Victor Gabriel Pereira Mendonça

Número: 36°

Data: 28/05/2024

Sumário

1. Introdução	3
1.1. Requisitos	3
1.1.1. O que são requisitos funcionais?	3
1.1.2. O que são requisitos não funcionais?	4
1.2. Descrição do Usuário	4
1.2.1. Atores	4
2. Requisitos Funcionais	5
2.1. Sistema de Login (RF001)	5
2.2. Gerenciamento de Usuários (RF002)	5
2.3. Gerenciamento de Funcionários (RF003)	6
2.4. Gerenciamento de Produção (RF004)	6
2.5. Gerenciamento da Folha de Pagamento (RF005)	7
3. Requisitos Não-Funcionais	7
3.1. Usabilidade (RNF001)	7
3.1.1. Responsividade	7
3.1.2. Validação de Campos	8
3.1.3. Mensagens de Aviso	8
3.2. Banco de Dados MySQL (RNF002)	8
3.3. PHP (RNF003)	8
3.4. Frameworks (RNF004)	9
3.4.1. Bootstrap	9
3.5. Parsley (RNF005)	9
3.5. Requisitos Mínimos Para o Uso (RNF005)	10
3.5.1. Bootstrap	10
3.5.2. PHP	11
3.5.2.1. PHP 8.0	11
3.5.3. PDO	11
3.5.4. Parsley	12
4. Referências Bibliográficas	12

1. Introdução

O Sistema Gerenciador de produção de cerâmica é um sistema que busca facilitar a organização e gerenciamento de um setor de suma importância de uma empresa como essa. A diferença do administrador para um usuário do sistema é que ele pode adicionar, editar e excluir os usuários que poderão ter acesso a esse sistema. O projeto tem como intuito que o usuário possa adicionar, editar e excluir seus funcionários, assim tendo ainda o local onde seria informado a quantidade de milheiros produzidos em um dia que posteriormente por uma opção nessa área o usuário finalizaria a produção da semana, que após isso o sistema calcula os milheiros produzidos no período e manda para a folha de pagamento que relaciona quanto cada funcionário ganha de acordo com a quantidade de milheiros produzidos. Esse software com poucas alterações pode ser alterado para diversos tipos de indústrias e oferta a possibilidade de diversos aprimoramentos futuros.

Esta documentação busca descrever o seu sistema e os requisitos do software, minuciando os requisitos funcionais e não funcionais. Além disso, descreve os tipos de usuários e suas interações com o sistema, bem como as tecnologias e frameworks utilizados no desenvolvimento do projeto.

1.1. Requisitos

Os requisitos de software são descrições funcionais e não funcionais do que um sistema de software deve fazer e como deve operar. Eles são divididos em duas categorias principais: requisitos funcionais e requisitos não funcionais.

1.1.1. O que são requisitos funcionais?

Estes descrevem as funções específicas que o software deve executar. Eles geralmente são declarados em termos de entradas, saídas e comportamento do sistema em resposta a várias entradas.

Exemplos:

- Processamento de transações
- Gerenciamento de usuários
- Busca e filtragem de dados
- Geração de relatórios

1.1.2. O que são requisitos não funcionais?

Os requisitos não funcionais são os critérios que descrevem as características operacionais de um sistema de software além das suas funcionalidades específicas.

Exemplos:

- Frameworks
- Manutenibilidade
- Portabilidade
- Usabilidade

1.2. Descrição do Usuário

um usuário é uma pessoa ou entidade que interage com um sistema de computador, software, rede ou serviço para realizar tarefas específicas. Os usuários podem incluir uma ampla gama de pessoas, desde funcionários de uma empresa que utilizam sistemas internos, até clientes que interagem com aplicativos ou serviços online.

1.2.1. Atores

Administrador/Usuário: O administrador pode adicionar, editar e excluir os usuários, assim ainda podendo atuar como um usuário comum do sistema.

Usuário Comum: O usuário comum pode gerenciar seus funcionários, adicionar a produção dos dias que posteriormente virá a ser enviada junto com as informações para a folha de pagamento com o valor ganho por cada funcionário.

2. Requisitos Funcionais

2.1. Sistema de Login (RF001)

O sistema deve permitir que o usuário crie sua conta e faça o login, caso o usuário e a senha sejam válidos.

Entradas: Nome de usuário e senha.

Saídas: Mensagens de sucesso ou erro.

Pré-condições: Nenhuma.

Atores: administrador e usuário comum

- O usuário ou administrador deve ser redirecionado para a página inicial caso o login seja bem sucedido.
- O usuário ou administrador deve exibir uma mensagem de erro caso o login ou a senha estejam incorretos.
- O usuário ou administrador deve exibir uma mensagem caso os inputs estejam vazios.

2.2. Gerenciamento de Usuários (RF002)

O sistema deve permitir que o administrador possa adicionar, editar e excluir os usuários que possa ter acesso ao software.

Pré-condições: O administrador deve está logado ao sistema.

Entradas: Informações necessárias do sistema(nome de usuário e senha).

Saídas: Mensagem de sucesso ou erro.

Atores: administrador

- O sistema deve exibir uma mensagem caso o processo seja executado corretamente.
- O sistema deve exibir uma mensagem caso ocorra um erro.
- O sistema deve permitir que o administrador possa ver os usuários adicionados.
- O sistema deve permitir que o administrador possa adicionar, editar e excluir os usuários.

2.3. Gerenciamento de Funcionários (RF003)

O sistema deve permitir que o administrador ou o usuário comum possa adicionar, editar e excluir os funcionários da produção informando seu nome e o ganho por milheiro.

Pré-condições: O usuário comum ou administrador deve está logado ao sistema.

Entradas: Informações necessárias do sistema(nome e o ganho por milheiro).

Saídas: Mensagem de sucesso ou erro.

Atores: administrador e usuário comum

- O sistema deve exibir uma mensagem caso o processo seja executado corretamente.
- O sistema deve exibir uma mensagem caso ocorra um erro.
- O sistema deve permitir que o usuário comum ou administrador possa ver os funcionários adicionados.
- O sistema deve permitir que o usuário comum ou administrador possa adicionar, editar e excluir os funcionários.

2.4. Gerenciamento de Produção (RF004)

O sistema deve permitir que o usuário informe qual foi a produção do sistema naquele dia que depois virá a ser somada para ser enviada para a folha de pagamento.

Pré-condições: O usuário comum ou administrador deve está logado ao sistema.

Entradas: Informações da produção (dia da produção e milheiros produzidos).

Saídas: Mensagem de sucesso ou erro.

Atores: administrador e usuário comum

- O sistema deve exibir uma mensagem caso o processo seja executado corretamente.
- O sistema deve exibir uma mensagem caso ocorra um erro.
- O sistema deve permitir que o usuário comum ou administrador possa ver as produções.

- O sistema deve permitir que o usuário comum ou administrador possa adicionar ou excluir às produções diárias.
- O sistema deve permitir que o usuário comum ou administrador possa finalizar a produção e que assim enviará às informações para a folha de pagamento.

2.5. Gerenciamento da Folha de Pagamento (RF005)

O sistema deve permitir que o usuário comum ou administrador possa visualizar às folhas de pagamento para cada funcionário e após o pagamento possa finalizar e assim apagar a folha finalizada.

Pré-condições: O usuário comum ou administrador precisa está logado ao sistema.

Entradas: Informações das folhas de pagamento (nome do funcionário, salário, milheiros produzidos e quanto ganha por cada milheiro produzido).

Saídas: Listagem da folha de pagamento de cada funcionário.

- O sistema deve permitir a visualização da folha de cada funcionário.
- O sistema deve permitir finalizar a folha de pagamento para que seja apagada.

3. Requisitos Não-Funcionais

3.1. Usabilidade (RNF001)

A usabilidade é a facilidade com que os usuários podem interagir com um sistema de computador, software, aplicativo ou website para atingir seus objetivos de forma eficaz, eficiente e satisfatória. Em outras palavras, é a medida em que um sistema é amigável para o usuário.

3.1.1. Responsividade

A responsividade é a capacidade de um sistema, aplicativo ou website de se adaptar e responder de forma rápida e eficiente às ações do usuário ou às mudanças no ambiente.

3.1.2. Validação de Campos

A validação de campos é o processo de verificar se os dados inseridos em campos de entrada de um sistema atendem aos critérios especificados antes de serem processados ou armazenados. Essa prática é fundamental para garantir a integridade, consistência e segurança dos dados em um sistema.

3.1.3. Mensagens de Aviso

As mensagens de aviso são notificações que alertam os usuários sobre situações potencialmente problemáticas ou condições incomuns que ocorrem em um sistema, aplicativo ou rede. Essas mensagens são projetadas para chamar a atenção dos usuários para possíveis problemas ou eventos importantes que exigem sua atenção ou ação.

3.2. Banco de Dados MySQL (RNF002)

O MySQL é um sistema de gerenciamento de banco de dados relacional (SGBDR) de código aberto amplamente utilizado. Ele é conhecido por sua confiabilidade, desempenho e facilidade de uso, sendo uma escolha popular para aplicativos da web, desde pequenos sites até grandes sistemas corporativos.

O MySQL é o banco de dados mais conhecido no mundo. De acordo com oDB-engines, o MySQL é o segundo banco de dados mais popular, ficando atrás do Oracle Database. O MySQL alimenta muitas das aplicações mais acessadas, como Facebook, Twitter, Netflix, Uber, Airbnb, Shopify e Booking.com.

Vantagens:

As suas principais vantagens do MySQL é o seu desempenho rápido, escalabilidade, compatibilidade, suporte da comunidade e sua ampla gama de recursos avançados.

3.3. PHP (RNF003)

PHP é um acrônimo recursivo para Hypertext Preprocessor. É uma linguagem de script open source de uso geral e adequada para o desenvolvimento web, podendo ser embutida dentro do HTML. Foi originalmente criada em 1994 por Rasmus Lerdorf

e posteriormente desenvolvida por uma comunidade de desenvolvedores de software. O PHP é uma das linguagens de programação mais populares para desenvolvimento web devido à sua facilidade de aprendizado, ampla adoção e poderosas capacidades.

Vantagens:

As suas principais vantagens se dar por sua sintaxe simples e intuitiva, possuir uma grande comunidade de desenvolvedores ao redor do mundo, compatibilidade com diversos sistemas operacionais, ser gratuito e de código aberto, suportar Bancos de Dados e possui diversos frameworks disponíveis.

3.4. Frameworks (RNF004)

Os Frameworks são conjuntos de ferramentas, bibliotecas e padrões de desenvolvimento que fornecem uma estrutura organizada e reutilizável para facilitar o desenvolvimento de software. Eles são projetados para acelerar o processo de desenvolvimento, fornecendo soluções prontas e abstrações para problemas comuns encontrados no desenvolvimento de aplicativos.

3.4.1. Bootstrap

O Bootstrap é um framework front-end que fornece estruturas de CSS para a criação de sites e aplicações responsivas de forma rápida e simples. Além disso, pode lidar com sites de desktop e páginas de dispositivos móveis da mesma forma.

Vantagens:

- As vantagens do bootstrap se dar pela rapidez de desenvolvimento, responsividade, consistência e personalização fácil, tornando-o uma ferramenta útil para criar aplicações web de alta qualidade.

3.5. Parsley (RNF005)

O Parsley.js é uma biblioteca JavaScript de validação de formulários que simplifica o processo de validação de entrada de dados em formulários HTML. Ele permite que você defina regras de validação diretamente no HTML, sem a necessidade de escrever scripts de validação personalizados. O Parsley.js fornece

uma maneira fácil de adicionar validação de formulários, como verificar se um campo está preenchido, se um endereço de e-mail é válido, se um número está dentro de um determinado intervalo e muito mais.

Vantagens:

- o Parsley.js é uma boa escolha e flexível para adicionar validação de formulários aos projetos web, oferecendo uma variedade de recursos e vantagens que facilitam o processo de validação de entrada de dados e melhoram a experiência do usuário.

3.5. Requisitos Mínimos Para o Uso (RNF005)

São os requisitos de versão mínima do navegador para rodar o site, compatibilidade com os principais navegadores (Chrome, Firefox, Safari, Edge), requisitos de hardware e software para o servidor e os dispositivos clientes.

3.5.1. Bootstrap

- Navegadores alternativos que usam a versão mais recente do WebKit, Blink ou Gecko (tanto diretamente, quanto através da API web view da plataforma) não suportam explicitamente o Bootstrap.
- Suporte a navegadores modernos como Google Chrome, Mozilla Firefox, Safari, Microsoft Edge.
- Suporte básico ao Internet Explorer 11 (embora o Bootstrap 5 tenha removido o suporte oficial, projetos legados podem exigir).
- Organização do projeto que permita a inclusão dos arquivos principais do Bootstrap (CSS e JS) e suas dependências, como Popper.js e jQuery (caso necessário).
- Estrutura de pastas adequada para arquivos de estilos personalizados, scripts e assets.
- Minificação dos arquivos CSS e JS para melhorar o desempenho do carregamento da página.

3.5.2. PHP

- Configuração adequada do servidor para processar arquivos PHP (geralmente via módulo ou FastCGI).
- Apache, Nginx, IIS ou qualquer outro servidor web compatível com PHP.
- PHP é compatível com diversos sistemas operacionais, incluindo Linux, Windows, macOS e outros sistemas baseados em Unix.

3.5.2.1. PHP 8.0

- Servidor Web: Apache 2.4 ou Nginx 1.18.
- Sistema Operacional: Ubuntu 20.04, Windows 10, macOS Catalina.
- Sistema Operacional: Ubuntu 20.04, Windows 10, macOS Catalina.
- Extensões: ctype, fileinfo, mbstring, openssl, PDO, tokenizer, xml.
- Memória: 1GB RAM (mínimo para produção).
- Banco de Dados: MySQL 5.7+, MariaDB 10.2+, PostgreSQL 10+, SQLite 3.31+.
- Ferramentas: Composer, Git.

3.5.3. PDO

- **PHP 5.1.0 ou superior:** PDO foi introduzido no PHP 5.1.0, então você precisa pelo menos dessa versão. No entanto, é recomendado usar uma versão mais recente do PHP, como PHP 7.x ou 8.x, para garantir suporte e segurança.
- **Servidor de banco de dados compatível:** O servidor de banco de dados que você planeja usar deve estar instalado e configurado adequadamente. Isso inclui servidores como MySQL, PostgreSQL, SQLite, etc.
- **Servidor Web:** Um servidor web como Apache, Nginx, ou IIS, configurado para processar arquivos PHP.
- **Sistema Operacional:** Qualquer sistema operacional compatível com PHP, como Linux, Windows, ou macOS.

3.5.4. Parsley

- **HTML5:** O Parsley depende de várias funcionalidades HTML5, como required, type, min, max, pattern, etc. Portanto, seu formulário deve usar esses recursos HTML5 para que o Parsley funcione corretamente.
- **jQuery:** Embora o Parsley não seja estritamente dependente do jQuery, é altamente recomendado usá-lo com jQuery, pois simplifica a inicialização e o uso da biblioteca. Portanto, certifique-se de incluir a biblioteca jQuery no seu projeto.
- **Parsley.js:** Obviamente, você precisará incluir o arquivo parsley.js ou parsley.min.js no seu projeto. Você pode baixar o Parsley.js do seu site oficial ou incluí-lo via CDN.

4. Referências Bibliográficas

<https://www.alura.com.br/artigos/bootstrap#:~:text=Bootstrap%20%C3%A9%20um%20framework%20front,dispositivos%20m%C3%B3veis%20da%20mesma%20forma.>

Acessado: 24/05/2024

<https://www.mestresdaweb.com.br/tecnologias/requisitos-funcionais-e-nao-funcionais-o-que-sao#:~:text=Em%20geral%2C%20um%20requisito%20funcional,confiabilidade%2C%20e%20assim%20por%20diante.>

Acessado: 24/05/2024

<https://www.oracle.com/br/mysql/what-is-mysql/>

Acessado: 24/05/2024

<https://www.w3schools.com/>

Acessado: 24/05/2024

<https://coodesh.com/blog/dicionario/o-que-e-usabilidade/>

Acessado: 24/05/2024

https://www.php.net/manual/pt_BR/

Acessado: 24/05/2024

<https://parsleyjs.org/>

Acessado: 24/05/2024