
S.O.S docs



Controle de Patrimônios Documento de Arquitetura de Software

Versão 4.0.00

O Sistema opera para controle de patrimônios em itens variados, dessa forma é possível se obter uma melhor organização, neste documento estará explicado métodos aplicados para a funcionalidade do sistema.

Controle de Patrimônios	Version: 4.0.00
Documento de Arquitetura de Software	Date: 20/09/2020
Projeto	

Histórico da Revisão

Data	Versão	Descrição	Autor
20/09/2020	4.0.00	Foi feito a parte inicial pedido no escopo	Keven Klain

Controle de Patrimônios	Version: 4.0.00
Documento de Arquitetura de Software	Date: 20/09/2020
Projeto	

Documento de Arquitetura de Software

1. Introdução

O software irá fazer registros de patrimônios e determinadas marcas para um melhor controle sobre os patrimônios.

1.1 Finalidade

Administrar o controle d patrimônios distribuídos

1.2 Escopo

Este documento serve para constante atualização, qualquer alteração feita sob o projeto será registrado neste arquivo.

1.3 Visão Geral

O documento vai está organizado de acordo com a funcionalidade do sistema e como excuta-lo de forma simples.

2. Representação Arquitetural

O sistema foi toda organizado em MVC, utilizando principalmente o Framework Angular do Google, que exige está organização.

3. Visão Lógica

O sistema foi separado em uma logica incluindo o, Frontend, utilizando, Angular, Bootstrap, a parte de back, vem com a curva de aprendizado grande, utilizando o Banco de Dados Postgresql, Php, Api Service e Json.

3.1 Pacotes de Design Significativos do Ponto de Vista da Arquitetura

Angular/Bootstrap: Foi utilizado no frontend, devido a praticidade em gerar validações importantes do sistema com alerts simples de se entender para o usuário.

PHP:Foi utilizado como a parte lógica do backend, gerando consultas no banco, carregando inputs gerados pelo o usuário, alimentando a Api via Json e gerando inserts, gerando mensagens de sucesso e falhas.

Posgresql Foi utilizando para gerar um ambiente onde os dados possam ser armazenados e utilizando logicas nos receptivos itens que não podem se repetir.

4. Visão de Processos

Cadastro

Patrimonio

- POST patrimônios - Inserir um novo patrimônio
- N° Tombo / Id da Marca / Nome da Marca / Descrição do Produto

Marca

- POST marcas - Inserir uma nova marca
- Id da Marca / Nome da Marca

Controle de Patrimônios	Version: 4.0.00
Documento de Arquitetura de Software	Date: 20/09/2020
Projeto	

Controle

Patrimonios Cadastrados

- GET patrimonios - Obter todos os patrimônios
Tela View de Todos os Itens
- GET patrimonios/{id} - Obter um patrimônio por ID
Campo de Pesquisa
- PUT patrimonios/{id} - Alterar os dados de um patrimônio
Botão de Alterar
- DELETE patrimonios/{id} - Excluir um patrimônio
Botão de Excluir

Marcas

- GET marcas - Obter todas as marcas
Tela View de Todos os Itens
- GET marcas/{id} - Obter uma marca por ID
Pesquisar marca pela ID
- GET marcas/{id}/patrimônios – Obter todos os patrimônios de uma marca
View Com Todos o Patrimonios da Marca
- PUT marca/{id} - Alterar os dados de uma marca
Botão de Alterar
- DELETE marca/{id} - Excluir uma marca
Botão Excluir

Anexos

Armazenamento de Dados

- O sistema deverá permitir ao usuário realizar upload/download de um arquivo PDF através dos serviços e interface.

5. Visão de Implantação

Primeira parte da Implementação

Precisamos do PHP – 7 Instalado, para rodar o servidor PHP e termos resultados em nossas requisições.

Como instalar:

Acesse: <https://windows.php.net/download/>

Baixe o https://aka.ms/vs/16/release/VC_redist.x64.exe Visual Studio 2017 or 2019

Baixe o <https://windows.php.net/downloads/releases/php-7.4.10-Win32-vc15-x64.zip> PHP zipado

Thread Safe, extraia o arquivo com o 7zip ou Winrar, copie para o diretoria raiz do seu Windows.

Logo após ser feito isso, abra o path do seu Windows e coloque a pasta do diretório e o php.exe.

Logo após ser feito isso, aperte Win+R, inicie seu cmd e faça o seguinte comando.

Php -S 127.0.0.1:8041

Controle de Patrimônios	Version: 4.0.00
Documento de Arquitetura de Software	Date: 20/09/2020
Projeto	

Após realizado este comando seu servidor php estará funcionando, se desejar fazer um arquivo .bat, facilita ao executar isso em aplicações futuras.

Segunda parte da Implementação

Precisamos do postgresql, para instalar acesse: <https://www.postgresql.org/download/>

Realize o download e execute, ele irá pedir configurações simples de instalação como portas que Que deseja ocupar em seu servidor local e configurações de senhas para acessar o banco toda vez Precisar.

Terceira parte da Implementação

Faça o Download do Angular: <https://angular.io/guide/setup-local>

Neste link vai está explicando os comando necessários no seu cmd, execute-o como administrador.

Escolha onde deseja instalar.

Após isso realize o download do Node.js: <https://nodejs.org/en/download/>, escolha a plataforma Desejada.

A instalação é simples, apenas execute e aperte next até finalizar.

Finalizando

Abra o cmd, crie uma pasta para armazenar o projeto, acesse novamente o cmd, digite os comandos abaixo e o projeto está pronto para ser iniciado.

Ng new hcode – Criar seu novo projeto

Escolha a opção CSS

Ng g c “nome do arquivo” – onde será criado as pastas do seu projeto

Ng server – inicia seu servidor, coleto o local host gerado e incie teste em sua aplicação

6. Visão da Implementação

Iniciar o autoloading na aplicação.

Iniciar o container/repositório de dependências e carregar as configurações.

Rotear a chamada que foi feita para ele para o controller correto, injetando as dependências necessárias.

Capturar erros para evitar mensagens de debug ou telas em branco aos usuários.

Em alguns casos de alguns frameworks por exemplo o controller pode devolver um objeto para seu invocador, em outros casos já vi implementações em que o controller despacha diretamente a resposta para o cliente.

6.1 Visão Geral

O projeto foi separado com a logica MVC, ara fácil manutenção e maior desempenho, com isso levou cerca de 26 horas para ser implementado.