

Documento de Arquitetura

Índice

1. **Arquitetura do Software**
 - 1.1. **Finalidade do Documento**
 - 1.2. **Contexto**
2. **Diagrama da Arquitetura**

Documento de Arquitetura

Controle do documento

Versão	Data	Autor / Revisor	Descrição
0.1	18/10/2017	João Henrique	1. Versão inicial a. Criação do documento b.

1. Arquitetura do Software

1.1. Finalidade do Documento

Este documento tem como objetivo fornecer uma visão arquitetural abrangente do sistema. Ele pretende capturar e transmitir as decisões arquiteturas significativas que foram tomadas em relação ao sistema.

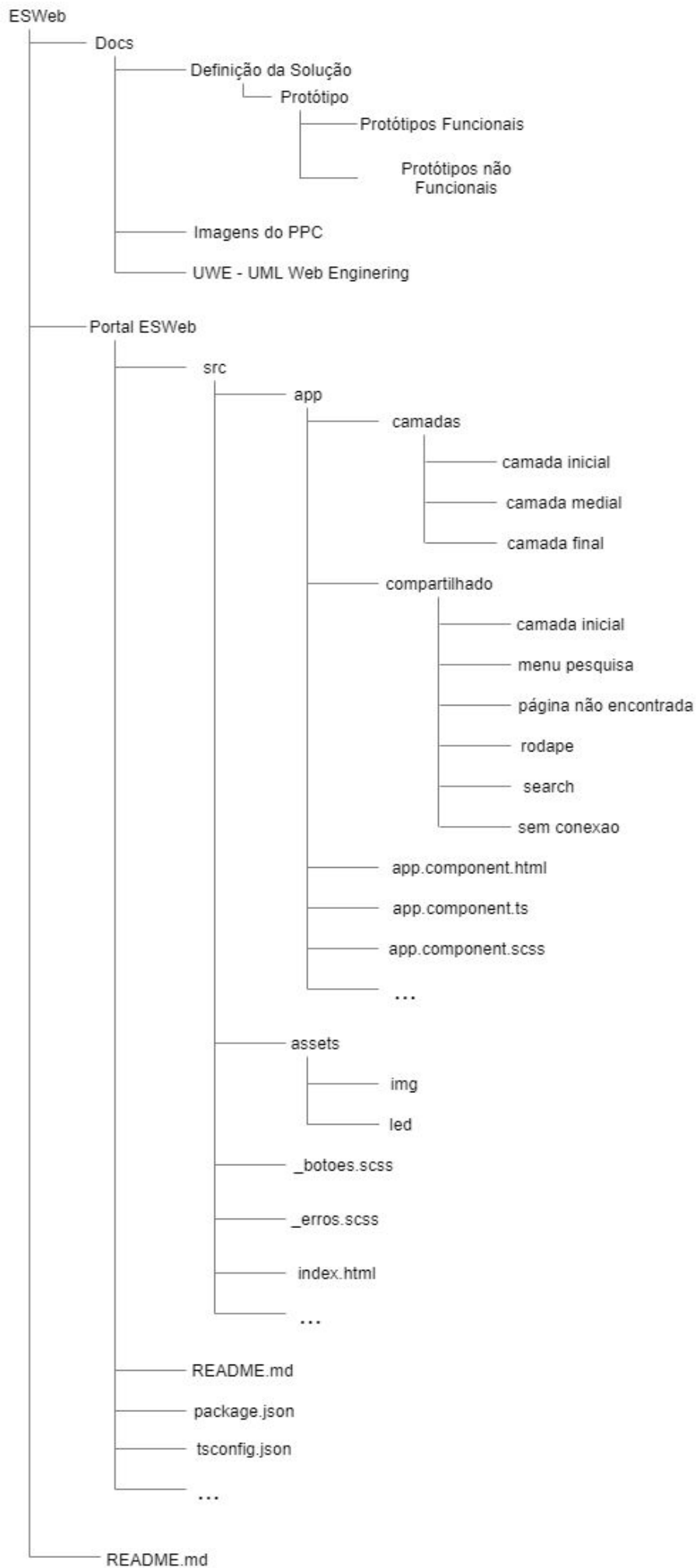
1.2. Contexto

O projeto tem como objetivo fornecer e elucidar uma visão coesa e integrada da Engenharia de Software de tal forma que o estudante desenvolva uma sólida percepção de como fazer uso do conhecimento desta área em toda a sua extensão. Levando em conta que toda a população deveria ter acesso a tais informações, de forma fácil e rápida, o contexto da WEB se tornou um requisito essencial para a criação de nosso software. Para isso foi realizado um estudo a cerca de todas as tecnologias e ferramentas disponíveis atualmente, e após uma análise detalhada, foi escolhido como tecnologia o *Angular 4* juntamente com frameworks auxiliar (Ex: Bootstrap). Uma vez que o framework *angular 4* foi decidido para ser utilizado em nosso projeto, o mesmo segue uma estrutura padrão de pastas e arquivos necessários para sua compilação e melhor organização, sempre buscando o máximo de qualidade e simplicidade no que é produzido. A estrutura de pastas e arquivos, assim como a descrição do mesmo será explicada nos tópicos seguintes. O projeto não possuirá um banco de dados, uma vez que

todo o conteúdo que será consumido estará dentro da própria estrutura do mesmo, armazenado e estruturado em formato JSON.

2. Diagrama da Arquitetura

Na imagem abaixo teremos uma visão da estruturação do projeto, e em seguida o detalhamento do mesmo.



Como dito anteriormente, o projeto ESWeb foi estruturado em um padrão já proveniente do Angular 4, uma vez que este é criado automaticamente, é organizado, e se encaixa no contexto de nossos objetivos para com o produto final.

ESWeb (Principais Pastas)

O projeto inicia-se com com duas pastas e um arquivo README.md (onde este último terá informações gerais a respeito do projeto e integrantes da equipe). As pastas possuem por finalidade separar o que será a documentação necessária (onde está incluso arquivos de textos, imagens, protótipos, entre outros).

1 - Docs

A pasta Docs, como dito anteriormente guarda as documentações presentes para projeto, abrigando as pastas: Definição da Solução, Imagens do PCC e UWE - UML Web Engeneering.

1.1 - Definição da Solução/Protótipos

A pasta Solução abriga um visual da solução previamente definida pelo grupo e acordada com o stakeholder principal (Professor Fábio Lucena). E dentro dela teremos os Protótipos funcionais e não funcionais.

1.1.1 - Protótipos Funcional

A pasta Protótipo Funcional abriga o código necessário para “rodar” o mesmo.

1.1.2 - Protótipo não Funcional

A pasta protótipo não funcional apresenta três arquivos .pdf, onde em cada um deles é apresentado uma visão prévio de cada camada (lembrando que o projeto trabalha com o contexto de três camadas, como será explicado mais a frente).

Os arquivos .pdf podem ser encontrados e visualizados no seguinte link: [protótipos não funcionais](#).

1.2 - Imagens do PPC

Esta pasta contém todas as imagens que serão utilizadas na apresentação das informações sobre o curso de Engenharia de Software, todas retiradas do próprio documento [PPC](#).

1.3 - UWE - UML Web Engeneering

Nesta pasta é possível encontrar arquivos .pdf contendo os diagramas de classes, diagramas de casos de uso, os modelos de apresentação de cada camada, assim como o modelo de navegação.

2 - Portal ESWeb

A pasta Portal ESWeb é onde estará presente todo o código fonte do projeto principal, e também onde conseguimos encontrar a estrutura padrão definida pelo framework Angular.

2.1 - src/app

A pasta src/app conterá duas pastas principais, as pastas Camadas e Compartilhado. Enquanto a primeira terá todo o código fonte dividido para cada camada (cada camada terá seu próprio conjunto de código fonte), a segunda apresenta toda a codificação que é “compartilhada” por todo o sistema de camada.

2.1.1 - Camadas

Pasta com a codificação de cada camada.

Primeira Camada: A primeira camada consiste em uma exposição de notícias relevantes sobre a Engenharia de Software, como a área de atuação, a remuneração média de mercado, a possibilidade de crescimento, a satisfação dos atuantes na área de engenharia de software, entre outras notícias que pode captar a atenção do usuário. Estas notícias relevantes estarão divididas em contêineres como pode ser visto no protótipo, e em cada contêiner haverá um botão/link que permitirá o usuário acessar a camada 2.

Segunda Camada: A segunda camada se baseia na apresentação dos dados do curso, como disciplinas e áreas de conhecimento, em acordo com os tópicos do SWEBOOK V3.0. Tais informações serão exibidas em gráficos interativos. Assim como a camada 1, esta também distribuirá os dados apresentados em contêineres onde cada possuirá um meio de acesso para a camada 3.

Terceira Camada: A terceira camada mostrará resumos das ementas do PPC de modo menos formal. Com a finalidade de mostrar aos interessados as informações mais minuciosas de cada disciplina a ser cursada. Esta também fornecerá o documento oficial do PPC disponível no site do INF.

2.1.2 - Compartilhado

A pasta Compartilhado possuirá componentes que poderão ser utilizados em mais de uma das telas do sistema, componentes tais como:

- Carregando
- Menu pesquisa
- Página não encontrada
- Rodapé
- Search
- Sem conexão

2.2 - Assets

A pasta Assets conterà duas pastas, as pastas Img e Led, onde na pasta Img conseguimos encontrar imagens de ícones que serão utilizados em nosso projeto (ícones do facebook, google plus, entre outros), e na pasta Led (Large External Data) será o local onde iremos armazenar grandes quantidades de dados em formato de arquivo .JSON. Nesta pasta poderá ser encontrado por exemplo o arquivo PPC fragmentado de acordo com o padrão JSON, para ser utilizado como fonte de informação e pesquisa.