n-Educa

Projeto Arquitetural

There is guidance within this template that appears in a style named InfoBlue. This style has a hidden font attribute that allows you to toggle whether it is visible or hidden in this template. Use the Microsoft® Word® menu **Tools > Options > View > Hidden Text** check box to toggle this setting. There is also an option for printing: **Tools > Options > Print**.

# Objetivo

A arquitetura do projeto foi baseada na escalabilidade e na manutenção rápida, onde escolhemos desenvolver o projeto em plataforma Web, deixando-o em um único servidor, para disponibilizar aos clientes de forma fácil e rápida. E a manutenção ser modificado em um único local para todos os clientes simultaneamente.

# Metas Arquiteturais e Filosofia

A Arquitetura do será voltada para as nuvens, pensando na disponibilidade (implementação) rápida junto aos clientes, bastando somente ativar o registro da Escola que quiser o n-Educa, e também em agilidade de manutenção deste nosso produto.

Com base nesses quesitos, temos que garantir o acesso simultâneo e concorrente, e ao mesmo tempo garanto integridade dos dados, deixando o sistema seguro e dando, ou não, acesso de acordo com cada perfil do usuário.

# Premissas e Dependências

Para o desenvolvimento do n-Educa se faz necessário o uso de um servidor local para o desenvolvimento e teste do sistema. E em sua implementação, em servidor Web, é necessário que a Internet esteja disponível. Para a sua implantação é preciso de um Servidor Web configurado com PHP versão 5.3 ou superior e MySQL versão 5 ou superior, acesso via FTP.

Em seu desenvolvimento, o projeto está sendo construindo em cima do framework na linguagem PHP, Slim Framework, onde é totalmente baseado no REST (GET, POST, PUT e DELETE), retornando dados JSON, onde será consumido pelo AngularJS, na forma de integração com o HTML (layout do sistema).

# Requisitos Críticos da Arquitetura

Como o sistema é totalmente Web, precisa necessariamente de Internet para testar após sua implantação no Servidor Web. Também estar atualizado com o novo modelo de projeto RESTful, onde o sistema é totalmente baseado.

# Decisões, Restrições e Justificativas

* Utilização do REST, que faz uso das mensagem GET, POST, PUT e DELETE do HTTP, deixam o sistema distribuído mais veloz, através do JSON, invés do XML que carrega muita informação desnecessária;
* Fácil manutenção, onde alterando o código-fonte no servidor Web, já altera todo o sistema para todos os usuários (clientes);
* Uso do Slim Framework, baseado na linguagem PHP, para desenvolver a Aplicação, dando agilidade tanto no desenvolvimento, quanto na manutenção, do código-fonte do sistema.

# Principais Abstrações

Devido ao padrão REST, onde utilizamos o modelo RESTful, temos a liberdade de modularizar a aplicação em 3 distintas camadas: Visualização, Aplicação e Banco de Dados. Onde cada uma delas pode ser abstraidas, sendo trabalhandas de forma distintas.

# Camadas do Framework da Arquitetura

## Visualização – HTML, CSS e JavaScript com bibliotecas jQuery e AngularJS

## Aplicação – PHP com o framework de desenvolvimento Slim Framework

## Banco de Dados – MySQL

# Visões Arquiteturais

## Visualização

É uma camada de Visualização em HTML, CSS e JavaScirpt, com bibliotecas jQuery e AngularJS, que irá se relacionar com a Aplicação, através de JSON.

Devido a separação da camada de Visualização com a Aplicação, pode ser criada outras camada, como por exemplo, em Android, Objetive-C, .NET, C#/C++, entre quaisquer outras que consumam JSON.

## Aplicação

Responsável pelo recebimento dos dados vindos da Visualização (Layout), tratando de acordo com a necessidade, para ser inseridos, alterados ou removidos do Banco de Dados. E também pela captura dos dados armazenados no Banco de Dados e tratado para enviar para a camada de Visualização.

## Banco de Dados

Irá armazenar os dados de todo o sistema, dados esses que são tanto inerente ao sistema, quanto aos dados adicionados pelos usuários (clientes) do sistema.