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

Este documento descreve a filosofia, decisões, restrições justificativas, elementos significativos, e quaisquer outros aspectos gerais do sistema que dão forma ao projeto e implementação.   
[Sempre abordar secções 2 a 6 deste modelo. Outras seções são recomendadas, dependendo da quantidade de romance arquitetura, a quantidade de manutenção esperado, as habilidades da equipe de desenvolvimento, ea importância de outras preocupações arquitetônicas.]

# Metas Arquiteturais e Filosofia

A Arquitetura do será voltada para as nuvens, pensando na implementação rápida junto ao cliente, 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 concorrentes, 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, 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.

# Requisitos Críticos da Arquitetura

Como o sistema é totalmente Web, precisa necessariamente de Internet para testar após sua implantação no Servidor Web.

# Decisões, Restrições e Justificativas

[Liste as decisões que foram tomadas em relação as abordagens arquitetônicas e as restrições de serem colocados no caminho que os desenvolvedores a construir o sistema. Elas servirão como diretrizes para definir partes de arquitetura significativos do sistema. Justifique cada decisão ou restrição para que os desenvolvedores entendam a importância da construção do sistema de acordo com o contexto criado por essas decisões e restrições. Isso pode incluir uma lista de prós e contras para orientar os desenvolvedores na construção do sistema.]   
♣ decisão ou restrição e justificação   
♣ decisão ou restrição e justificação

# Mecanismos Arquiteturais

[Liste os mecanismos de arquitetura e descrever o estado atual de cada um. Inicialmente, cada mecanismo pode ser apenas o nome e uma breve descrição. Eles vão evoluir até que o mecanismo é uma colaboração ou padrão que podem ser aplicados diretamente a algum aspecto do projeto.]

## Mecanismo Arquitetural 1

[Descrever o objetivo, os atributos e função do mecanismo de arquitetura.]

## Mecanismo Arquitetural 2

[Descrever o objetivo, os atributos e função do mecanismo de arquitetura.]

# Principais Abstrações

[Lista e descrever brevemente as principais abstrações do sistema. Isto deve ser uma lista relativamente curta dos conceitos fundamentais que definem o sistema. Os principais abstrações costumam traduzir para as classes de análise iniciais e padrões importantes.]

# Camadas do Framework da Arquitetura

[Descrever o padrão de arquitetura que você vai usar ou como a arquitetura será consistente e uniforme. Esta poderia ser uma simples referência a um existente ou conhecido padrão de arquitetura, como a estrutura da camada, uma referência a um modelo de alto nível do quadro, ou uma descrição de como os principais componentes do sistema devem ser colocados juntos.]

# Visões Arquiteturais

[Descrever as visões de arquitetura que você vai usar para descrever a arquitetura de software. Isto ilustra as diferentes perspectivas que você vai disponibilizar para analisar e documentar decisões de arquitetura.]

## Visualizações recomendadas • Lógica: Descreve a estrutura eo comportamento de parcelas significativas da arquitetura do sistema. Isso pode incluir a estrutura do pacote, as interfaces críticas, classes e subsistemas importantes, e as relações entre esses elementos. Ele também inclui visualizações física e lógica de dados persistentes, se a persistência será incorporado ao sistema. Este é um subconjunto documentada do cartão. • Operacional: Descreve os nós físicos do sistema e os processos, segmentos e componentes que são executados nos nós físicos. Este ponto de vista não é necessário se o sistema funciona em um único processo e linha. • Caso de uso: Uma lista ou diagrama de casos de uso que contêm arquitetonicamente requisitos significativos.