<Amanita>

Documento de Arquitetura de Software

Versão <1.0>

Índice Analítico

1. Introdução 4

1.1 Finalidade 4

1.3 Definições, Acrônimos e Abreviações 4

1.5 Visão Geral 4

3. Metas e Restrições da Arquitetura 4

4. Visão de Casos de Uso 5

4.1 Realizações de Casos de Uso 5

5. Visão Lógica 5

5.1 Visão Geral 5

6. Visão de Processos 5

7. Visão de Implantação 5

8. Visão da Implementação 5

10. Tamanho e Desempenho 6

11. Qualidade 6

Documento de Arquitetura de Software

# Introdução

Aqui detalharemos o software junto dos nossos métodos de aplicação para o sistema, estará nossa visão lógica e passos necessários para o devido funcionamento no estabelecimento do cliente.

## Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema.

## Definições, Acrônimos e Abreviações

EMIN – Estoque mínimo;

EMAX – Estoque máximo;

ON – Produto ativo;

OFF – Produto inativo;

## Visão Geral

Vamos detalhar como o sistema como software se comporta na máquina do cliente, explicar como ele é arquitetado, apresentar suas características e detalhar as mesmas. Como o funcionar do estabelecimento e seus funcionários impactaram na forma em que o Amanita foi desenvolvido para ser tanto eficiente quanto simples para o operante.

# Metas e Restrições da Arquitetura

A aplicação foi desenvolvida inteiramente para uso único do estabelecimento do proprietário, não será feita nenhuma chave ou licenciamento para que seja distribuído. O sistema não precisa de nenhum suporte correlacionado e não precisa de conexão com a internet para funcionar. Ou seja, o nível de segurança do sistema é alto pois além de não ter perigo de dados serem interceptados na rede, também não tem nenhum dado relativamente secreto que possa ser roubado já que não há a necessidade de conexão do sistema coma internet.

# Visão de Casos de Uso

[Esta seção lista casos de uso ou cenários do modelo de casos de uso quando eles representam funcionalidade central e significativa do sistema final ou, quando têm uma grande cobertura arquitetural — eles experimentam muitos elementos arquiteturais ou quando enfatizam ou ilustram um ponto complexo e específico da arquitetura.]

## Realizações de Casos de Uso

[Esta seção ilustra o funcionamento do software, apresentando algumas realizações (ou cenários) de cas/ Visão Geral os de uso selecionadas e explica como os diversos elementos do modelo de design contribuem para a respectiva funcionalidade.]

# Visão Lógica

Nossa visão lógica foi que, para desenvolvermos um sistema onde ao levantarmos dados descobrirmos que alguns dos funcionários não possui nível de alfabetização básico, o sistema deve ser o mais simples possível para que seja inclusivo, mesmo que o operador possa não saber ler, a associação ainda sim deve ser feita, para uma alteração simples. Então nossa interface acima de tudo precisa ser simples e didática, que foi a linha que seguimos. Com botões intuitivos e de fácil associação.

## Visão Geral

Assim que o sistema é aberto, ele já é redirecionado para a página principal (e única) do sistema. Onde está todas as informações necessárias. A lista com todos os produtos cadastrados, com suas respectivas categorias e singularidades (se é refrigerado, quantidade, valor, etc), na página também está todos os botões de comando, sendo eles: adicionar, alterar, excluir e on/off.

# Visão de Processos

[Esta seção descreve a decomposição do sistema em processos leves (threads simples de controle) e processos pesados (agrupamentos de processos leves). Organize a seção em grupos de processos que se comunicam ou interagem. Descreva os modos principais de comunicação entre processos, como transmissão de mensagens e interrupções.]

# Visão de Implantação

Para que o software seja implantado e executado com sucesso, não será necessário muito. Como o sistema é leve, compacto e elaborado para que seja o mais simples possível, qualquer máquina que seja superior a um i3, com um Windows 7 ou superior, rodará o software sem problemas. Assim como na instalação, que será rápida e sem complicações. Não será necessário conexão com banco de dados, não será necessária intervenção de outras máquinas, o sistema rodará por si mesmo, e a máquina local servirá como base para armazenamento das informações contidas no software.

# Visão da Implementação

[Esta seção descreve a estrutura geral do modelo de implementação, a divisão do software em camadas e os subsistemas no modelo de implementação e todos os componentes significativos do ponto de vista da arquitetura.]

# Tamanho e Desempenho

O software é compacto e leve, não exige um processador potente para que seja executado sem problemas. É recomendado que o cliente tenha um sistema operacional atualizado para melhor experiência do próprio usuário, mas se não for o caso, o sistema também irá rodar normalmente. O tamanho da aplicação não é grande coisa, mas pode aumentar consideravelmente dependendo de quantos produtos sejam alimentados ao sistema, podendo variar da base (100mb) até o infinito conforme o usuário vai adicionando itens.

# Qualidade

Não é porque o software foi desenvolvido para ser simples, leve e compacto que ele peca em suas funcionalidades, ele cumpre o papel que foi desenvolvido para cumprir que é gerenciar um sistema de estoque, seja ele qual for.

Há portabilidade, pois, se instalado em outros sistemas operacionais, ou em outras versões, ele rodará sem grandes problemas, é recomendável que a plataforma do usuário seja atualizada pois o software foi desenvolvido e testado em plataformas atualizadas, e para melhor desempenho, deve ser rodado no mesmo.

Não há implicações de segurança pois seus dados não necessitam de intervenção de nada além da própria máquina onde está alocado, não há envio de dados que possam ser interceptados, assim como não há dados “úteis” que possam ser roubados, já que no sistema não é alocado nenhum tipo de dado que seja confidencial.