Falcon Framework

Versão <0.1>

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 25/03/2016 | 0.1 | Adicionado Introdução e Funcionalidades | Thiago |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice Analítico

1. Introdução 4

1.1 Definições, Acrônimos e Abreviações 4

1.2 Referências 4

1.3 Visão Geral 4

2. Funcionalidade 5

Atores 6

2.1 RF1 - Modelar um banco de dados 6

2.2 RF2 -Visualizar um modelo cadastrado 6

2.3 RF3 - Gerar código Cliente Servidor 6

2.3.1 RF 3.1 - Gerar código do servidor 7

2.3.2 RF 3.2 - Gerar código DAO do cliente 7

2.3.3 RF 3.3 - Gerar SDK sincronização do cliente. 7

2.3.4 RF 3.4 - Gerar SDK de RESTFUL do cliente. 7

2.4 RF 4 - Versionar Código gerado 7

2.5 RF 5 - Efetuar Login 7

3. Usabilidade 7

3.1 <Requisito de Usabilidade Um> 7

4. Confiabilidade 7

4.1 <Requisito de Confiabilidade Um> 8

5. Desempenho 8

5.1 <Requisito de Desempenho Um> 8

6. Suportabilidade 8

6.1 <Requisito de Suportabilidade Um> 8

7. Restrições de Design 8

7.1 <Restrição de Design Um> 8

8. Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line 8

9. Componentes Adquiridos 8

10. Interfaces 8

10.1 Interfaces do Usuário 8

10.2 Interfaces de Hardware 9

10.3 Interfaces de Software 9

10.4 Interfaces de Comunicação 9

11. Requisitos de Licenciamento 9

12. Observações Legais, de Copyright e Outras 9

13. Padrões Aplicáveis 9

# Introdução

A Especificação Suplementar visa introduzir itens extraordinários, além dos não detectados anteriormente nas demais Especificações utilizadas na documentação do Falcon Framework

Entre os requisitos estão incluídos:

* Requisitos funcionais
* Atributos de qualidade Falcon Framework a serem criados, incluindo requisitos de usabilidade, confiabilidade, desempenho e suportabilidade.
* Outros requisitos, como sistemas operacionais e ambientes, requisitos de compatibilidade e restrições de design.

## Definições, Acrônimos e Abreviações

Definições, Acrônimos e Abreviações vide Artefato Glossário, volume pertencente ao âmbito de documentações (artefatos) do Falcon Framework.

## Referências

[Esta subseção fornece uma lista completa de todos os documentos mencionados na **Especificação Suplementar**. Identifique cada documento por título, número do relatório (se aplicável), data e organização de publicação. Especifique as fontes a partir das quais as referências podem ser obtidas. Essas informações podem ser fornecidas por um anexo ou outro documento.]

## Visão Geral

[Esta subseção descreve o que o restante da **Especificação Suplementar** contém e explica como o documento está organizado.]

Esta subsessão descreve de forma sucinta, a função de cada sessão deste documento.

1. **Introdução** descreve função da Especificação Suplementar, no que tange ao Falcaon Framework em visão geral.

Na seqüência, os tópicos estão dispostos em:

1. **Funcionalidades**: descreve os Requisitos Funcionais do Falcon Framework, que são expressos em linguagem natural.
2. **Usabilidade**: inclui os requisitos que afetam a usabilidade, tais como períodos mensuráveis de tarefas típicas, entre outros.
3. **Confiabilidade**: descreve itens como disponibilidade bem com outros itens relevantes de confiabilidade.
4. **Desempenho**: descreve características de desempenho do Falcon Framework, como tempo de resposta de uma transação, taxa de transferência. Há também outros recursos que podem afetar o desempenho, como memória, disco, comunicação, etc.
5. **Suportabilidade**: indica os requisitos que aprimorarão a suportabilidade e a manutenabilidade do Falcon Framwork, incluindo padrões de codificação, convenções de nomeação, bibliotecas de classes, acesso à manutenção e utilitários de manutenção.
6. **Restrições de Designer**: indica todas as restrições de design referentes ao Falcon Framework que está sendo criado. As restrições de design englobam restrições quanto à linguagem de software, requisitos de processo de software, restrições de arquitetura e bibliotecas de classe.
7. **Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line:** indicam os sistemas de ajuda ao usuário, observações de ajuda e manuais de auxílio.
8. **Componentes Adquiridos:** descreve os componentes adquiridos a serem utilizados no Falcon Framework, ou que influam de forma substancial no seu funcionamento.
9. **Interfaces:** define as interfaces que devem ser suportadas pelo aplicativo. Deve conter especificidades, protocolos, portas e endereços lógicos adequados, para que o Software possa ser desenvolvido e verificado em relação aos requisitos de interface.
10. **Requisitos de Licenciamento:** define todos os avisos legais necessários, garantias, observações sobre direitos autorais, observações sobre patente.
11. **Padrões Aplicáveis:** descreve, por meio de referências, os padrões aplicáveis e as seções específicas destes que se aplicam ao Falcon Framweork. Este tópico tem utilidade para futuras certificações que possam ser requeridas no futuro.

# Funcionalidade

Nesta seção sera ilustrado o diagrama de casos de uso e uma especificação do mesmo, das funcionalidades do Falcon Framework. Será ilustrado um diagram geral, e nas próximas subsessões haverá uma especificação de cada caso de uso.



## Atores

|  |  |  |
| --- | --- | --- |
| **Nome** | **Descrição** | **Casos de uso que interage** |
| Desenvolvedor | Usuário que interage com o sistema com o objetivo de gerar um código para sua persistencia de banco online e offline. | * Visualizar um modelo cadastrado * Modelar um banco de dados * Gerar código Cliente Servidor * Versionar Código gerado |
| Falcon Framework System | Sistema terceiro responsável por gerar o código solicitado pelo desenvolvedor. | * Gerar código Cliente Servidor * Versionar Código gerado |

## RF1 - Modelar um banco de dados

Usuário podelar modelar o banco de dados que deseja persistir no client-servidor

## RF2 -Visualizar um modelo cadastrado

Usuário podela visualizar alguem modelo de banco de dados já cadastrado.

## RF3 - Gerar código Cliente Servidor

Usuário poderá gerar o código cliente servidor, escolhendo qual tecnologia deseja que o código seja gerado. Dependendo da tecnologia o RF3 se desmembrará em algum dos próximos.

## RF 3.1 - Gerar código do servidor

Gerará o código de servidor em NodeJS ou Ruby on Rails, baseado na Modelagem do desenvolvedor, criando os endpoints do padrão RESTFUL.

## RF 3.2 - Gerar código DAO do cliente

Gerará um banco de dados em iOS ou Android, já com a DAO e Model para acessar essas informações

## RF 3.3 - Gerar SDK sincronização do cliente.

Gerará um SDK contendo a lógica para persistir os dados offline e online no cliente Android ou iOS.

## RF 3.4 - Gerar SDK de RESTFUL do cliente.

Gerarará um SDK para fazer requisições no servidor gerado, já com as operações básicas de RESTFUL.

## RF 4 - Versionar Código gerado

O usuário receberá o código gerado do sistema Falcon que sera guardado como uma build no histórico do sistema.

## RF 5 - Efetuar Login

O usuário logará no sistema para poder ter suas modelagens e histórico de builds salvas.

[A descrição do requisito deve ser feita aqui.]

# Usabilidade

[Esta seção deve incluir todos os requisitos que afetam a usabilidade. Eis alguns exemplos:

* especifique o tempo de treinamento necessário para que usuários normais e usuários com conhecimentos avançados se tornem produtivos em operações específicas
* especifique períodos de tempo mensuráveis para tarefas típicas ou
* especifique requisitos que estejam em conformidade com padrões de usabilidade comuns (como, por exemplo, os padrões CUA da IBM ou os padrões GUI da Microsoft)]

## <Requisito de Usabilidade Um>

A descrição do requisito.

# Confiabilidade

[Os requisitos de confiabilidade do sistema devem ser especificados aqui. Eis algumas sugestões:

* Disponibilidade – especifique a porcentagem de tempo disponível ( xx.xx%), as horas de uso, o acesso à manutenção, as operações de modo degradado, etc.
* Tempo Médio entre Falhas (MTBF) – normalmente especificado em horas, mas também poderá ser especificado em termos de dias, meses ou anos.
* Tempo Médio para Reparo (MTTR) – quanto tempo o sistema poderá ficar sem funcionar após uma falha?
* Exatidão – especifique a precisão (resolução) e exatidão (através de algum padrão conhecido) necessárias na saída dos sistemas.
* Taxa máxima de erros ou defeitos – geralmente expressa em termos de erros/KLOC (thousands of lines of code, milhares de linhas de código) ou de erros/ponto de função.
* Taxa de erros ou defeitos – categorizada em termos de erros de pouca, média ou muita importância: os requisitos devem definir o que se entende por erro “crítico”(por exemplo, perda total de dados ou total incapacidade de usar determinadas partes da funcionalidade do sistema).]

## <Requisito de Confiabilidade Um>

[A descrição do requisito deve ser feita aqui.]

# Desempenho

[As características de desempenho do sistema devem ser descritas nesta seção. Inclua tempos de resposta específicos. Quando aplicável, faça referência, por nome, aos Casos de Uso relacionados.

* Tempo de resposta de uma transação (médio, máximo)
* Taxa de transferência (por exemplo, transações por segundo)
* Capacidade( por exemplo, o número de clientes ou de transações que o sistema pode acomodar)
* Modos de degradação (o modo aceitável de operação quando o sistema tiver sido degradado de alguma maneira)
* Uso de recursos: memória, disco, comunicações, etc]

## <Requisito de Desempenho Um>

[A descrição do requisito deve ser feita aqui.]

# Suportabilidade

[Esta seção indica todos os requisitos que aprimorarão a suportabilidade ou manutenibilidade do sistema que está sendo criado, incluindo padrões de codificação, convenções de nomeação, bibliotecas de classes, acesso à manutenção e utilitários de manutenção.]

## <Requisito de Suportabilidade Um>

[A descrição do requisito deve ser feita aqui.]

# Restrições de Design

[Esta seção deve indicar todas as restrições de design referentes ao sistema que está sendo criado. As restrições de design representam decisões de design que foram impostas e devem ser obedecidas. Entre os exemplos desse tipo de restrição estão linguagens de software, requisitos de processo de software, uso prescrito de ferramentas de desenvolvimento, restrições de design e de arquitetura, componentes comprados, bibliotecas de classes, etc.]

## <Restrição de Design Um>

[A descrição do requisito deve ser feita aqui.]

# Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line

[Descreve os requisitos (se houver) de documentação de usuário on-line, sistemas de ajuda, observações sobre ajuda, etc.]

# Componentes Adquiridos

[Esta seção descreve todos os componentes adquiridos a serem usados no sistema, restrições de utilização ou de licenciamento aplicáveis e quaisquer padrões associados de compatibilidade/interoperabilidade ou de interface.]

# Interfaces

[Esta seção define as interfaces que devem ser suportadas pelo aplicativo. Ela deve conter especificidades, protocolos, portas e endereços lógicos adequados, entre outros, para que o software possa ser desenvolvido e verificado em relação aos requisitos de interface.]

## Interfaces do Usuário

[Descreva as interfaces de usuário que deverão ser implementadas pelo software.]

## Interfaces de Hardware

[Esta seção define todas as interfaces de hardware que devem ser suportadas pelo software, incluindo a estrutura lógica, os endereços físicos, o comportamento esperado, etc.]

## Interfaces de Software

[Esta seção descreve as interfaces de software para outros componentes do sistema de software. Podem ser componentes adquiridos, componentes reutilizados de outro aplicativo ou componentes que estão sendo desenvolvidos para subsistemas fora do escopo desta SRS, mas com os quais esse aplicativo de software deve interagir.]

## Interfaces de Comunicação

[Descreva todas as interfaces de comunicação com outros sistemas ou dispositivos como, por exemplo, redes locais, dispositivos seriais remotos, etc.]

# Requisitos de Licenciamento

[Esta seção define todos os requisitos de imposição de licenciamento ou outros requisitos de restrição de utilização que devem ser exibidos pelo software.]

# Observações Legais, de Copyright e Outras

[Esta seção descreve todos os avisos legais necessários, garantias, observações sobre direitos autorais, observações sobre patentes, logomarcas, marcas comerciais ou problemas de conformidade com logotipos referentes ao software.]

# Padrões Aplicáveis

[Esta seção descreve, por meio de referências, todos os padrões aplicáveis e as seções específicas desses padrões que se aplicam ao sistema que está sendo descrito. Entre esses padrões estão incluídos, por exemplo, padrões legais, de qualidade e reguladores, padrões de indústria referentes à usabilidade, interoperabilidade, internacionalização, compatibilidade com o sistema operacional, etc.]