

Pedal Pi - Documento de Requisitos Incremento um

de Pedal Controller Projects
especificado por Paulo Mateus Moura da Silva

Versão 1.1.0
01/04/2016

1. Prefácio

Como forma de padronizar o processo de desenvolvimento do Pedal Pi, este documento foi desenvolvido e está acessível para todos os leitores, sejam estes clientes (instrumentistas) e desenvolvedores (colaboradores, seja, engenheiros de sistema, engenheiros de teste, engenheiros de manutenção ou especialistas em marketing e divulgação).

Versão 1.0.0

18/03/2016 - Paulo Mateus Moura da Silva

Em sua versão inicial, o documento de requisitos concentra-se em aspectos básicos, visando o desenvolvimento contínuo incremental de forma sustentável. Em relação ao uso, a entrega de um produto viável mínimo que ofereça um fácil controle inicial em forma de *WebService* também é vislumbrada. Por fim, metas de segurança, praticidade e uso são definidas.

Versão 1.1.0

01/04/2016 - Paulo Mateus Moura da Silva

Nesta versão, foram acrescentados os requisitos referentes ao controle do Pedal Pi por aplicações controladoras. Como esta camada é o ponto de gerenciamento (na visão do usuário final), os casos de uso serão derivados principalmente da seção 4.2.3, mas abrangerão também os outros requisitos.

Futuras versões

Futuras versões deverão ser lançadas para aprofundar os requisitos de sessão de usuário, instalação de efeitos, automatização na atualização de sistema, suporte a periféricos.

4. Requisitos

Requisitos funcionais são descritos em uma estrutura inspirada no *Modelo SISP: Lista de Requisitos v0.2*. A apresentação deste será conforme o seguinte padrão:

[CATEGORIZAÇÃO DO REQUISITO - SIGLA - NUMERO]	<i>Uma descrição sobre o requisito</i> Prioridade: <i>Classificação da prioridade, seja Essencial (alta) ou Desejável (média).</i> Observações: <i>(Somente se houver) Informações adicionais, como requisitos com que se relaciona e justificativa de sua existência.</i>
--	--

Onde:

- **[CATEGORIZAÇÃO DO REQUISITO - SIGLA - NUMERO]:** Nome do requisito, composto pelos campos:
 - **Categorização do Requisito:** RF (Requisito funcional) ou RNF (requisito não funcional);
 - **Sigla:** Sigla do tipo de requisito (o tipo de requisito será descrito na subseção no qual o requisito se encontra);
 - **Número:** Número do requisito.

Requisitos não funcionais são descritos de forma similar aos Requisitos funcionais exceto pelo campo **Prioridade**, que torna-se opcional, tendo em vista que **todos são Essenciais**, exceto os ditos de forma explícita.

Neste documento, não há uma clara diferenciação de *Requisitos de Usuário* e *Requisitos de Sistema*. Pela simplicidade da aplicação, foram tomados esforços para que os requisitos fossem descritos com o nível mais alto possível, de forma em que clientes e gerentes possam entender, mas com nível de detalhes o suficiente para que o documento também dê suporte as desenvolvedores e analistas.

Tendo ainda em vista a divisão da aplicação em partes como *WebService* (Seção 4.2.1) e *Aplicação controladora* (Seção 4.2.3), perceberá uma aparente duplicação de Requisitos Funcionais. Entretanto, lembre-se que cada requisito trás um valor de agregação na camada no qual este se enquadra e as camadas se relacionam entre si. O projeto como um todo será evidente nas definições de Caso de Uso.

4.1 Requisitos não funcionais

4.1.5 Requisitos não funcionais de Usabilidade

[RNF - USA - 01] Para usuários experientes, caso estes queiram utilizar o equipamento como um computador de propósito geral, a case que envolve o hardware não deve bloquear entradas *HDMI* e *USB*.

[RNF - USA - 02] A interface de edição de *patch* deve ser usável de modo em que usuários pouco familiarizados com o sistema, mas acostumados com a abstração dos efeitos e com a interface de dispositivos móveis deverão em menos de 10 minutos compreender a organização dos sistema.

Observações

Requisitos não funcionais de Usabilidade relacionados ao controle do dispositivo por periféricos serão introduzidas em documentos posteriores.

4.2 Requisitos Funcionais

4.2.3 Requisitos funcionais - Aplicação controladora

[RF - AC - 01] Deverão existir duas seções na aplicação: Pedal Pi e Local, de forma em que a primeira seção possibilite edições diretamente no dispositivo e Local permita fazer cópia de segurança no dispositivo no qual a aplicação está instalada.

Prioridade: Desejável

Observação: Em versões futuras, poderá habilitar a edição *offline*, ou seja, poderá ser possível a edição de efeitos estando desconectado do dispositivo.

[RF - AC - 02] A aplicação deve pesquisar os dispositivos conectados de forma automática quando iniciado e estabelecer a conexão de forma automática

Prioridade: Desejável

[RF - AC - 03] Listagem dos bancos - juntamente com seus *patches* - já persistidos no Pedal Pi deverão ser exibidos na seção da

aplicação Pedal Pi. (Ver [RF - WS - 03], [RF - WS - 04] e [RF - AC - 01])

Prioridade: Essencial

[RF - AC - 04]

A seleção de um *patch* deve abrir a visão de edição, onde mostra todas as informações referentes a ele (ver [RF - WS - 05]). O usuário poderá intuitivamente (ver [RNF - USA - 02]) editar (ver [RF - AC - 05]), remover e fazer uma cópia de segurança (descrita em [RF - AC - 07]).

Prioridade: Essencial

Observação: Ver também [RF - WS - 07]

[RF - AC - 05]

A edição de determinado patch ([RF - WS - 07]) deverá incluir a adição de novos efeitos (obtidos por [RF - WS - 02] e respeitando as restrições descritas em [RF - WS - 08]), troca de ordem dos efeitos (não implementado no WS), alteração dos valores dos parâmetros ([RF - WS - 10]), troca de estado dos efeitos ([RF - WS - 09]).

Prioridade: Essencial

Observação: A conexão entre efeitos, que também faz parte da edição do *patch*, será descrita em [RF - AC - 06].

[RF - AC - 06]

As conexões entre efeitos ([RF - WS - 11]) deverão ser realizadas de forma automática, conforme a ordem dos efeitos.

Prioridade: Essencial

Observação: Um gerenciamento de conexões manuais fica para futuras versões (com prioridade Desejável) ou para aplicações controladoras mantidas pela comunidade.

[RF - AC - 07]

É desejável que seja possível a realização de cópias de segurança de *patches* e bancos existentes na área Pedal Pi nas telas de seleção de efeito ([RF - AC - 04]), seleção de banco ([RF - AC - 08]) e edição de efeito ([RF - AC - 05]).

Para cópias de segurança de *patches*, o usuário deverá escolher para qual banco localizado na área Local ([RF - AC - 01]) este irá ser armazenado. Para cópias de segurança de um banco, o banco todo será copiado para a área Local.

Prioridade: Desejável

Observação: Nomes conflitantes deverão ser resolvidos pelo usuário

[RF - AC - 08]

Ao selecionar um *patch* para edição, o banco deste *patch* será automaticamente o *banco em uso* ([RF - WS - 12]) e o patch editado será marcado como patch atual ([RF - WS - 13]).

Prioridade: Desejável

Observação: É necessária validação com o cliente através da prototipação, de forma que este requisito respeite [RNF - USA - 02].

[RF - AC - 09]

Na seção Pedal Pi da aplicação (ver [RF - AC - 01]), o sistema possibilitará o gerenciamento dos bancos. Ou seja, a criação de bancos, visualização de seus *patches* ([RF - AC - 03]), remoção de bancos da memória do Pedal Pi e criação de cópias de segurança ([RF - AC - 07]).

Prioridade: Essencial

Observação: Para mais detalhes, veja também [RF - WS - 06].

Requisitos funcionais relacionados a configuração de periféricos pela aplicação controladora serão contemplados na versão posterior deste documento que incluir Conexão de periféricos controladores.