Integrações para app

**Life Saver**



Versão 1.0

Índice

[1. Sobre este documento 4](#_Toc109647629)

[1.1. Introdução 4](#_Toc109647630)

[1.2. Objetivos 4](#_Toc109647631)

[1.3. Tecnologias 4](#_Toc109647632)

[2. Definições do projeto 5](#_Toc109647633)

[2.1. Introdução 5](#_Toc109647634)

[2.2. Constantes 5](#_Toc109647635)

[2.3. Enumerações 5](#_Toc109647636)

[3. Módulos 6](#_Toc109647637)

[3.1. Autorização 6](#_Toc109647638)

[3.2. Configuração 6](#_Toc109647639)

[4. Autorização – Registro 7](#_Toc109647640)

[4.1. Objetivo 7](#_Toc109647641)

[4.2. Backend 7](#_Toc109647642)

[5. Autorização – Confirmação de registro 8](#_Toc109647643)

[5.1. Objetivo 8](#_Toc109647644)

[5.2. Backend 8](#_Toc109647645)

[6. Autorização – Reenviar token de registro 9](#_Toc109647646)

[6.1. Objetivo 9](#_Toc109647647)

[6.2. Backend 9](#_Toc109647648)

[7. Autorização – Esqueceu sua senha 10](#_Toc109647649)

[7.1. Objetivo 10](#_Toc109647650)

[7.2. Backend 10](#_Toc109647651)

[8. Autorização – Confirmação de troca de senha 11](#_Toc109647652)

[8.1. Objetivo 11](#_Toc109647653)

[8.2. Backend 11](#_Toc109647654)

[9. Autorização – Login do usuário 12](#_Toc109647655)

[9.1. Objetivo 12](#_Toc109647656)

[9.2. Backend 12](#_Toc109647657)

[10. Configuração – Criação de pasta 13](#_Toc109647658)

[10.1. Objetivo 13](#_Toc109647659)

[10.2. Backend 13](#_Toc109647660)

[11. Configuração – Obter pasta 14](#_Toc109647661)

[11.1. Objetivo 14](#_Toc109647662)

[11.2. Backend 14](#_Toc109647663)

[12. Configuração – Listagem de pastas 15](#_Toc109647664)

[12.1. Objetivo 15](#_Toc109647665)

[12.2. Backend 15](#_Toc109647666)

[13. Configuração – Edição de pasta 16](#_Toc109647667)

[13.1. Objetivo 16](#_Toc109647668)

[13.2. Backend 16](#_Toc109647669)

[14. Configuração – Novo item demanda financeira 17](#_Toc109647670)

[14.1. Objetivo 17](#_Toc109647671)

[14.2. Backend 17](#_Toc109647672)

[15. Configuração – Obter item demanda financeira 18](#_Toc109647673)

[15.1. Objetivo 18](#_Toc109647674)

[15.2. Backend 18](#_Toc109647675)

[16. Configuração – Listagem de itens de uma pasta 19](#_Toc109647676)

[16.1. Objetivo 19](#_Toc109647677)

[16.2. Backend 19](#_Toc109647678)

# Sobre este documento

## Introdução

Neste documento teremos as descrições das integrações entre todas as partes envolvidas para a entrega do portal **LIFE SAVER** da startup BuildingBOX.

## Objetivos

* Fornecer ao usuário um app integrado com toda sua inteligência financeira.
* Disponibilizaremos controles de entradas, saidas e limites em seu mobile;
* Haverá disparos de SMS quando algum “evento” acontecer;
  + Exemplos de eventos:
    - Total do mês previsto para gasolina excedeu em “300 reais”;
    - Total diário para comida excedido;
    - Condomínio este mês acima do padrão;
* Categorização e gráficos de gastos para visualização da evolução;
* A idéia é a pessoa sempre saber se pode ou não gastar / investir em qualquer momento;

## Tecnologias

O portal de instalação está divido entre:

* **NodeJS, JS** para front-end webApp;
* **Net Core 5** para processamento de backend;
* **PostgreSQL** para banco de dados (v14);

# Definições do projeto

## Introdução

Este documento contempla todos os módulos abaixo:

* Autorização;
* Configuração;

## Constantes

codeExpirationMinutes 5

## Enumerações

**TimePeriod**daily = 1, weekly = 2, monthly =3, yearly = 4, random = 5

# Módulos

## Autorização

Neste módulo teremos a sequencia abaixo de eventos:

* Registro;
* Re-envio de token ao mobile (registro);
* Confirmação de registro;
* Esqueceu sua senha;
* Re-envio de token ao mobile (senha);
* Confirmação de troca de senha;
* Login do usuário;

## Configuração

Neste módulo teremos a sequencia abaixo de eventos:

* Criação de pasta (ou subpasta) de itens;
* Obter pasta de itens;
* Listagem de todas as subpastas de uma pasta;
* Edição de uma pasta;
* Criação de um item de demanda financeira;
* Obter dados de um item;
* Listagem de itens de uma pasta;
* Edição de um item;
* Registrar resultado (drop) de um item;
* Listar resultados de uma pasta;
* Editar drop de item;
* Remover drop de item;

# Autorização – Registro

## Objetivo

Cadastrar, de forma gratuita, um usuário no aplicativo.  
O novo usuário receberá um SMS em seu mobile para verificação de segurança.

## Backend

**URL**/api/v1/auth/register

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json | {  “name”  “email”  “mobile”  “password”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Usuário criado com sucesso. |  |
| (400) | Name Email Mobile  Password | Nome e sobrenome requeridos Precisa de @ e dominio Somente números e acima de 10 digitos Telefone não pode já estar cadastrado No minimo 6 digitos quaisquer |

# Autorização – Confirmação de registro

## Objetivo

Finalizar a jornada de onboarding através do código recebido no mobile.  
O usuário receberá em seu email uma mensagem de boas vindas.

## Backend

**URL**/api/v1/auth/register\_confirm

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json | {  “mobile”  “code”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Usuário registrado com sucesso. |  |
| (400) | Mobile  Code | Somente números e acima de 10 digitos Telefone precisa estar cadastrado Precisa ser um número de 6 digitos Código não pode ter já expirado |

# Autorização – Reenviar token de registro

## Objetivo

Caso o usuário tenha alguma dificuldade durante o tempo entre o envio do SMS com o código e o tempo de expiração, haverá uma outra opção de re-envio do mesmo.

## Backend

**URL**/api/v1/auth/resend\_code\_register

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json | {  “mobile”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Token re-enviado |  |
| (400) | Mobile | Somente números e acima de 10 digitos Telefone precisa estar cadastrado |

# Autorização – Esqueceu sua senha

## Objetivo

Caso o usuário não lembre sua senha, ele poderá acionar um botão de ajuda.

## Backend

**URL**/api/v1/auth/forgot\_pass

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json | {  “mobile”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Token re-enviado |  |
| (400) | Mobile | Somente números e acima de 10 digitos Telefone precisa estar cadastrado |

# Autorização – Confirmação de troca de senha

## Objetivo

Depois de receber o token no celular, o usuário poderá informar uma nova senha.

## Backend

**URL**/api/v1/auth/forgot\_pass\_confirm

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json | {  “mobile”  “codigo”  “password”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Usuário registrado com sucesso. |  |
| (400) | Mobile  Codigo  Password | Somente números e acima de 10 digitos Telefone precisa estar cadastrado Somente números de 6 dígitos  No mínimo 6 digitos quaisquer |

# Autorização – Login do usuário

## Objetivo

Após o cadastro completo, o usuário poderá logar no aplicativo.

## Backend

**URL**/api/v1/auth/login

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json | {  “email”  “password”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Usuário logado com sucesso | {  “token”  } |
| (400) | Email    Password | Precisa de @ e dominio Precisa estar cadastrado Precisa estar ativo Precisa senha estar de acordo No mínimo 6 digitos quaisquer |

# Configuração – Criação de pasta

## Objetivo

Criar uma pasta ou subpasta financeira de interesse, marcando-a como “recebimento” (salário por exemplo) ou não.

## Backend

**URL**/api/v1/config/folder\_add

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “fkFolder”  “name”  “income”  } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Pasta criada com sucesso |  |
| (400) | fkFolder  Name  Income | “Null” para pasta inicial  Numero para subpasta  Nome da pasta válido (1 e 30 caracteres) Nome ainda nao utilizado pelo usuário  True ou False para recebimentos  (itens vão herdar este parâmetro) |

# Configuração – Obter pasta

## Objetivo

Obter dados de uma pasta.

## Backend

**URL**/api/v1/config/folder\_get

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “id” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Detalhes da pasta | {  “result”  {  “name”  }  } |
| (400) | fkFolder  Name  Income | “Null” para pasta inicial  Numero para subpasta  Nome da pasta válido (1 e 30 caracteres) Nome ainda nao utilizado pelo usuário  True ou False para recebimentos  (itens vão herdar este parâmetro) |

# Configuração – Listagem de pastas

## Objetivo

Listagem de todas as subpastas de um numero de pasta.  
Usar “null” para primeiro nivel.

## Backend

**URL**/api/v1/config/folder\_list

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json Authorization (Bearer ‘token’) | {  “fkFolder” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Listagem de todas as subpastas em um nivel | {  “results”:   [  {  “id”,   “name”  }  ]  } |
| (400) | fkFolder | Numero invalido para subpasta |

# Configuração – Edição de pasta

## Objetivo

Editar uma pasta, trocando o seu nome (deverá ser unico).

## Backend

**URL**/api/v1/config/folder\_edit

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “id”  “new\_name”  “new\_timePeriod”  “new\_standardValue” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Pasta renomeada com sucesso |  |
| (400) | Id  New\_name | Numero de pasta inválido  Novo nome já existe no diretório atual |

# Configuração – Novo item demanda financeira

## Objetivo

Criar uma nova categoria de registros para demandas financeiras.

## Backend

**URL**/api/v1/config/item\_add

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “fkFolder”  “name”  “timePeriod”  “standardValue” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Item financeiro criado com sucesso |  |
| (400) |  |  |

# Configuração – Obter item demanda financeira

## Objetivo

Obter os dados de um item financeiro.

## Backend

**URL**/api/v1/config/item\_get

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “id” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Dados do item | {  “result”  {  “id”,  “fkFolder”,  “date”,   “income”,  “name”,  }  } |
| (400) |  |  |

# Configuração – Editar item demanda financeira

## Objetivo

Editar uma categoria de registros para demandas financeiras.

## Backend

**URL**/api/v1/config/item\_edit

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “id”  “new\_name”  “new\_timePeriod”  “new\_standardValue” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Item financeiro editado com sucesso |  |
| (400) |  |  |

# Configuração – Listagem de itens de uma pasta

## Objetivo

Listar todos os itens de uma pasta.

## Backend

**URL**/api/v1/config/item\_listing

|  |  |  |
| --- | --- | --- |
| Método | Header | Body |
| POST | Content-Typeapplication/json  Authorization (Bearer ‘token’) | {  “fkFolder” } |

|  |  |  |
| --- | --- | --- |
| Retorno | Observação | Detalhes |
| (200) | Listagem completa | {  “results” : [  {  “id”,  “name”,  “timePeriod”,  “standardValue”  },  ]  } |
| (400) |  |  |