|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Operação** | **Inputs** | | | **Output** | **Descrição** |
| **Inscrição** | **Nome do campo** | **Tipo de Dados/ Tamanho** | **Valores Permitidos** | * Sucesso/Erro:   + Devolve os dados da classe **PrimaveraResultStructure();**   Em caso de sucesso:   1. Deverá enviar o código de sucesso como forma de confirmação.   Em caso de erro:   1. Deverá enviar o código do erro, e fazer um *rollback* das operações e apresentar a tela de erro; 2. Mandar o erro para uma ***hashTable*** sem afectar a operação no Fenix.   **Resumindo:**  1° Grava no Fenix;  2° Envia para o Primavera;  3° Output: Sucesso ou Erro.  (Em caso de erro, seguir a descrição acima). | * Esta operação será responsável por inscrever um aluno através do método **inscreverAluno(<*inputs listados*>)**; * Será necessário o preenchimento dos campos obrigatórios; * Será necessário a validação dos campos, de forma a evitar que o utilizador insira dados errados; * Depois do preenchimento dos dados, deverá existir um botão **Gravar;** * A informação deverá se encontrar na base de dados do Fenix, a espera de serem válidados; * Só depois de serem válidados pela SA, esta informação será carregada no Primavera através de um botão **Aprovar.**   **NOTA1**: O campo “bolsa”, será manipulada manualmente no Primavera;  **NOTA2**: O campo “status”, é que vai definir se ocorreu um erro ou não em relação a comunicação Fenix-Primavera. Onde só depois do Primavera confirmar que gravou o estudante, este campo irá retornar um “*ok*”, caso não, permanecerá com “erro”. O Primavera ainda, vai informar ao Fenix a data de sincronização do dia em que for gerada a fatura. |
| nrEstudante (\*) | String(12) | Alfanuméricos |
| nomeCompleto (\*) | String (50) | no numbers |
| morada (\*) | String (50) |  |
| bairro (\*) | String (50) |  |
| nrTelefone (\*) | String (20) | (82) 123 456 7 |
| nrTelefone2 | String (20) | (84) 123 456 7 |
| nuit | String (20) |  |
| bolsa | Boolean |  |
| emailPessoal | String (100) |  |
| emailAlternativo | String (100) |  |
| tipoIngresso (\*) | String (100) |  |
| status (\*) | String (250) | Erro / Sucesso |
| observacao | String (250) |  |

**Legenda:**

* **SA –** Secretaria Académica;
* **(\*) –** Campos obrigatórios

**ANEXO 1 – Situações possíveis de erro (Definido no campo** ***PrimaveraResultStructure.statusCode*)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipo de Problema** | **Código** | **Código do Nível – Descrição** | **Código/Subnível – Descrição** | **Procedimento após o erro** |
| Códigos de Sucesso | **0** | **00** – Sucesso completo | **000** – Gravado com sucesso | **-** |
| **01** – Sucesso com aviso | **010** – Aluno gravado, mas existe outro com nome idêntico | **-** |
| Erro de Marshalling de dados | **1** | **10** – Validação dos campos | **100** - Primavera encontrou um campo com nome inválido | Insere um nome válido sem números e caractéres especiais |
| **101** - Primavera encontrou um campo com data de registo impossivel (ex. data no futuro) | Verificar a data inserida no campo |
| **102 –** Primavera encontrou um campo com nr° de telefone inválido | Insere correctamente o nr° de telefone |
| **103** – Primavera encontrou um campon com Email inválido | Inserir correctamente o endereço de mail |
| **104** – Nr° de caractéres maior que “X” | Verificar a quantidade de caractéres inseridos no campo “Y” |
| Erro de sistema no Primavera | **2** | **20** – Erros com a DB | **200** - Primavera não consegue conectar a Base de Dados | Consultar os técnicos do projecto |
| **201** – Primavera não consegue conectar a Tabela “X” | Consultar os técnicos do projecto |
| **21** – Erros com ficheiros DLL | **210** - Primavera não consegue carregar um DLL | Consultar os técnicos do projecto |
| **22** – Outros erros | **220** - Erro não especificado | Consultar os técnicos do projecto |
| Erro lógico no Primavera | **3** | **30** – Erros de inscrição | **300** - Aluno já existe com o mesmo código | Consutar os técnicos do projecto |
| **301** – Já existe um aluno com o mesmo nr° de BI | Verificar bem os dados, pode ser que esteja a introduzir um aluno existente |
| **302** – O tipo de Ingresso não foi especificado | Especificar o tipo de ingresso |
| **31** – Erros na definição de *itens* | **310** - *item* referenciado, tem PVP 0 |  |
| Erros de rede | **4** | **40 –** Erros comuns | **400 –** *Time Out* | Verificar a conectividade com a rede |

A classe **PrimaveraResultStructure()** será definida como mostra a tabela abaixo:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Classe** | **Inputs** | | | **Métodos** | **Output** | **Descrição** |
| **PrimaveraResultStructure()** | **Nome dos campo/atributos** | **Tipo de Dados/ Tamanho** | **Valores Permitidos** | Um método **toString()** que vai retornar todos o valores dos atributos desta classe, assim como a descrição do problema. | Mensagem de erro/sucesso, de acordo com o tipo de problema e o seu código associado na hashTable. | * Este método vai conter toda a estrutura de erro com as situações posíveis que podem ocorrer no método **inscreverAluno(<*inputs listados*>)**; * A descrição dos erros, o tipo de problema assim como os níveis, podem se encontrar na tabela acima. * Cada linha na hasTable, irá mapear a um código que representa a um tipo de problema (***key0: value000***; ***key0:value010***. Onde a chave irá se referir ao código e o valor ao nível de código.). Por exemplo, Para:   *PrimaveraResultStructure.codigo = 000,* teríamos:  *PrimaveraResultStructure.extraInfo.put (“0”,”000”)*  Para: *PrimaveraResultStructure.codigo = 010*, teríamos: *PrimaveraResultStructure.extraInfo.put (“0”,”010”)* |
| codigo (\*) | Integer (50) |  |
| descricao (\*) | String (50) |  |
| subnivel (\*) | String (50) |  |
| tipoProblema (\*) | String (50) |  |
| extraInfo | Hashtable | Dados de diferentes tipos. |

**Mapeamento/Representação dos códigos de erros numa HashTable**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **KEY** | **0** | **1** | **2** | **3** | **4** |
| **VALUE** | **000** | **100** | **200** | **300** | **400** |
| **010** | **101** | **201** | **301** |  |
|  | **102** | **210** | **302** |  |
|  | **103** | **220** | **310** |  |
|  | **104** |  |  |  |