**ANEXO 1 – Definição de situações possíveis de erro *PrimaveraResultStructure***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 caracteres especiais |
| **101** - Primavera encontrou um campo com data de registo impossível (ex. data no futuro) | Verificar a data inserida no campo |
| **102 –** Primavera encontrou um campo com nr° de telefone inválido | Insere corretamente o nr° de telefone |
| **103**– Primavera encontrou um campo com Email inválido | Inserir corretamente o endereço de mail |
| **104** – Nr° de caracteres maior que “X” | Verificar a quantidade de caracteres 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 | Consultar 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 |

**Definição de Rejeições e Avisos pela validação do Primavera no *PrimaveraResultStructure***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipo de Problema** | **Código** | **Código do Nível – Descrição** | **Código/Subnível – Descrição** | **Procedimento após Rejeição** |
| Rejeições pela validação do Primavera | **5** | **50 –** Rejeições Comuns | **500** – Descrição da rejeição. | - Analisar e agir segundo a descrição da rejeição. |
| Avisos pela Validação do Primavera | **6** | **60 –** Avisos Comuns | **600** – Descrição do Aviso. | - Analisar e agir segundo a descrição do aviso. |

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Classe** | **Atributos** | | | **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. | * Esta Classe vai conter toda a estrutura de erro, avisos, rejeições e resposta dependendo da operação. * Os atributos **estadoConta** e **refBancaria** deverão ser preenchidos quando são chamadas as operações **consultaConta**() e **criarAluno**() respetivamente. * A descrição dos erros, o tipo de problema, rejeições, avisos assim como os níveis, podem se encontrar na tabela acima. * Cada linha no hashTable, irá mapear a um código que representa a um tipo de problema, rejeição ou aviso (***key0: value000***; ***key0:value010***. Onde a chave irá se referir ao código e o valor ao nível de código |
| codigo (\*) | Integer (50) |  |
| descricao (\*) | String (50) |  |
| subnivel (\*) | String (50) |  |
| tipoProblema (\*) | String (50) |  |
| estadoConta | String (90) |  |
| refBancaria | String (90) |  |
| extraInfo | Hashtable | Dados de diferentes tipos. |

**Mapeamento/Representação dos códigos de erros, rejeições e avisos.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **KEY** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **VALUE** | **000** | **100** | **200** | **300** | **400** | **500** | **600** |
| **010** | **101** | **201** | **301** |  |  |  |
|  | **102** | **210** | **302** |  |  |  |
|  | **103** | **220** | **310** |  |  |  |
|  | **104** |  |  |  |  |  |

**NB:** Os valores para o caso das rejeições e avisos, deverão ser associados a descrição,

Ex: **500** – Cliente não pode ser removido pois possui documentos associados.