**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**

Engenharia Informática e de Computadores

**Projecto e Seminário**

**Relatório de Progresso**

**Rapid Application Development**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Membros do Grupo** | | | | | |
| **26657** | **Ricardo Neto** | **31923** | **Nuno Sousa** | **32223** | **Paulo Pires** |
| *26657@alunos.isel.pt* | | *31923@alunos.isel.pt* | | *32223@alunos.isel.pt* | |
| *Tlm. 91.420.02.53* | | *Tlm. 96.205.04.70* | | *Tlm. 92.757.77.35* | |

|  |  |
| --- | --- |
| **Orientador de Projecto** | |
| **Eng. Fernando Miguel Carvalho** | **mcarvalho@cc.isel.ipl.pt** |

**Índice**

[1. Ponto de Situação 4](#_Toc260675764)

[**1.1.** **Actividades Desenvolvidas à Presente Data** 4](#_Toc260675765)

[**1.1.1.** **Definição do Dicionário de Dados** 4](#_Toc260675766)

[**1.1.1.1.** **Modelo de Tipos** 5](#_Toc260675767)

[**1.1.1.1.1.** **Tipos por Omissão** 5](#_Toc260675768)

[**1.1.1.1.2.** **Tipos *Custom*** 6](#_Toc260675769)

[**1.1.1.2.** **Environments** 6](#_Toc260675770)

[**1.1.1.2.1.** **Data Environment** 6](#_Toc260675771)

[**1.1.1.2.2.** **Mail Server Environment** 7](#_Toc260675772)

[**1.2.** **Trabalhos em Curso** 8](#_Toc260675773)

[**1.2.1.** **Definição do Dicionário de Dados** 8](#_Toc260675774)

[**1.2.1.1.** **Entidade** 8](#_Toc260675775)

[**1.2.1.2.** **Processo de Negócio** 8](#_Toc260675776)

[**1.2.1.3.** **Permissões** 8](#_Toc260675777)

**Índice de Tabelas**

[Tabela 1 *- Elementos Base* 4](#_Toc260646853)

[Tabela 2 *- Especificação de Tipos por Omissão* 5](#_Toc260646854)

[Tabela 3 *- Significado de atributos de tipos por natureza do tipo* 5](#_Toc260646855)

[Tabela 4 *- Atributos relativos a um Data Environment do tipo 'SQL Server'* 7](#_Toc260646856)

[Tabela 5 *- Atributos relativos a um MailServer Environment* 7](#_Toc260646857)

1. **Ponto de Situação**
   1. **Actividades Desenvolvidas à Presente Data**
      1. **Definição do Dicionário de Dados**

O objectivo desta actividade é o de criar a estrutura responsável por conter a descrição de todos os constituíntes da solução.

O formato escolhido para codificar esta descrição é o XML, validado pelo respectivo XSD.

**De toda a informação passível de ser utilizada, destacam-se os seguintes elementos:**

|  |  |
| --- | --- |
| ****Tipo**** | ****Descrição**** |
| ****Tipos**** | **Gestão de tipos, tendo disponíveis por omissão os tipos primitivos básicos.** |
| ****Environments**** | **Definição de servidores de base de dados, aplicacionais, e-mail, ftp, LDAP,...** |
| ****Entidades**** | **Criação de entidades passando pela definição de atributos e seus domínios à relação entre entidades e sua cardinalidade e persistência.** |
| ****Processos**** | **Definição de processos com determinação de entidades envolvidas e regras a aplicar.** |
| ****Permissões**** | **Criação de matriz de permissões baseada em RBAC com vista à sua aplicação aos processos definidos.** |

Tabela 1 *- Elementos Base*

* + - 1. **Modelo de Tipos**
         1. **Tipos por Omissão**

Com o objectivo de ser disponibilizado um modelo de tipos por omissão com correspondência a tipos utilizados nas linguagens de programação (e.g. C#), bem como, em motores de bases de dados relacionais (e.g. SQL), optou-se por utilizar parte do modelo de tipos oferecido pela maioria destes últimos, no caso concreto do Microsoft SQL Server.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tipo | Eq. C# | Eq. SQL | max\_length | Precision | scale |
| guid | Guid | uniqueidentifier | 16 bytes | 0 | 0 |
| datetime | datetime | datetime | 8 bytes | 23 digits | 3 |
| int | int | int | 4 bytes | 10 digits | 0 |
| long | long | bigint | 8 bytes | 19 digits | 0 |
| double | double | real | 4 bytes | 24 digits | 0 |
| float | float | float | 8 bytes | 53 digits | 0 |
| decimal | decimal | decimal | 17 bytes | 38 digits | 38 |
| boolean | boolean | bit | 1 bytes | 1 digits | 0 |
| string(n) | string | varchar | 8000 chars | 0 | 0 |
| binary | byte[] | binary | 8000 bytes | 0 | 0 |
| char | char | char | 8000 chars | 0 | 0 |

Tabela 2 *- Especificação de Tipos por Omissão*

Desta forma, uniformiza-se a caracterização de um tipo disponibilizado pela plataforma, definindo para cada um os parâmetros *max\_length*, *precision*, e *scale*.

Dependendo da natureza do tipo (numérico, caracter, decimal), o valor de cada um dos parâmetros referidos assume significados diferentes:

|  |  |
| --- | --- |
| Parâmetro | Significado |
| *Precision* | Número de dígitos num número |
| *Length* | Para tipos numéricos e binários representa o número de bytes ocupados.  Para tipos caracter representa o número de caracteres |
| *Scale* | Número de dígitos à direita do ponto decimal |

Tabela 3 *- Significado de atributos de tipos por natureza do tipo*

No que respeita à respresentação dos tipos disponibilizados pela plataforma, optou-se por criar um elemento que irá conter todos os tipos base a serem disponibilizados. Cada tipo tem assim os atributos acima referenciados com os valores mostrados.

<platformTypes>

…

<platformType name="datetime" maxLength="8" precision="23" scale="3"/>

<platformType name="int" maxLength="4" precision="10" scale="0"/>

…

</platformTypes>

* + - * 1. **Tipos *Custom***

Os tipos *custom* terão um nome único e terão que extender um tipo da plataforma. A redefinição dos atributos do seu tipo base é uma operação optativa.

No que respeita à sua representação no dicionário de dados, optou-se novamente por criar um elemento que será o contentor (customTypes) de elementos representativos de tipos *custom* (customType).

<customTypes>

<customType name="nif" type="int">

<minValue value="100000000"/>

<maxValue value="999999999"/>

</customType>

</customTypes>

A utilização de um tipo *custom* permite que sejam aplicadas validações ao mesmo durante a execução da aplicação.

As validações possíveis foram pensadas na forma de elementos internos, neste caso *minValue* e *maxValue*. Desta forma, a inclusão de novos critérios de avaliação apenas fica sujeita à restrição deste novo critério ter que ser um elemento interno de *customType*.

Através do *schema* é garantida a unicidade do atributo *name* do elemento *customType*, bem como, a integridade referencial do atributo *type* do elemento *customType*

* + - 1. **Environments**

Esta divisão do dicionário de dados será a que irá guardar informações acerca de todas as entidades externas à solução, com as quais a mesma irá ter necessidade de interagir.

* + - * 1. **Data Environment**

As fontes de dados, como parte integrante de qualquer solução, terão que ser referenciadas, não só para a obtenção/persistência de informação, mas também para permitir que o utilizador opte pela construção automática do seu modelo de dados.

Cada elemento presente nesta divisão deverá indicar qual o tipo de servidor de dados que representa (e.g. *...type=”MySQL”...*). Desta forma, a infra-estrutura poderá solicitar a construção de um objecto que saiba interpretar o elemento XML referente ao tipo especificado.

Após verificação das propriedades de uma ligação, determinou-se quais os atributos necessários à definição de uma origem de dados do tipo SQL Server.

|  |  |  |
| --- | --- | --- |
| Atributo | Observações | Obrigatório |
| Name | Nome único descritivo | X |
| Type | Constante (‘SQLSERVER’) | X |
| Server | Nome ou IP do servidor | X |
| Instance | Nome da instância | X |
| Catálogo | Nome da base de dados | X |
| Port | Porto no servidor |  |
| Username | Nome de utilizador | X |
| Password | Password | X |
| ConnPool Min | Número mínimo de ligações à BD |  |
| ConnPool Max | Número máximo de ligações à BD |  |
| Security | Informação relativa à segurança na ligação com o servidor (SSPI – utilizador de contexto) |  |
| Timeout | Timeout associado à ligação |  |

Tabela 4 *- Atributos relativos a um Data Environment do tipo 'SQL Server'*

Através do *schema* é garantida a unicidade do atributo *name* do elemento *dataEnvironment*.

* + - * 1. **Mail Server Environment**

A funcionalidade de envio de e-mails é algo que pode ser exigido na execução da aplicação (e.g. output de processo de negócio), como tal, permite-se que sejam configurados servidores de e-mail no sentido de permitir aos programadores a utilização do serviço sem haver preocupação a definição dos seus parâmetros.

|  |  |  |
| --- | --- | --- |
| Atributo | Observações | Obrigatório |
| Name | Nome único descritivo | X |
| Server | Nome do servidor SMTP | X |
| Port | Porto no servidor SMTP | X |
| Username | Nome de utilizador (e.g. mailbox) | X |
| Password | Password | X |
| Timeout | Timeout para ligação ao servidor |  |
| Autenticação | Tipo de autenticação (basic, NTLM) | X |
| SSL/TLS | Flag indicadora de ligação segura | X |

Tabela 5 *- Atributos relativos a um MailServer Environment*

* 1. **Trabalhos em Curso**
     1. **Definição do Dicionário de Dados**

No que respeita a esta tarefa estão ainda por definir os elementos *Entity*, *Permission* e *BusinessProcess*, todavia, apresenta-se uma definição sucinta dos mesmos.

* + - 1. **Entidade**

Um elemento *entity* é uma entidade pertencente ao dominio do problema (e.g. Cliente). Este elemento deverá conter informação acerca dos atributos da entidade e o tipo de persistência aplicada, bem como, *data environment* designado para o efeito.

No que respeita aos atributos de um *proxy*, deve ser indicado o nome do atributo e o tipo (referenciando obrigatoriamente um tipo do modelo de tipos).

Caso o atributo pertença a uma relação com outra entidade, esta relação terá também que ser definida indicando obrigatoriamente a cardinalidade e as entidades participantes na mesma.

* + - 1. **Processo de Negócio**

Um processo de negócio irá envolver a definição de quais os inputs do processo. Uma vez que, conforme indicado, a granularidade das permissões será ao nível do processo, este deverá indicar se são exigidas permissões ou não.

* + - 1. **Permissões**

O esquema de permissões a utilizar irá seguir os principios definidos no modelo RBAC (*Role Based Access Control*), sendo que, será necessário definir a origem da informação de *user assignments* e *permission assignments*.

A granularidade do esquema de permissões será ao nível do processo de negócio, fixando-se como objectivo a geração de código que represente uma exigência declarativa de permissões.