**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**

Engenharia Informática e de Computadores



**Projecto e Seminário**

**Proposta de Projecto**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Membros do Grupo** | | | | | |
| **26657** | **Ricardo Neto** | **31923** | **Nuno Sousa** | **32223** | **Paulo Pires** |

**Introdução**

Com o aparecimento de software cada vez mais barato, torna-se imperativo para as software houses reduzirem os custos de desenvolvimento de produtos, no sentido de conseguirem operar num mercado global cada vez mais competitivo.

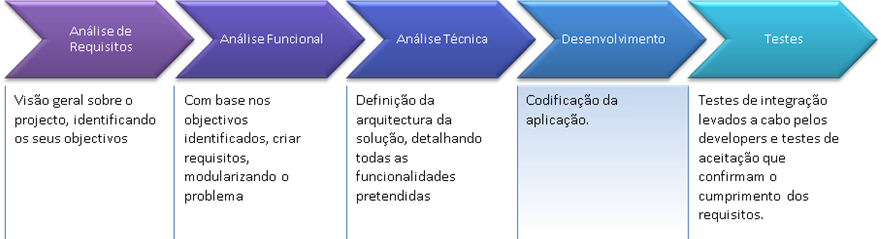
O meio para chegar a esse fim divide-se em diminuir o nível de remuneração dos seus recursos humanos ou diminuir o ciclo de desenvolvimento do produto.

Com o presente projecto pretende-se criar uma solução que permita o desenvolvimento de aplicações de forma mais rápida, diminuindo assim todo o ciclo de desenvolvimento incluído.

**Análise**

**Abordagem**

Considerando o ciclo de desenvolvimento de uma aplicações destacam-se as seguintes fases:



**No caminho para a fase de desenvolvimento, são produzidos outputs suficientes para se conseguir descrever um problema com algum detalhe.**

**Pretende-se que, ao estruturar essa informação num formato específico, seja possível codificar estruturas sobre as quais a equipa de desenvolvimento iniciará o seu trabalho.**

**Arquitectura**

**Ficheiro descritor do problema**

**A presente solução visa criar regras para a construção de um ficheiro com meta-informação, que incluirá todos os outputs, descrevendo assim os vários elementos a serem codificados.**

**De toda informação passível de estar presente neste ficheiro, destacamos os seguintes elementos:**

|  |  |
| --- | --- |
| **Tipo** | **Descrição** |
| **Environments** | **Definição de servidores de base de dados, aplicacionais, e-mail, ftp, LDAP,...** |
| **Tipos** | **Gestão de tipos, tendo disponíveis por omissão os tipos primitivos básicos.** |
| **Proxies** | **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.** |

**Módulo de Coordenação**

Com base no ficheiro descritor, passa-se agora à fase de geração automática de código.

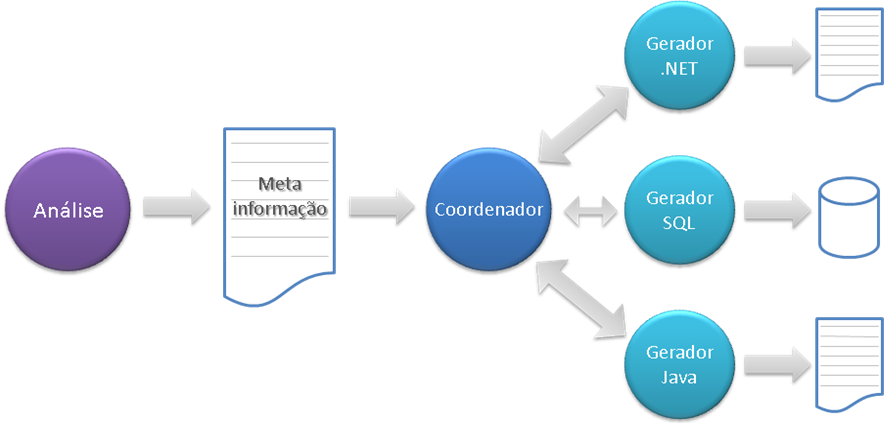
Com vista a permitir que o processo de codificação possa ter várias linguagens alvo, irá ser criado um módulo de coordenação que terá a função de orquestrar todo o processo.

Ao verificar o tipo de geração (e.g. Java, .NET, Ruby, SQL, MySQL...) indicada no ficheiro descritor, o coordenador irá solicitar a criação de uma instância desse gerador, recolher informação do ficheiro, passando-a ao gerador, através de uma interface bem definida.

**Gerador Automático de Código**

A possibilidade de geração de código em várias linguagens será suportada, naturalmente, por contrato, criando uma interface bem definida que definirá o comportamento que um gerador deverá ter.

O gerador terá que disponibilizar um factory que permita ao coordenador obter uma instância para o mesmo.



**Trabalho a desenvolver**

**Obrigatório**

**- Estrutura e definição de ficheiro descritivo**

**- Interface e módulo de coordenação**

**- Interface de gerador**

**- Implementação de gerador .NET e SQL**

**Optativo**

**- Editor de ficheiro descritivo**

**Calendarização**