

# Tema 2 – Associação Portuguesa Investigação Científica Parte 1

João Ferreira – [up201605330@fe.up.pt](mailto:up201605330@fe.up.pt)  
Nuno Lopes – [up201605337@fe.up.pt](mailto:up201605337@fe.up.pt)  
Pedro Costa – [up201605339@fe.up.pt](mailto:up201605339@fe.up.pt)

17 de novembro de 2017

# Índice:

1. Índice.....	2
2. Tema do trabalho.....	3
3. Solução implementada.....	4, 5
4. Diagramas UML.....	6
5. Casos de utilização.....	7
6. Principais dificuldades.....	8
7. Distribuição do trabalho pelos membros do grupo.....	9

## 2. Tema do trabalho:

A Associação Portuguesa de Investigação Científica (APIC) promove e divulga projetos de qualquer área científica e apoia-os financeiramente. Os projetos são propostos por um conjunto de associados.

Cada associado está ligado a uma instituição de ensino e/ou investigação, possui um conjunto de áreas e de subáreas de interesse e pode ainda pesquisar sobre os associados existentes em certa área ou subárea. Cada um paga a sua cota anual.

Os associados com as cotas em dia chamam-se “contributors”. Associados que não têm as cotas em dia, mas com atraso inferior a 5 anos consideram-se “subscribers”.

A APIC possui também uma rede para distribuição de emails. Só os contributors podem enviar emails para o resto dos associados e receber. Já os subscribers apenas podem receber. Por fim, os associados que não pagam as cotas há mais de 5 anos não podem enviar nem receber emails da rede de distribuição. Pode-se medir a atividade de cada associado através do número de emails que envia e recebe da rede.

Por fim, se, pelo menos, 3 associados quiserem promover um evento podem obter apoio financeiro da APIC. O evento terá de ter um local, data, tema e lista de organizadores. Os eventos podem ser conferências, para as quais deve haver um número aproximado de participantes, ou escolas de verão, para as quais há uma lista de formadores. Deve-se ter em conta o número de associados envolvidos no evento para se decidir se há ou não apoio ao evento.

### 3. Solução implementada:

Implementámos a Associação Portuguesa Investigação Científica (APIC) através da classe Associação. A Associação terá um nome e uma cota anual que terá de ser paga por todos os seus Associados e que irá ser adicionada ao fundo inicial. Este fundo é usado para ajudar a suportar os custos dos Eventos promovidos pela Associação. Por fim, a Associação permite aos seu Associados usarem uma Rede de distribuição de Mails. A informação da APIC está guardada num únicos ficheiros de texto.

Tal como já foi referido, a Associação é formada por um conjunto de Associados. Apesar de cada um possuir um nome, decidimos distingui-los pelo seu ID, que é único, sequencial e fornecido por um membro-dado estático da classe Associado, garantindo assim o bom funcionamento do programa. Cada membro desta classe possui também uma instituição onde trabalha, um conjunto de Áreas científicas de interesse e uma certa quantidade de dinheiro que é usado para pagar a cota anual definida pela Associação. Cada Associado tem ainda um número de divulgações que representa quantas vezes este enviou um Mail para a Rede de distribuição. Toda a informação dos Associados está presente num ficheiro.

Há duas opções para o pagamento das cotas anuais da parte dos Associados. Assim, o pagamento pode ser efetuado automaticamente, ou seja, quando passa um ano, todos os Associados com uma quantidade de dinheiro superior à cota pagam esta, ou manualmente, o que significa que um certo Associado só a paga se o utilizador assim o entender. Definimos três classes de excepções para o pagamento das cotas. A classe NotUpToDate é usada para o caso de o utilizador querer pagar um ano tendo dívidas anteriores, a classe NotEnoughMoney usa-se se quiser pagar uma cota que não tem dinheiro para pagar e a AlreadyPaid é empregue se se quiser pagar um ano que já foi pago.

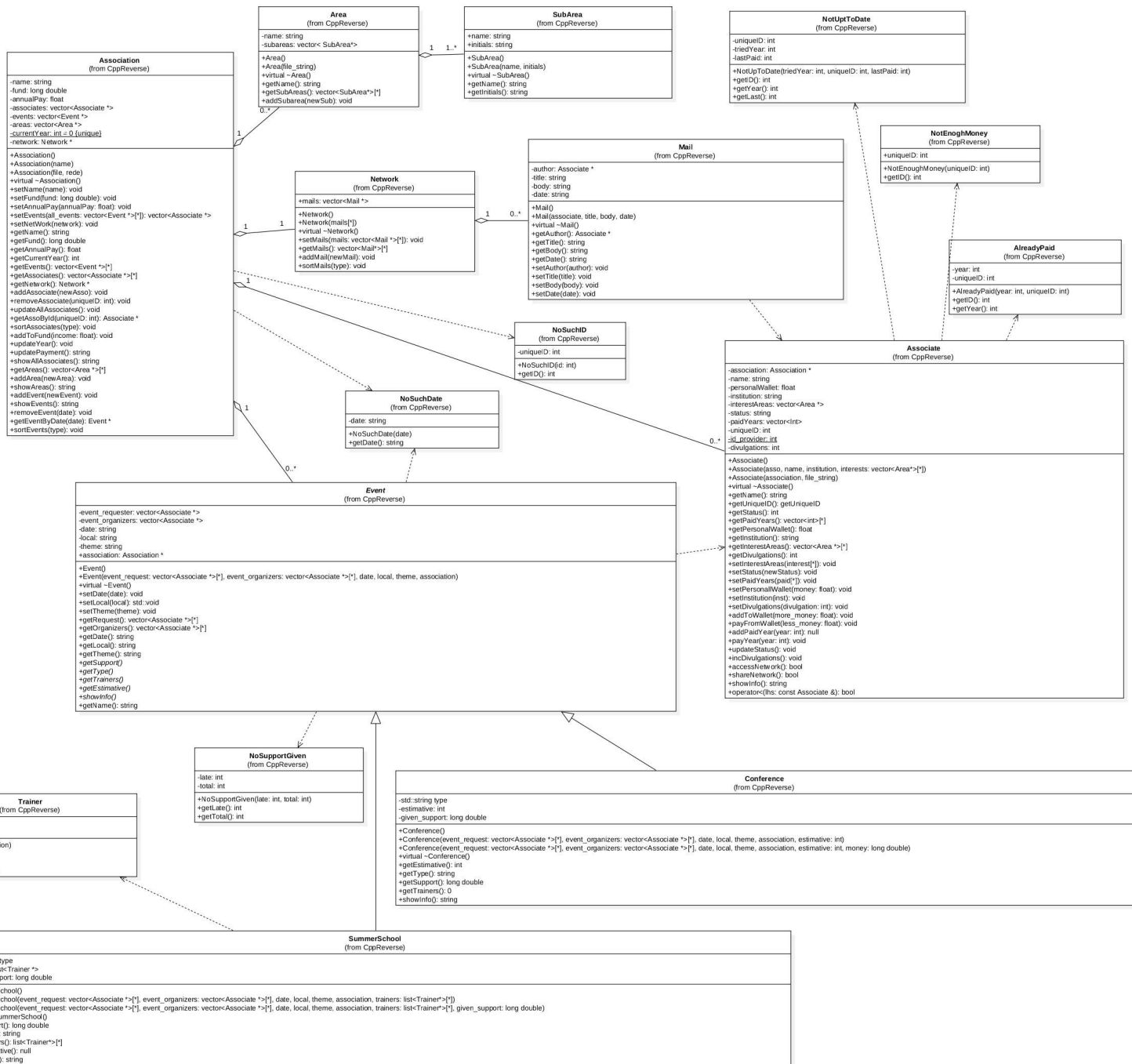
A classe Evento é uma classe base das classes Conferência e Escola de Verão. Assim, todos os Eventos possuem um grupo de, pelo menos, três Associados que propõem a realização deste à Associação e um grupo de Associados que irão ajudar a organizar o Evento. Cada Evento tem também uma data, local e tema associados. A diferença entre as duas classes derivadas está no facto de que a classe Conferência possui um outro parâmetro que é um número estimado de participantes, enquanto que a classe Escola de Verão tem um conjunto de formadores que foram outra classe, classe Formador. Cada um destes possui um nome e uma instituição que representa. Todos os Eventos possuem a sua informação guardada num só ficheiro.

Já as classes Área e SubÁrea representam um conjunto de áreas de carácter científico nas quais os Associados se identificam. Cada Área possui um nome e um grupo de SubÁreas, que por sua vez possuem um nome e uma sigla. As Áreas e SubÁreas estão guardadas num ficheiro.

Como foi dito anteriormente, a APIC fornece uma Rede de distribuição de Mails. A classe Rede de distribuição possui um conjunto de Mails que foram enviados. Já a classe Mail possui como parâmetros um autor, um título, uma data e o corpo de texto. Para a visualização dos Mails, o utilizador terá de, por momentos, tornar-se um Associado, introduzindo o seu ID único e assim visualizar e enviar Mails de acordo com o seu estatuto de “contributor”, “subscriber” ou apenas membro da Associação. Todos os Mails estão guardados num só ficheiro.

Decidimos implementar algoritmos de ordenação parcial ou total com critérios a definir pelo utilizador em todas as classes a que este tem acesso. Isto permite, então, uma visualização amigável e mais intuitiva da informação. No entanto, no momento inicial e final do programa este automaticamente lê e guarda a informação de uma forma pré-definida por nós. Dado que algumas funções possuem membros-dado e função decidimos implementar funções de comparação template, poupando algumas linhas de código e tempo.

#### 4. Diagramas UML:



## 5. Casos de utilização:

- Criar Associados de uma Associação;
- Remover Associados de uma Associação;
- Alterar informações de um Associado, nomeadamente as suas áreas de interesse e a sua instituição;
- Ver informações dos Associados, sendo possível escolher entre apenas visualizar a informação de um Associado, ver de todos, ou uma listagem parcial dos Associados em que os critérios de escolha são o identificador único do Associado e a sua “carteira”;
- Organizar a estrutura de dados dos Associados em função do nome, do identificador único ou do dinheiro disponível na “carteira”;
- Criar um Evento para pedir apoio à Associação;
- Remover o Evento da Associação;
- Alterar informações de um Evento, nomeadamente a sua data, o local, o tema e dependendo do seu tipo de Evento podem ser alterados os Formadores se o Evento for uma Escola de Verão ou alterar o número esperado de participantes se o Evento for uma Conferência;
- Organizar a estrutura de dados dos Eventos em função do nome, local ou tema;
- Pagar as cotas dos Associados individualmente ou automaticamente;
- Visualizar Associados com cotas em atraso;
- Divulgar um Email para a Rede da Associação;
- Visualizar todos os Emails da Rede da Associação;
- Organizar os Emails em função do título ou da data;
- Ler e guardar todas as informações sobre a Associação, os Associados, os Eventos e os Emails se o utilizador desejar.

## 6. Principais dificuldades:

Quanto à estrutura e planeamento das classes não houve grandes dificuldades na sua realização, porém encontramos algumas dificuldades na forma como iríamos gerir o tempo real e como seria efetuado o pagamentos das cotas por parte dos Associados. Mesmo assim conseguimos chegar a uma solução que para nós nos parece ser a mais aceitável.

Além disso, a adição dos comentários, dos métodos get e set de todas as classes e as exceções apesar de serem fáceis de realizar foram bastante trabalhosas e demoradas.

Por último, a listagem parcial pela data ofereceu alguma dificuldade, tendo em conta que estas se encontravam numa string e não no formato ideal para a correta comparação. Apesar disto, até conseguimos implementar dois algoritmos diferentes para a resolução do problema.



## 7. Distribuição do trabalho pelos membros do grupo:

O nosso grupo tentou distribuir o trabalho igualmente pelos três elementos do grupo.

Todos os elementos contribuíram para a realização do projeto, ajudando com ideias para o desenvolver e na aplicação do código. No entanto é de se notar que houve maior empenho por parte do membro Pedro Costa.

De um modo geral a realização deste projeto correu bastante bem, havendo sempre uma boa relação entre os membros, levando assim à conclusão do projeto dentro do prazo estipulado.