

**Gestão de Projeto de Software 2019/2020**

**SmartParking**

**Convenções de código**

**Versão: v0.1**

O código será desenvolvido tendo em conta algumas convenções de código, as principais das quais serão especificadas neste documento, o código do nosso projeto será desenvolvido em java, como tal iremos seguir as boas práticas e convenções genéricas da utilização desta linguagem.

1. **Razões para haver convenções de código**

* 80% do custo de vida útil de uma parte do software vai para manutenção.
* Raramente nenhum software, ao longo da sua vida, é mantido pelo autor original.
* As convenções de código melhoram a legibilidade do software, permitindo aos engenheiros perceber novo código de maneira mais rápida e completa.
* Ao enviar o código fonte como um produto, é necessário que seja bem embalado e limpo com qualquer outro produto.

1. **Organização dos Ficheiros**

* Um ficheiro consiste em secções que devem ser separadas por linhas em branco e comentários opcionais descrevendo cada secção.
* Um ficheiro com mais de 2000 linhas é excessivo e deve ser evitado.
* Ordenação de um ficheiro de código fonte:
  1. Comentários iniciais.
  2. Package e Imports.
  3. Declaração de Classes e Interfaces.

1. **Indentação**

* Evitar linhas compridas com mais de 80 caracteres, maior parte dos IDE já indicam esse limite.
* Caso a expressão não caiba numa única linha, separa-se de acordo com estes princípios:
  1. Terminar após uma vírgula.
  2. Terminar antes de um operador.
  3. Alinhar a nova linha com o ínicio da expressão da linha anterior.

1. **Comentários**

* Os comentários são importantes para comentar sobre o código em questão ou para comentar sobre a sua implementação em específico.
* Os comentários devem ser usados para fornecer uma visão geral do código e fornecer informações adicionais que não está disponível no próprio código.

1. **Declarações**

* Uma declaração por linha é recomendada, pois incentiva os comentários.
* Colocar declarações apenas no início dos blocos.
* Tentar inicializar as variáveis locais onde elas são declaradas.

1. **Convenção de nomenclatura**

* **Packages**: O nome do pacote deve ser único, deve ser sempre escrito em letras minúsculas. Componentes subsequentes do nome do pacote variam de acordo com uma organização próprias a cada um desses componentes.
* **Classes**: Os nomes de classe devem ser substantivos, em maiúsculas e minúsculas com a primeira letra de cada palavra em maiúscula. O nome das classes deve ser sempre simples e descritivo.
* **Interfaces**: Nomes de interfaces devem ser usadas com as primeiras letras em maiúsculas como nome de classes.
* **Métodos**: Métodos devem ser verbos, com a primeira letra em minúscula e as seguintes primeiras letras de cada palavra em maiúsculas.
* **Variáveis**: Os nomes de variáveis devem ser curtos, mas significativos. A escolha de um nome nome devem ser concebidos para indicar ao observador casual a intenção da sua utilização. Os nomes comuns para variáveis temporárias são i, j, k, m, n para inteiros, e, c, d, e para carateres.

# Bibliografia

Java Code Conventions by Sun Microsystems - [Documento](https://www.oracle.com/technetwork/java/codeconventions-150003.pdf)