**Características comuns de designs  
(Gabriel Garcia e Diogo Dias)**

Cada metodologia nova de design de software introduz novas heurísticas e notações, onde cada uma contem:

1. Um mecanismo que traduz o modelo de requisitos para uma representação

gráfica. Basicamente com cada nova metodologia de design de software, há sempre mudanças em diversos aspetos em relação a procedimentos feitos anteriormente, com a finalidade de melhorar esta transição de modelo de requisitos para algo mais visual, com intuito de ajudar a entender melhor o design e funcionamento da aplicação.

1. Uma notação para representar componentes funcionais e a suas interfaces. Geralmente esta notação para representar componentes funcionais, utiliza-se um diagrama de

componentes em UML, que tem como objetivo demonstrar, especificar, e documentar como um sistema interage entre si. De modo a certificar que as funções requeridas estão implementadas.

1. Heurísticas para refinamento e partição. Uma heurística é um atalho mental que permite que as pessoas resolvam problemas e façam julgamentos de forma rápida e eficiente.  
   Temos de ter algum tipo de heurística para o refinamento de ideias e partições, ou seja, por exemplo em relação aos diagramas que foram desenhados, verificar quais são os que realmente importam e fazer uma divisão.
2. Diretrizes para avaliação de qualidade. Uma diretriz é uma declaração pela qual determinar um curso de ação. Uma diretriz tem como objetivo otimizar processos específicos de acordo com uma rotina definida ou uma boa prática.  
   A avaliação da qualidade, é a coleta e análise de dados por meio da qual o grau de conformidade com padrões e critérios predeterminados é exemplificado. Se a qualidade, por meio desse processo, for considerada insatisfatória, são feitas tentativas para descobrir o motivo disso.   
   Tendo esta informação, sabemos então que estas diretrizes existem para um designer de software ter padrões e técnicas por onde se guiar, e não apenas desenhar o software conforme o seu gosto.

**Conceitos de design  
(João Mendes)**

**Abstração**: Esconder os detalhes para reduzir a complexidade e aumentar a eficiência e qualidade.

**Arquitetura:** Técnica de estruturar algo, neste caso em engenharia de software pode ser estruturar como é que componentes interagem entre si.

**Padrões de Design**: São soluções genéricas para problemas que ocorrem com frequência. Genéricas porque uma só solução pode ser adaptada para diferentes tipos de situações.

**Separação de preocupações:** Se temos um problema grande, podemos dividi-lo em problemas mais pequenos, focando-nos num só problema pequeno de cada vez até atingirmos o nosso objetivo.

**Modularidade:** Dividir o sistema em vários módulos para reduzir a complexidade do mesmo. Por exemplo: Num sistema de alugar livros online, teríamos um módulo o catálogo de livros, o perfil do utilizador, o processo de alugar, etc.

**Modularidade e Custo de Software  
(Sérgio Simões)**

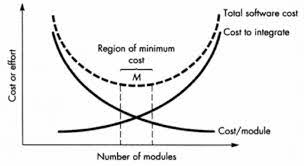
Modularity and Software Cost ou Modularidade e Custo de Software é a relação entre o custo e o número de módulos de um sistema, daí, podemos concluir o custo total do software.

Para explicar este conceito, podemos observar o seguinte gráfico com 3 linhas representadas:

* Custo total do software
* Custo de integração/comunicação entre os módulos
* Custo por módulo

**Podemos observar que:**

* Um baixo número de módulos origina um custo elevado por módulo devido ao facto de cada módulo ser mais complexo, o custo de integração acaba por ser mais reduzido, pois há uma menor preocupação em relação à comunicação entre os módulos, no entanto, o custo total do software é elevado.
* Um elevado número de módulos origina um custo baixo por módulo, devido à baixa complexidade dos mesmos, no entanto o custo total do software é elevado devido ao custo de integração que é mais elevado, pois há uma maior preocupação com a comunicação entre os módulos.
* A região de custo mínimo corresponde à região de equilíbrio entre o custo por módulo e o custo de integração dos mesmos que resulta num custo total do software mais reduzido.



Trabalho realizado por:

Diogo Dias Nº 202001673  
Gabriel Garcia Nº 202002361  
João Mendes Nº 201900459  
Sérgio Simões Nº 201801643