**Relatório de Projecto**

**Programação Aplicada**

**Avaliação Periódica**

**Autor(es):**

Mariana Magalhães a2022147454

Dinis Myroshnyk

**Data:** Abril 2024

Resumo

Este relatório é relativo a um Projeto da cadeira de programação aplicada em que tem como objetivo criar uma aplicação Java que utiliza programação orientada a objetos e manipulação de bases de dados relacionais para auxiliar uma editora no processo de revisão de livros. A aplicação deve gerenciar informações de revisão, incluindo acesso e usuários, manipulação de arquivos, validação de dados, notificações e interação com o usuário em modo texto. O projeto procura aplicar habilidades em programação orientada a objetos, manipulação de bases de dados relacionais com JDBC e uso de arquivos para armazenamento persistente de dados.

*O resumo faz uma breve descrição dos principais pontos do trabalho: (1) o objectivo do estudo ou o seu contexto, (2) a metodologia e métodos do estudo, (3) principais resultados e (4) conclusões relevantes. O resumo deve apresentar informação directa e concisa, preferencialmente em menos de 200 palavras. Esta deve ser a última secção do documento a ser escrita.*

**Palavras-chave**

Introduzir palavras-chave ou frases em ordem alfabética, separadas por vírgula.

**Índice**

[Resumo v](#_Toc65518655)

[Lista de Figuras xi](#_Toc65518656)

[Lista de Tabelas xiii](#_Toc65518657)

[Lista de Acrónimos xv](#_Toc65518658)

[1. Introdução 1](#_Toc65518659)

[1.1. Formatação do Documento 2](#_Toc65518660)

[2. Estado da Arte 5](#_Toc65518661)

[3. Objectivos e Metodologias 7](#_Toc65518662)

[3.1. Ferramentas e Tecnologias 7](#_Toc65518663)

[3.2. Planeamento 7](#_Toc65518664)

[4. Trabalho Desenvolvido 9](#_Toc65518665)

[4.1. Requisitos Implementados 9](#_Toc65518666)

[4.2. Classes e *Packages* 9](#_Toc65518667)

[4.3. Algoritmos 9](#_Toc65518668)

[4.4. Estruturas de Dados 9](#_Toc65518669)

[4.5. Armazenamento de Dados 10](#_Toc65518670)

[4.6. Procedimentos de Teste 10](#_Toc65518671)

[5. Conclusões 11](#_Toc65518672)

[5.1. Forças 11](#_Toc65518673)

[5.2. Limitações 11](#_Toc65518674)

[5.3. Trabalho Futuro 11](#_Toc65518675)

[6. Referências 13](#_Toc65518676)

[6.1. Lista de Referências 14](#_Toc65518677)

[6.2. Como Integrar Referências 15](#_Toc65518678)

[7. Anexos 17](#_Toc65518679)

# Lista de Figuras

[Figura 1‑1 *– Imagem de gráfico em Matlab (adaptado/reproduzido de* (Mathworks, 2020)*).* 3](#_Toc62047348)

# Lista de Tabelas

[Tabela 1‑1 *– Características dos planetas (adaptado/reproduzido de* (Jet Propulsion Laboratory, 2020)*).* 3](#_Toc62047341)

Lista de Acrónimos

**ER Modelo Entidade-Relacionamento**

**BD Bases de Dados**

**DBMS Database Management System**

**ESTGOH Escola Superior de Tecnologia e Gestão**

**IPC Instituto Politécnico de Coimbra**

**SGBD Sistema Gestor de Bases de Dados**

**SQL Structured Query Language**

# Introdução

*Neste projecto, pretende-se a implementação de uma aplicação completa em linguagem Java, sem componente gráfica para interacção com o utilizador, recorrendo ao paradigma de Programação Orientada a Objectos e com suporte para bases de dados relacionais.*

*A aplicação tem como objectivo auxiliar uma editora no processo de revisão de obras (livros). O sistema deve permitir*

*armazenar toda a informação inerente ao processo e possibilitar a consulta e pesquisa da mesma.*

## Formatação do Documento

***Nota****: Esta secção não deve constar no documento final e serve para explicar como um relatório deve ser estruturado e formatado. Este modelo (template) pretende apresentar as secções base que genericamente devem surgir em qualquer relatório ou tese. A estrutura base do documento deve ser adaptada ao objectivo do projecto em desenvolvimento, podendo acrescentar novos capítulos (de qualquer nível), embora seja recomendável manter todos os capítulo e secções presentes no modelo, com excepção da presente subsecção (1.1* Formatação do Documento*), que deve ser removido.*

### Estilos de Secções, Subsecções e Texto

*Cada capítulo deve ter um título formatado no estilo “Título 1”/”Heading 1” (nível 1). As subsecções devem ser formatadas com o estilo “Título 2”/ ”Heading 2” (nível 2). Se uma seção de nível inferior for necessária, o estilo “Título 3” / ”Heading 3” (nível 3)também pode ser usado (como exemplifica esta subsecção de nível 3). O texto deve seguir o estilo “Normal”.*

*É recomendável que os títulos de capítulos e subcapítulos sejam capitalizados (tenham a primeira letra das palavras em maiúscula), excepto os artigos de ligação (e.g. o, a, de, da). Esta abordagem designa-se por ‘lead-in’, auxiliando o leitor na estrutura do documento. No entanto, alguns livros de estilo, como o Chicago Manual of Style, não obrigam esse formato. O importante é manter a consistência ao longo do documento.*

*De notar que nas “páginas pares” (even pages) o título e a numeração (no cabeçalho e rodapé) mudam de localização para permitir a impressão frente e verso. Os capítulos (primeiro nível) devem iniciar numa página ímpar (odd page), ideal para situações de impressão do documento. Caso o capítulo não esteja numerado correctamente, coloque o cursor no início do título do capítulo e seleccione a opção “Quebras” > “Página Ímpar” (Odd Page) na aba “Layout”.*

*Usar um estilo consistente permite uma leitura fácil, simplifica a navegação pelo documento, facilita a alteração de formatos em diferentes localizações do texto e mostra um trabalho profissional e organizado.*

### Tabelas e Figuras

*Tabelas e figuras (como a Figura 1‑1) devem ser identificadas com a palavra "Tabela", "Figura" ou outro descritor apropriado e incluir um título e / ou legenda. Deve ser usado um formato consistente para títulos e legendas de tabelas, figuras e ilustrações ao longo do documento. Certifique-se de usar sempre um estilo “Legenda”/”Caption”. Este procedimento permite que o documento gere automaticamente uma lista de figuras e tabelas e mantenha as referências do texto às figuras ou tabelas actualizadas.*

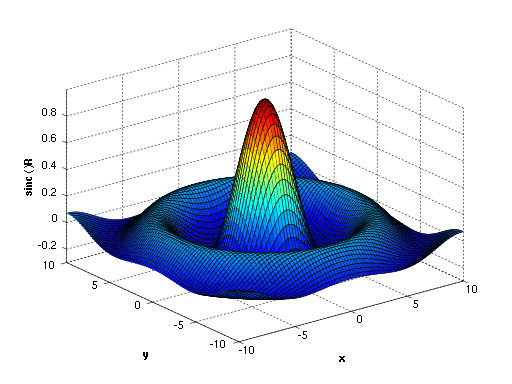
**

Figura 1‑1 *– Imagem de gráfico em Matlab (adaptado/reproduzido de* (Mathworks, 2020)*).*

*Tabelas (como a* Tabela 1‑1*), figuras, ilustrações, equações e outros elementos devem ser numerados consecutivamente na ordem de aparecimento no documento. Recomenda-se que esses elementos sejam numerados utilizando o número do capítulo como prefixo (e.g. Figura 5-2 significa a segunda figura do capítulo 5). Eles devem estar localizados no capítulo imediatamente seguinte à primeira referência ao elemento. Sempre que uma tabela ou figura sejam colocados num documento, é obrigatório que sejam referenciados e descritos no texto do documento, como exemplificado no início desta secção.*

Tabela 1‑1 *– Características dos planetas (adaptado/reproduzido de* (Jet Propulsion Laboratory, 2020)*).*

|  |  |  |
| --- | --- | --- |
| ***Name of Planet*** | ***Average Distance from [Sun](http://www.bobthealien.co.uk/sun.htm)*** | ***Diameter*** |
| [*Mercury*](http://www.bobthealien.co.uk/mercury.htm) | *57,900,000 km* | *4,878 km* |
| [*Venus*](http://www.bobthealien.co.uk/venus.htm) | *108,160,000 km* | *12,104 km* |
| [*Earth*](http://www.bobthealien.co.uk/earth.htm) | *149,600,000 km* | *12,756 km* |
| [*Mars*](http://www.bobthealien.co.uk/mars.htm) | *227,936,640 km* | *6,794 km* |
| [*Jupiter*](http://www.bobthealien.co.uk/jupiter.htm) | *778,369,000 km* | *142,984 km* |
| [*Saturn*](http://www.bobthealien.co.uk/saturn.htm) | *1,427,034,000 km* | *120,536 km* |
| [*Uranus*](http://www.bobthealien.co.uk/uranus.htm) | *2,870,658,186 km* | *51,118 km* |
| [*Neptune*](http://www.bobthealien.co.uk/neptune.htm) | *4,496,976,000 km* | *49,532 km* |

*A referência no texto a figuras e tabelas deve ser feita usando o recurso da ferramenta “****Referências cruzadas****”/”****Cross-reference****” do editor de texto, não manualmente. No caso do editor MS Word, a ferramenta encontra-se disponível na aba “Referências”, secção “Legendas”/”Captions”.*

### Estrutura Global

*É comum um relatório possuir um conjunto de secções base, como exemplificado neste documento: Resumo; Introdução; Estado da Arte; Objectivos e Metodologia; Trabalho Desenvolvido/Resultados; Conclusões e Referências. As conclusões devem apresentar três subsecções Forças; Limitações e Trabalho Futuro. A secção de Trabalho Desenvolvido pode ser dividida em várias secções de nível, nomeadamente para discutir Dados; Implementações/Algoritmos; e Resultados.*

*O documento deve possuir uma capa, um índice global, e índices de figuras e tabelas, caso existam no documento. O índice deve apresentar pelo menos os dois primeiros níveis de secções. Por questão de legibilidade, os níveis inferiores (3 ou superior) podem ser descartados do índice.*

*Na escrita de texto deve ser adoptada uma língua (e.g. português ou inglês) e um acordo ortográfico, mantendo-se consistente ao longo do documento. Caso o documento seja escrito na língua portuguesa, devem evitar-se anglicanismos, sempre que existam palavras a língua portuguesa equivalentes. Excepção feita a termos técnicos próprios da área em estudo.*

# Estado da Arte

*[Quando aplicável]*

*Todos os problemas têm um histórico de tentativas de solução por outros autores. É importante estar ciente de abordagens alternativas para o problema que estamos a tentar resolver. Desta forma devemos compilar os principais estudos que tratam de temas semelhantes, o que se denomina “Estado da Arte” (ou State of the Art).*

*Cada trabalho existente deve ser descrito resumidamente, apontando suas principais características e resultados. Pontos fortes e fracos também devem ser enunciados. Por fim, cada trabalho deve ser comparado com nossa abordagem, mostrando as principais diferenças e a razão pela qual nossa abordagem é necessária e relevante.*

*Após a leitura do estado da arte, o leitor deve estrar convencido de que actualmente não existe uma solução adequada para o problema apresentado na introdução, sendo a abordagem proposta neste trabalho a mais adequada e necessária.*

# Objectivos e Metodologias

O objetivo deste projeto é desenvolver uma aplicação em linguagem Java, utilizando programação orientada a objetos e manipulação de bases de dados relacionais, para auxiliar uma editora no processo de revisão de obras (livros). A aplicação deve permitir o armazenamento e consulta de informações relacionadas ao processo de revisão, incluindo a gestão de acesso e utilizadores, manipulação de ficheiros, validação de dados, notificações, interação com o utilizador em modo texto, entre outros. O projeto visa aplicar conhecimentos e competências relacionados com programação orientada a objetos, manipulação de bases de dados relacionais através de JDBC, uso de ficheiros para armazenamento de dados de forma persistente, e desenvolvimento de interfaces para interação com o utilizador. A entrega do projeto inclui o código da aplicação, executável, descrição das classes, métodos e atributos através de um JavaDoc e um relatório final documentando a aplicação.

## Ferramentas e Tecnologias

PowerDesigner

Visual Studio Code (extensões: SQLTools, SQLTools MySQL/MariaDB/TiDB, Extension Pack for Java)

Beekeeper-studio

## Planeamento

*Esta subsecção deve descrever como foi planeado e distribuído no tempo as tarefas do projecto. Cada tarefa deve ser quantificada, indicando o tempo que tomou (em dias ou horas).*

*Em trabalhos de grupo deve ficar claro que tarefas foram desenvolvidas por cada elemento do grupo.*

*Sempre que possível, é recomendável disponibilizar um diagrama de Gantt.*

# Trabalho Desenvolvido

*Este capítulo representa a secção principal do trabalho, descrevendo o que realmente foi desenvolvido e respectivos resultados. Cada etapa deve ser detalhada e os resultados discutidos e apresentados de forma clara.*

*Deve mencionar com brevidade os principais requisitos do projecto. Uma lista mais detalhada dos requisitos implementados também deve ser fornecida como anexo. Os requisitos podem ser especificados de diferentes formas (e.g. casos de uso, user stories, descrição textual).*

*Esta secção pode ser dividida em várias secções de nível 1 caso se justifique, nomeadamente para discutir Dados; Implementações/Algoritmos; e Resultados.*

## Requisitos Implementados

*Uma lista dos requisitos implementados deve ser apresentada.*

*É aconselhável fornecer uma tabela com todos os requisitos e status correspondentes, indicando quais foram implementados.*

## Classes e *Packages*

*Apresentar e descrever todas as packages e classes do projecto. Os principais métodos de cada classe (e que não sejam óbvios) devem ser igualmente explicados.*

*Recomenda-se que sejam disponibilizados* ***diagramas de classes*** *(ou outros diagramas UML, sempre que se justifique) para melhor compreensão da estrutura do projecto.*

## Algoritmos

*Descrever com brevidade os principais algoritmos, essenciais para o funcionamento do projecto.*

## Estruturas de Dados

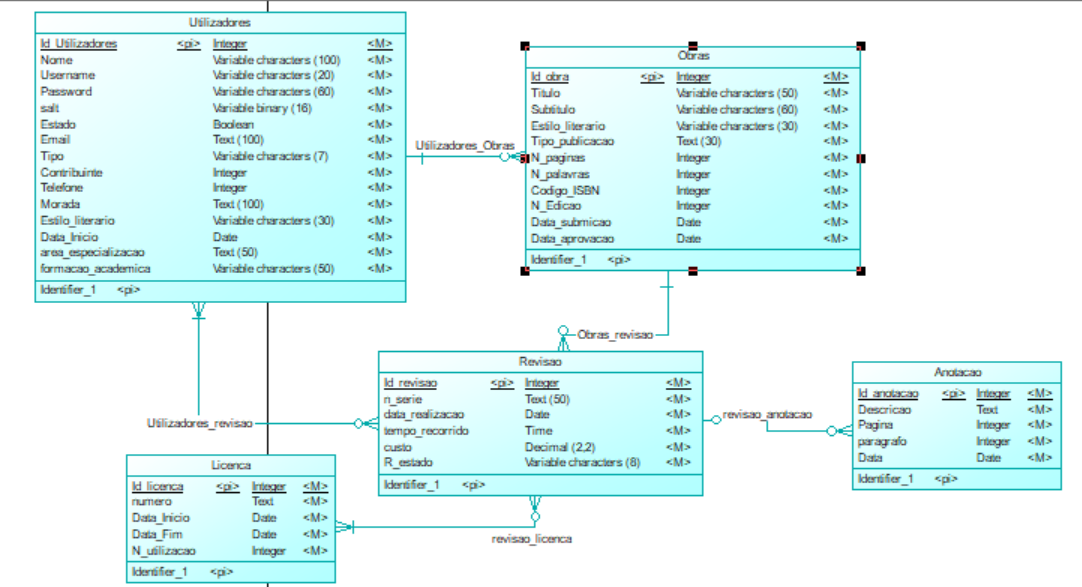
*Descrever com brevidade as estruturas de dados usadas para manipular dados no projecto (e.g. Vectors, ArrayLists, LinkedLists, Hashtables, Maps, Binary Trees), justificando as opções.*

## Armazenamento de Dados

Os métodos de armazenamento de dados são fundamentais para garantir a persistência dos dados em um projeto. Eles determinam como os dados são armazenados, recuperados e gerenciados. Neste caso usamos um Banco de Dados Relacional que utiliza uma estrutura de tabelas para armazenar dados. Cada tabela tem linhas e colunas, e os dados são armazenados em linhas. As relações entre os dados são estabelecidas através de chaves estrangeiras.

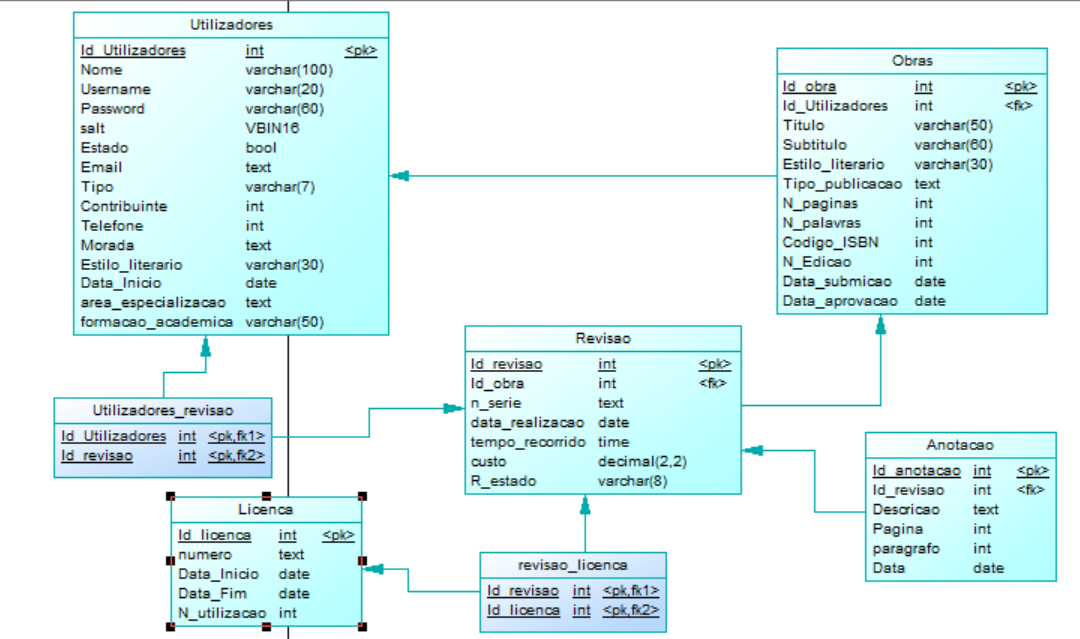
### Diagrama Entidade-Relacionamento

Tendo em conta os requisitos apresentados fomos construindo o seguinte diagrama conceptual. Este diagrama é uma representação visual que ajuda a entender a estrutura e as relações entre os diferentes componentes de um sistema ou projeto. É uma ferramenta valiosa para a comunicação e a compreensão do design de um sistema.



### Modelo Físico

A partir do diagrama conceptual, estruturado na secção anterior, é gerado o diagrama físico, que representa as tabelas e chaves forasteiras que são criadas no Sistema Gestor de Bases de Dados (SGBD), como resultado da aplicação das regras de relacionamentos binários entre entidades



Do modelo físico apresentado atrás destacam-se os seguintes aspectos:

* O modelo utiliza timestamps para registrar datas e horas importantes, como a data de submissão de uma revisão ou a data de criação de uma anotação, o que facilita o acompanhamento do histórico de cada item.
* O modelo usa um campo de estado para controlar o status de cada revisão. Isso permite que o sistema acompanhe o progresso de cada revisão e facilite a identificação de revisões que precisam de atenção.
* O modelo inclui uma tabela separada para gerenciar licenças, para facilitar o controle das licenças disponíveis e o seu uso em revisões.
* O modelo permite que anotações detalhadas sejam feitas em cada revisão. Isso facilita a comunicação entre os revisores e a identificação de problemas específicos.

## Procedimentos de Teste

*Descrever com brevidade os procedimentos de teste adoptados para garantir a funcionalidade e qualidade do código.*

# Conclusões

*O capítulo final deve descrever sumariamente o trabalho desenvolvido, indicando as forças e limitações. Deve-se resumir detalhadamente o objectivo do trabalho e as tarefas concretizadas.*

*É importante descrever as forças do trabalho, ou seja, as metas concretizadas com sucesso, bem como as limitações ou fraquezas do mesmo, indicando o que poderia ou deveria ser desenvolvido de outra forma.*

*Enquanto o documento deve ser técnico, este capítulo permite que sejam realizadas anotações mais pessoais. Assim, enquanto o restante texto deve ser escrito na terceira pessoa, neste capítulo é possível usar a primeira pessoa.*

## Forças

*Descrever as forças do trabalho, frisando o que o distingue das restantes abordagens e demonstrando como foi solucionado o problema inicial.*

## Limitações

*Descrever detalhadamente as limitações do trabalho, ou seja, de que forma não é possível solucionar partes do problema.*

## Trabalho Futuro

*Descrever as melhorias futuras, apresentando uma proposta para os próximos passos a desenvolver para atingir uma solução mais completa ou mais robusta, ou como expandir e diversificar a solução.*

# Referências

*As referências procuram atingir três objectivos: documentar o texto, reconhecer o contributo de outros autores externos e disponibilizar apontadores para fontes externas. Os autores podem e devem citar fontes que permitam suportar ou reforçar os seus argumentos. A construção de conhecimento é baseada em conhecimento já existente, sendo assim necessário indicar que trabalho e fontes são a base desta solução. Devem ser indicados os livros, páginas da Internet, artigos, ou outro qualquer tipo de documento e fonte consultados e que apoiaram o desenvolvimento deste trabalho.*

*Para gerir as referências, é recomendável o uso de um gestor de referências (desde o Microsoft Source Manager, integrado na ferramenta Word – aba “Referencias”, secção “Citações e Bibliografia” –, a aplicativos mais capazes e independentes de plataforma, como o Mendeley, Zotero ou EndNote).*

*Existem vários formatos para a apresentação de referências, sendo os mais comuns a* ***APA*** *(American Psychological Association),* ***ACM*** *(Association for Computing Machinery), e* ***IEEE*** *(Institute of Electrical and Electronics Engineers). Na aba Referencias do MS Word, secção Citações e Bibliografia, é possível definir o formato de referências* *através da opção “Estilo”.*

*Alguns exemplos de referências, usando as normas da ACM:*

*Artigo numa revista:*

ABDELBAR, A.M., AND HEDETNIEMI, S.M. 1998. Approximating MAPs for belief networks in NP-hard and other theorems. Artificial Intelligence 102, 21-38.

*Um livro:*

GINSBERG, M. 1987. Readings in Nonmonotonic Reasoning. Morgan Kaufmann, Los Altos, CA.

*Capítulo de um livro:*

GREINER, R. 1999. Explanation-based learning. In The Encyclopedia of Cognitive Science, R. WILSON AND F. KEIL, Eds. MIT Press, Cambridge, MA, 301-303.

*Artigo publicado numa conferência:*

MAREK, W., AND TRUSZCZYNSKI, M. 1989. Relating autoepistemic and default logics. In Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning, Toronto, Canada, May 1989, H. BRACHMAN AND R. REITER, Eds. Morgan Kaufmann, San Mateo, CA, 276-288.

*Página da Internet:*

POLITÉCNICO DE COIMBRA, 2021. Plataforma de Inquéritos do Sistema Integrado de Garantia de Qualidade do IPC. Disponível em: <https://sigq.ipc.pt/inqueritos/>, acedido em: Janeiro de 2021, última actualização em: 2018.

## Lista de Referências

*Deve assim surgir uma lista das fontes externas utilizadas. As referências devem ser ordenadas alfabeticamente (formatos APA ou ACM) ou pela ordem em que surgem no texto (formato IEEE, neste caso cada referência tem um número se sequência associado):*

Jet Propulsion Laboratory. (2020). *Solar System Sizes and Distances, Reference Guide.* California Institute of Technology.

Mathworks. (2020). *Creating 3-D Plots*. Obtido em Janeiro de 2021, de Matlab Documentation: https://www.mathworks.com/help/matlab/visualize/creating-3-d-plots.html

## Como Integrar Referências

*Enquanto esta secção lista todas as fontes, no texto do documento estas fontes devem ser referenciadas sempre que algum elemento seja baseado ou replicado de uma fonte externa, de acordo com o formato de referência adoptado (e.g. APA: (Author et al., 2020), IEEE: [1] ).*

*Por exemplo, considere-se a seguinte passagem, onde é realizada uma citação explicita (cópia):*

a abstracção de controlo “*implementa uma forma simples de reutilização de código, ao permitir que um bloco de instruções seja escrito independentemente do programa principal*” **(Martins, 2010)**.

*É realizada uma citação explicita, ou seja, parte do texto é integralmente copiado da fonte, sendo formatado em itálico e referenciado.*

*No exemplo seguinte não é realizada uma citação explícita. O texto é escrito pelo autor mas baseado numa fonte externa:*

Em programação orientada a objectos, um objecto caracteriza-se por uma identidade única, um conjunto de atributos que o caracterizam e um conjunto de acções ou métodos **(Martins, 2010)**.

*O autor usou uma fonte, interpretou e apresentou um texto próprio baseado nessa fonte. Não realizou cópia integral, pelo que não é necessário a sua formatação em itálico, mas uma vez que o conhecimento é baseado num outro autor, continua a ser necessário citar a fonte.*

*Na secção final do documento, Referências, deve então surgir a lista de fontes que suportaram o texto. No exemplo seguinte é usado o formato APA:*

Referências

Martins, F., 2010. “Java 5 e Programação por Objectos”, Editora Lidel Edições Técnicas.

Oracle, 2016. “Java Collections”, disponível em <https://docs.oracle.com/javase/tutorial/collections/index.html>, acedido em: Fevereiro 2016, última actualização: 2015.

# Anexos

*Elementos técnicos devem ser colocados como anexo. A especificação técnica (diagramas conceptual e físico, código de criação da base de dados, diagramas de classe), protótipos, especificação de testes, manuais de utilizadores e outros documentos técnicos podem ser disponibilizados como anexo ao documento.*

*O documento principal descreve os objectivos, metodologia e resultados do trabalho. As componentes técnicas não devem ser abordadas no documento principal, mas disponibilizadas como anexo.*