

3 Lab: Modelação do domínio com classes

3.1 Enquadramento

Objetivos de aprendizagem

- Identificar conceitos/classes na descrição de um problema.
- Caracterizar as estruturas de dados de um problema como classes e associações.
- Utilizar associações “simples”, agregações, composições e generalizações.

Preparação

- Informação tutorial: [“What is Class Diagram?”](#)

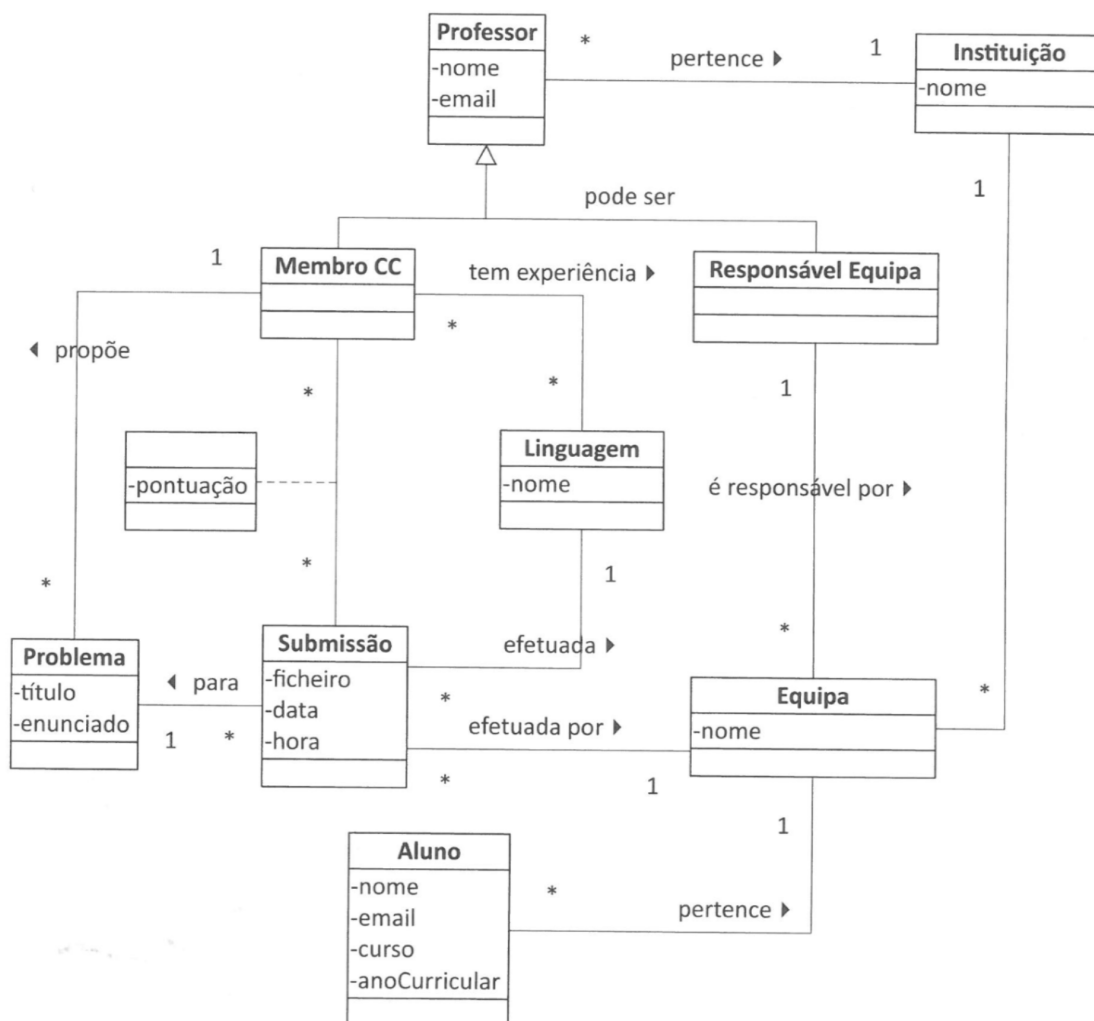
3.2 Análise por objetos de um domínio

No modelo do domínio estamos à procura de “categorias de coisas”, ou conceitos, que são representadas como classes. Para cada categoria vamos encontrar alguns atributos (dados que devem ser memorizados no sistema de informação). Nesta fase, o analista não está preocupado em representar métodos/funções nas classes (como é próprio da programação).

Os conceitos estão relacionados entre si, formando uma rede de conceitos, ligados por associações. O Diagrama de Classes fornece os elementos de modelação para construir esse mapa, seguindo a técnica de análise por objetos.

3.2.1 Conceitos associados a um concurso de programação

- Faça uma leitura interpretativa do diagrama junto (explique o diagrama por palavras suas. E.g. “Cada Equipa tem um Professor responsável.”
- Considerando o modelo representado no diagrama, explique se as seguintes afirmações têm ou não suporte no modelo, isto é, se são V/F face ao que está no diagrama.
 - Todas as Equipas precisam de indicar um Professor responsável. ✓
 - Podem existir Professores que não coordenam nenhuma Equipa.
 - A Entrega (submissão) é feita por vários Alunos.
 - Uma Submissão é avaliada por um Membro do CC.
 - Uma Equipa poder ser composta por alunos de várias Instituições (i.e., a Equipa não é de uma Instituição).
 - Um Membro do CC só pode avaliar entregas resolvidas com linguagens de programação para as quais é especialista.
 - As Entregas de uma Equipa são sempre feitas pelo capitão da equipa.
 - As Entregas de uma Equipa relativa a um Desafio podem ser avaliadas por Docentes diferentes.



Modelo de classes relativo à organização de concursos de programação; CC: Conselho científico das provas. [In: Borges et al, "Modelação de Dados em UML: uma abordagem por problemas."]

3.2.2 Conceitos associados ao funcionamento da Biblioteca

Pesquise os seguintes livros no [catálogo da Biblioteca da UA](#):



- "UML Distilled", de Martin Fowler (3a edição).
- "Use case driven object modeling with UML", de D. Rosenberg (2007).

Crie um diagrama UML para mapear os **conceitos do domínio** relativo ao funcionamento da biblioteca¹, com base na informação consultada e o seguinte conhecimento da área do problema:

- as obras podem ter vários autores. Para desambiguar os autores, usa-se o nome e o ano de nascimento.
- os utilizadores pesquisam obras por autor, título, ano, ou uma combinação desses elementos.
- Um livro pode ser classificado em diferentes tópicos (ou descritores).
- para cada obra, podem existir vários exemplares, com cota e código de barras únicos, que podem ser levantados pelos utilizadores, em regime de empréstimo.
- existem multas para devoluções tardias, mas nem todos os utilizadores têm o mesmo

¹ Neste exercício, pretende-se esboçar o mapa de conceitos. O mais importante é identificar os conceitos e as associações. Por isso, intencionalmente, pode-se omitir a especificação detalhada das classes (i.e.: lista completa de atributos, tipos de dados,...).

tempo para reter os livros em empréstimo domiciliário. Há que distinguir entre utilizadores que são alunos, professores ou utilizadores externos. O tempo de empréstimo normal é de 15, 90 e 30 dias respetivamente. Todos os utilizadores têm um número mecanográfico alfanumérico.

- f) Para inscrever um utilizador externo, é necessário confirmar a sua identidade (contra a apresentação do cartão de cidadão) e a morada (com a apresentação de uma fatura, titulada ao utilizador, do fornecimento de eletricidade, água ou serviço similar).
- g) Os utilizadores podem também pedir a reserva de obras para utilização numa data futura (sendo atribuído o primeiro exemplar disponível).
- h) Existem vários polos (e.g.: Biblioteca Campus Santiago, Mediateca, Biblioteca ESTGA,...) nos quais se encontram os exemplares. Cada polo tem o seu próprio horário de funcionamento.

3.2.3

Considerando as necessidades de informação associadas à gestão de projetos de desenvolvimento de software:

- a) Os projetos têm uma duração prevista, com data de início e de fim bem definidas.
- b) O projeto tem um *Project Leader* atribuído (que é um funcionário) e um departamento responsável. O Eng. Casimiro está responsável pelo projeto “Stock-RFID+” dada a sua especialização em sistemas embebidos.
- c) Cada projeto é organizado em várias tarefas (ou atividades), com uma duração prevista definida. As tarefas são sempre referentes a um projeto, no qual estão definidas.
- d) Um projeto vai ter uma equipa de funcionários atribuída, entre *developers*, especialistas de interação e especialistas de DevOps. No entanto, os funcionários podem estar associados a um projeto por um tempo inferior à sua duração, sendo importante saber o período em que cada um trabalhou nesse.
- e) Também é necessário saber quem esteve envolvido na realização de cada tarefa (uma tarefa pode envolver diferentes pessoas).
- f) Os projetos externos (ao contrário dos projetos internos), têm um cliente bem definido.
- g) Uma tarefa tem de ter uma descrição clara, para todos a entender de forma objetiva. Para além disso, tem uma data de início, de fim, e uma duração esperada. Uma tarefa pode ser dividida em outras tarefas, que a compõe, se ajudar no planeamento e monitorização do projeto.
- h) No contexto de uma tarefa, há lugar a vários registos de intervenção, i.e., um *log* das atualizações/ações que cada funcionário fez no contexto daquela tarefa (e.g.: refinar modelo de casos de uso, versão inicial do UI,...). Em cada intervenção, o funcionário explica o que foi feito e o nr de horas gastas.
- i) As tarefas evoluem de acordo com estados bem definidos: “Em aberto”, “Em progresso”, “Concluída”, ou “Cancelada”.
- j) O Eng. Casimiro está a analisar o desempenho da equipa no projeto “Stock-RFID+”; concluiu que os tempos médios, para cada estado, foram: “Em aberto”, 2 dias; “Em progresso”, 4.5 dias. 60% das tarefas encontram-se concluídas.

Construa um modelo para representar o mapa de informação que se depreende do texto.

Concretize tipos de dados para os atributos. Para além das associações, procure oportunidades para usar: enumerados, atributos derivados, operações (se relevantes para a fase de análise).

3.3 Modelo do domínio da encomenda *online* de comida

Este “atividade-projeto” deve dar origem a um relatório e ser submetido no Moodle.
[\[template\]](#) disponível no Moodle]

3.3.1

Considere a área das encomendas de comida online. Sugere-se, para o efeito, focar a análise num serviço concreto, possivelmente um que já lhe seja familiar.

Desenvolva um modelo do domínio para o caso de estudo que escolheu. O seu modelo do domínio deve ter **a capacidade expressiva suficiente para permitir captar/memorizar a informação necessária aos processos de encomenda e entrega de comida.**

Explore o seu caso de estudo e procure desenvolver **um mapa completo e representativo da informação necessária.** Veja, por exemplo, em um ou mais sites a informação envolvida.

(Considere as técnicas exploratórias apresentadas na secção 3.4 para fazer o levantamento de conceitos candidatos.)

Analise o seu modelo. Certifique-se que a capacidade expressiva do modelo é suficiente para responder aos seguintes requisitos:

- a) Os clientes pesquisam online a oferta de menus/opções e compõem o seu pedido.
- b) A oferta pode envolver diferentes restaurantes parceiros, que é possível pesquisar de forma integrada. (Embora um pedido concreto deva ser confeccionado por um único restaurante.)
- c) O pedido (encomenda) origina um pagamento e uma entrega que é assegurada por um estafeta.
- d) Os clientes podem seguir o progresso do seu pedido, desde que foi criado até que seja satisfeito.
- e) Os menus oferecidos pelos restaurantes parceiros mudam; a própria lista de restaurantes parceiros muda.
- f) O preço dos menus pode mudar de acordo com promoções limitadas no tempo.
- g) Os responsáveis [da plataforma] consultam a evolução diária das encomendas, quer globalmente, quer por código postal.

3.3.2

Verifique se, no seu modelo, há situações em que seja oportuno mostrar a evolução do estado associado a um conceito, i.e., se existem entidades com um ciclo de vida associado.

Modele, para essas entidades, a máquina de estados associada (com um Diagrama de estados).

No caso do problema dos cheques-dentista, abordado em outros exemplos, há um ciclo de vida associado: começa por ser Emitido; depois, entre Em Utilização, em várias consultas; depois, evolui ara Utilizado (na última consulta); depois, fica Pendente para pagamento (ao dentista), e a seguir Pago. Mas, depois de Emitido, também pode transitar para o estado Cancelado.

3.4 Material suplementar

[Larman](#) refere duas estratégias para pesquisar conceitos (objetos do domínio): seguir uma lista de categorias; procurar nomes na descrição do problema (e.g.: na narrativa dos casos de utilização).

Pesquisa de conceitos numa lista de categorias

Categoria (de classes conceptuais)	Exemplos
• Transacções comerciais	

Orientação: Estas são essenciais (envolvem dinheiro), por isso comece com as transações.	Sale, Payment Reservation
<ul style="list-style-type: none"> As entradas no registo de uma transação ("linhas" da transação) Orientação: As transações vêm frequentemente com itens relacionados, por isso considere estes a seguir.	SalesLineItem
<ul style="list-style-type: none"> Produto ou serviço relacionado transacionado Orientação: As transações comportam "coisas" individuais (um produto ou serviço). Considere-as a seguir.	Item Flight, Seat, Meal
<ul style="list-style-type: none"> onde é que a transação é registada? Orientação: Importante.	Register, Ledger FlightManifest
<ul style="list-style-type: none"> papéis das pessoas ou organizações relacionadas com a transação; atores no caso de utilização Orientação: Normalmente precisamos de ter conhecimento sobre as partes envolvidas numa transação.	Cashier, Customer, Store Passenger, Airline
<ul style="list-style-type: none"> local da transação; ponto de serviço 	Store Airport, Plane, Seat
<ul style="list-style-type: none"> eventos que merecem destaque, muitas vezes com uma hora ou lugar que precisamos de guardar 	Sale, Payment Flight
<ul style="list-style-type: none"> objetos físicos Orientação: É especialmente relevante na criação de software de controlo de dispositivos, ou simulações.	Item, Register Airplane
<ul style="list-style-type: none"> contentores de coisas (físicas ou informação) 	Store, Bin Airplane
<ul style="list-style-type: none"> coisas dentro de um "contentor" 	Item Passenger
<ul style="list-style-type: none"> registos contabilísticos, de trabalho, contratos, matéria jurídica 	Receipt, Ledger MaintenanceLog
<ul style="list-style-type: none"> instrumentos financeiros 	Check, LineOfCredit TicketCredit
<ul style="list-style-type: none"> horários, manuais, documentos que são regularmente referidos para a realização de trabalhos 	DailyPriceChangeList RepairSchedule

Pesquisa de conceitos por análise textual

Outra técnica útil (devido à sua simplicidade) é a análise linguística: identificar os substantivos nas descrições textuais de um domínio e considerá-los como classes conceptuais ou atributos candidatos

Alguns destes substantivos são classes conceptuais candidatas, alguns podem referir-se a classes conceptuais que são ignoradas nesta iteração, e alguns podem ser simplesmente atributos de classes.

Um ponto fraco desta abordagem é a imprecisão da linguagem natural; substantivos diferentes podem representar a mesma classe conceptual ou atributo, entre outras ambiguidades.

Fluxo Básico:

1. O Cliente chega a uma caixa POS com artigos para comprar.
2. Caixa inicia uma nova venda.
3. Caixa introduz o identificador do artigo.
4. O sistema regista a linha de venda e apresenta a descrição do item, o preço, e o total provisório. O preço é calculado a partir de um conjunto de regras de preços.

...