



Instituto Politécnico de Coimbra

Instituto Superior de Contabilidade e Administração de Coimbra

LICENCIATURA EM INFORMÁTICA DE GESTÃO

Gestão de Eventos

Trabalho de Linguagens Orientadas por Objetos, em Java

Leonel Fontes 11883

Samuel Anjos 13305

Janeiro, 2015

ÍNDICE

| | |
|---|---|
| ÍNDICE..... | 1 |
| Introdução..... | 2 |
| 1. Diagrama Casos de Uso | 3 |
| 1.1. Descrição do Diagrama Casos de Uso | 3 |
| 2. Diagrama de Classes..... | 5 |
| 2.1. Descrição do Diagrama de Classes | 5 |
| CONCLUSÃO | 7 |

Introdução

A elaboração do presente projeto foi feita no âmbito da disciplina de Linguagens Orientadas por Objetos.

Foi pedido que fosse criada uma aplicação de “Gestão de Eventos”, com o objetivo de gerir eventos para festas de Aniversário e de Pijama. Foi elaborado o Diagrama Casos de Uso que faz uma breve introdução às utilidades da aplicação desenvolvida.

Também foi elaborado um Diagrama de Classes, para relacionar as classes consoante os requerimentos pedidos no trabalho, este tomou uma forma dinâmica à medida que o trabalho foi desenvolvido.

1. Diagrama Casos de Uso

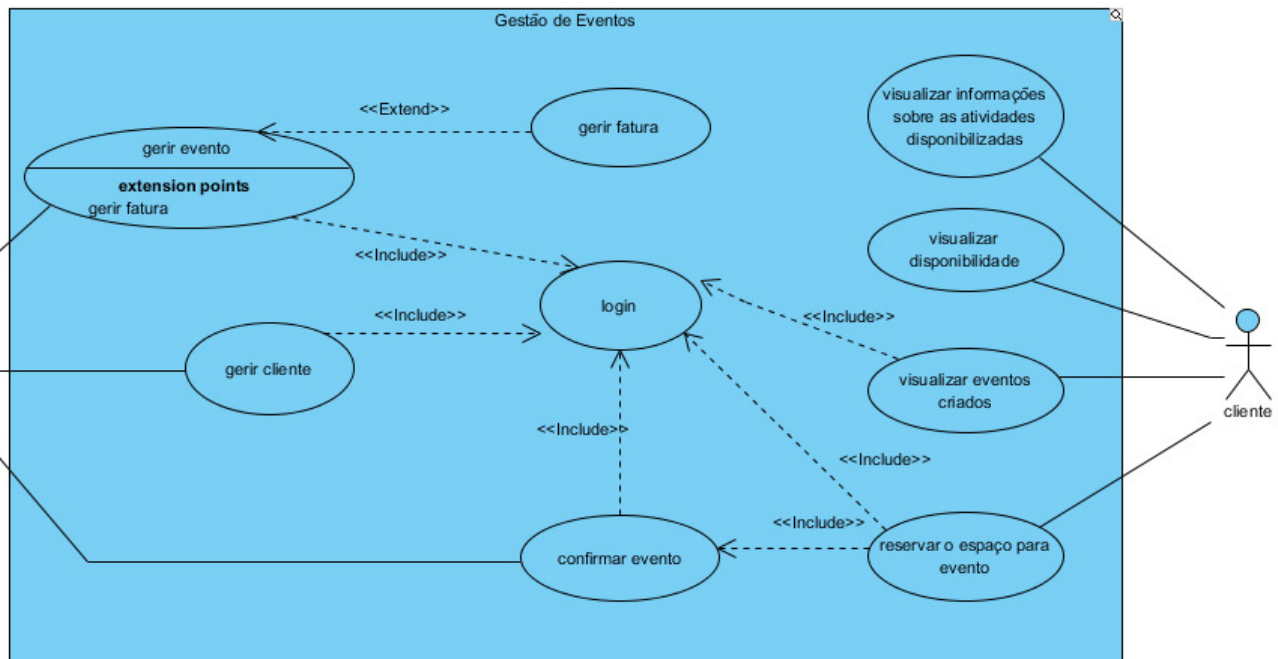


Ilustração 1: Diagrama Casos de Uso

1.1. Descrição do Diagrama Casos de Uso

Criámos dois atores no Diagrama, sendo eles o proprietário e o cliente, pois são eles os utilizadores para que o sistema foi criado com o objetivo de atender às suas necessidades.

O proprietário tem como principais ações: gerir evento, gerir cliente e confirmar evento. Mas para executar qualquer uma destas ações é necessário que proceda ao login, portanto colocámos uma relação de “include”. A ação de gerir cliente consiste em poder criar, alterar e apagar qualquer registo do cliente. A ação gerir evento consiste em criar eventos, pesquisar eventos por id, data, hora e tipo de evento, podendo também remover e alterar. Esta ação também inclui a opção de criar uma fatura após a confirmação do evento, logo gerir fatura “extends” de gerir evento. A ação confirmar evento consiste em o proprietário confirmar o pré-registo do evento criado pelo cliente.

O cliente tem como ação visualizar informações sobre as atividades desenvolvidas, e visualizar disponibilidades, não sendo necessário qualquer tipo de autenticação.

Após a autenticação o cliente vai ter a sua principal ação que é reservar o espaço para um evento, para a qual é necessário alguns requisitos. O cliente pode também visualizar todos os eventos que lhe dizem respeito.

2. Diagrama de Classes

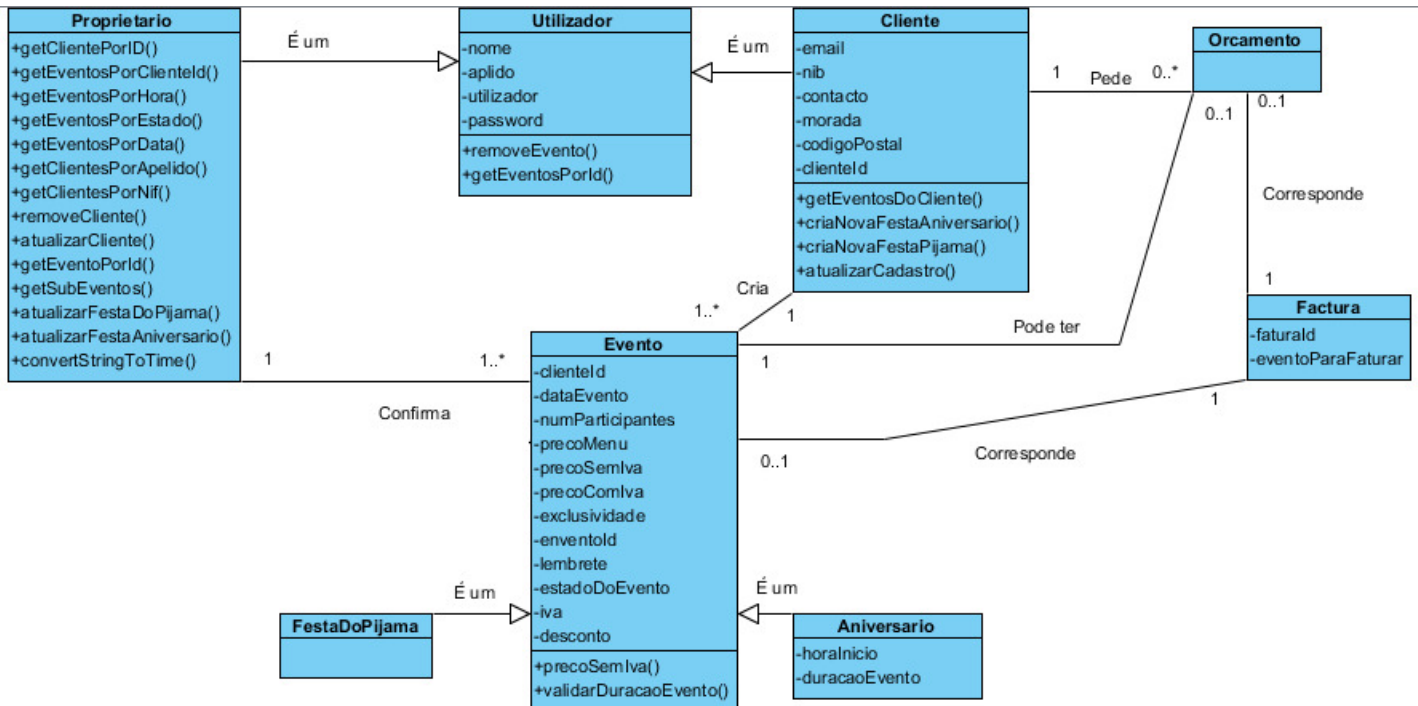


Ilustração 2: Diagrama de Classes

2.1. Descrição do Diagrama de Classes

O Diagrama da Ilustração 2 representa o nosso Diagrama de Classes.

Classe Utilizador: Esta classe foi criada como uma classe genérica para de seguida serem criadas as duas sub-classes **Proprietario** e **Cliente**, e estas são um tipo de **Utilizador**, portanto uma extensão de utilizador.

Classe Cliente: O cliente é um tipo utilizador, e tem alguns relacionamentos, sendo estes com a classe **Evento** onde o cliente pode criar um ou mais eventos. E pode também pedir ou não orçamentos. (classe não codificada no código enviado, classe para futura implementação)

Classe Proprietario: O proprietário é um tipo de utilizador que se relaciona com a classe **Evento**, com a relação de um para um ou mais, ou seja o proprietário pode confirmar um ou

mais eventos. Relaciona também com a classe Cliente, onde um proprietário controla um ou mais clientes.

Classe Evento: Esta classe foi criada como uma classe genérica para de seguida podermos especificar o tipo de eventos realizado, sendo estes representados pelas sub-classes FestaDoPijama e Aniversario. Num evento pode ou não existir um orçamento, tal como um evento pode ou não dar origem a uma fatura, pois o evento pode ou não ser realizado, para a realização do evento é necessário a confirmação do proprietário.

CONCLUSÃO

Ao longo do desenvolvimento deste trabalho, conseguimos perceber que um projeto desta envergadura deve ser muito bem organizado antes de começar, pois é um trabalho onde facilmente os erros de desenho se pagam caro a nível de execução. Foi assim que começamos a ter uma melhor compreensão do paradigma da linguagem orientada por objetos e com um pensamento mais abstrato a nível da solução de problemas informáticos.

Pois no início da nossa jornada a nossa preocupação era como iríamos executar as diversas tarefas pedidas, porque o nosso know-how da orientação a objetos estava ainda muito pouco trabalhado. Ao longo do trabalho fomos percebendo que o conhecimento estava lá mas a visualização do problema era de baixo para cima, com muito trabalho e com a ajuda da professora conseguimos perceber melhor como orientar o projeto a objetos e ter um pensamento mais abstrato a nível de LOO.

Estamos satisfeitos com o resultado do nosso trabalho, pois sabemos que esta aplicação foi um desafio que nos permitiu consolidar os conhecimentos que aprendemos ao longo deste semestre em Linguagem Orientada por Objetos.