**UNIVERSIDADE PAULISTA**

**ALEXANDRE CAPI RAMPANI**

**CLEITON FERNANDO**

**GUILHERME REVOREDO**

**JOÃO HENRIQUE LEVADA**

**LETÍCIA CRISTINA SOARES**

**PROJETO INTEGRADO MULTIDISCIPLINAR (4º SEMESTRE) – CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**ARARAQUARA**

**2018**

**ALEXANDRE CAPI RAMPANI**

**CLEITON FERNANDO**

**GUILHERME REVOREDO**

**JOÃO HENRIQUE LEVADA**

**LETÍCIA CRISTINA SOARES**

**PROJETO INTEGRADO MULTIDISCIPLINAR (4º SEMESTRE) – CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

Trabalho de conclusão do quarto semestre do curso de Análise e desenvolvimento de sistemas apresentado à UNIP.

Orientadora: Prof. Kamila Rios

**ARARAQUARA**

**2018**

*[epígrafe]*

**Resumo**

O objetivo principal deste trabalho é apresentar os detalhes de implementação e planejamento que levaram à construção de um sistema de helpdesk para empresas prestadoras de serviços na área de manutenção de computadores.

Ao longo do documento, portanto, serão apresentadas as camadas MVC (model, view e controller) que compõem o software, bem como todo o seu processo de desenvolvimento – desde o levantamento de requisitos até a implementação - o foi orquestrado por técnicas de gestão de projetos orientadas pelo PMBOK.

Por fim, será desenvolvido um estudo acerca da criação e manutenção de uma startup, modelo de empresa com o qual o sistema em questão se integraria perfeitamente, dada sua efetividade e a flexibilidade daquele modelo empresarial.

**Abstract**

The main objective of this work is to present the implementation and planning details that led to the construction of a helpdesk system for companies that provide services in the area of computer maintenance.

Throughout the document, therefore, the MVC (model, view and controller) layers that make up the software, as well as its entire development process - from requirements gathering to implementation - will be presented. It has been orchestrated by management techniques projects guided by the PMBOK.

Finally, a study will be developed on the creation and maintenance of a startup, a business model with which the system in question would integrate perfectly, given its effectiveness and the flexibility of that business model.

Sumário

[Lista de Figuras 7](#__RefHeading___Toc2717_1355530280)

[Lista de Tabelas 8](#__RefHeading___Toc2719_1355530280)

[1.INTRODUÇÃO 9](#__RefHeading___Toc2721_1355530280)

[2.PROJETO DE SISTEMAS ORIENTADO A OBJETO 11](#__RefHeading___Toc83_1187944198)

[2.1 Diagrama de casos de uso 11](#__RefHeading___Toc85_1187944198)

[2.2. Diagrama de classes 12](#__RefHeading___Toc87_1187944198)

[2.3.Diagramas de comunicação 12](#__RefHeading___Toc89_1187944198)

# **Lista de Figuras**

# Lista de Tabelas

# 1.INTRODUÇÃO

1. Com o desenvolvimento e popularização da internet em meados dos anos 2000 em desktops e agora, na década de 2010, em smartphones, a maneira como o software é projetado e implementado vem mudando de maneira bastante substancial. A ideia de que um sistema computacional é representado por tudo aquilo que está na máquina do usuário – bibliotecas e interface - caiu por terra com a adoção de novos modelos arquiteturais no desenvolvimento de produtos de software, que exploram com mais eficácia a infraestrutura proporcionada pelo protocolo HTTP e pela estabilidade cada vez maior das conexões à rede mundial de computadores.
2. Neste sentido, o sistema P4, que será apresentado ao longo deste trabalho, foi alicerçado sobre essas premissas arquiteturais e de desenvolvimento web. Serão pormenorizados, portanto, os detalhes das tecnologias empregadas em sua construção e a maneira como foram implementadas a fim de que o sistema tenha a qualidade esperada pelos clientes, ou seja, com escopo bem definido e fiel aos requisitos levantados no início do projeto, bem como com tempo e custo coerentes com o estipulado.
3. O modelo arquitetural escolhido para sustentar o projeto foi o MVC (model, view, controller), graças à sua capacidade de dividir responsabilidades dentro da comunicação entre as partes do sistema e garantir que haja redução da complexidade de sua manutenção. A concretização desse padrão arquitetural se deu por meio do REST, modelo cujo princípio é extrair todo o potencial do protocolo HTTP.
4. Dentro desse contexto, a linguagem utilizada para desenvolvimento backend da API responsável pela comunicação entre os módulos do sistema foi o Java, devido à sua flexibilidade já reconhecida no desenvolvimento web, quando em associação com frameworks voltados também para desenvolvimento web. No caso, optou-se pelo framework Spring, cujo módulo MVC dá respaldo a projetos de pequeno porte, mas também é respeitado no contexto de desenvolvimento de aplicações Java Enterprise.
5. A fim de garantir um desenvolvimento consciente do software e com retrabalho reduzido, utilizou-se o arcabouço teórico proposto pela engenharia de software no que diz respeito o projeto de sistemas orientado a objetivo. Para isso, a diagramação UML foi adotada com o objetivo de balizar a implementação em todos os seus aspectos.
6. Por fim, e com o objetivo de promover a organicidade e coesão durante o desenvolvimento, foram adotadas técnicas de gestão de projeto para manter o escopo sob controle da equipe. Algumas premissas do guia PMBOK que se mostraram adequadas ao baseline do projeto foram adotadas, bem como ferramentas e filosofias de desenvolvimento ágil, garantindo entregas concretas e periódicas.

# 2.PROJETO DE SISTEMAS ORIENTADO A OBJETO

# Um software de qualidade pressupõe sistematização no seu desenvolvimento, e é justamente a fim de garantir essa solidez que a engenharia de software e outras técnicas de desenvolvimento são empregadas na construção de sistemas computacionais. Dentro desse ferramental fornecido aos desenvolvedores e projetistas, pode-se destacar a orientação a objetos como um importante filão da engenharia de software.

# Conceitos como os de herança, classes e objetos foram adotados pelos engenheiros de software para que a concepção do sistema como um todo pudesse acompanhar as técnicas de orientação a objetos, cujo objetivo é aproximar o software do mundo real. Nascem daí as noções de análise de sistemas orientada a objeto e projeto de sistemas orientado a objeto, com a função de garantir aos envolvidos no projeto uma base sólida para sua evolução.

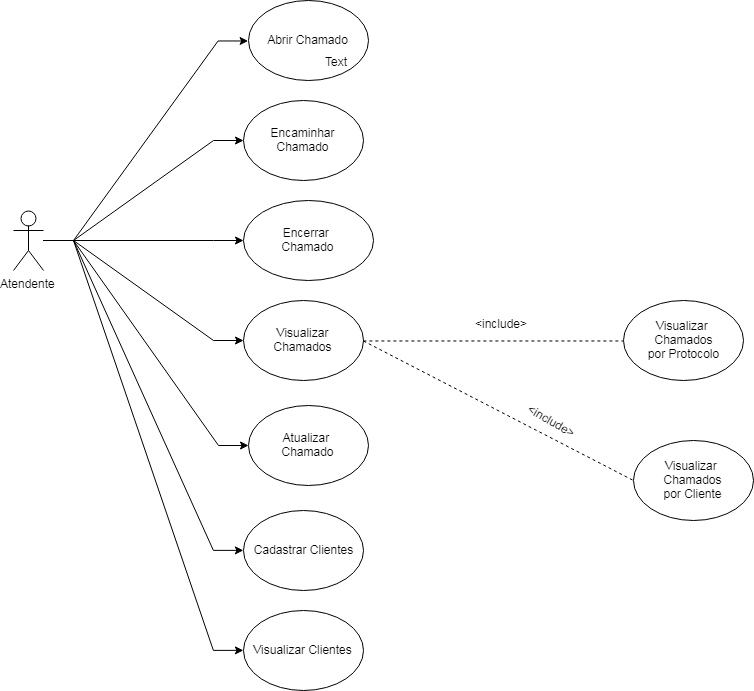
# Por análise de sistemas orientada a objeto, compreende-se o processo por meio do qual são identificados os pontos importantes do software, com base nos dados colhidos na fase inicial do projeto, como na fase de levantamento de requisitos, por exemplo. Essa informação construída pela etapa de análise é imprescindível para consolidação da próxima etapa, escopo deste capítulo: o projeto de sistemas orientado a objeto.

# Se a etapa de análise representa o passo em que se chega às informações importantes a nível técnico sobre o sistema, transcendendo o levantamento de requisitos e as informações de alto nível, o projeto de análise compreende o momento a partir do qual são traçadas estratégias para interligar as informações obtidas anteriormente, com a finalidade de se chegar ao resultado esperado pelo cliente. Pode-se resumir os dois processos sucintamente: enquanto um faz o levantamento de problemas, o outro procura valer-se de meios para solucioná-los.

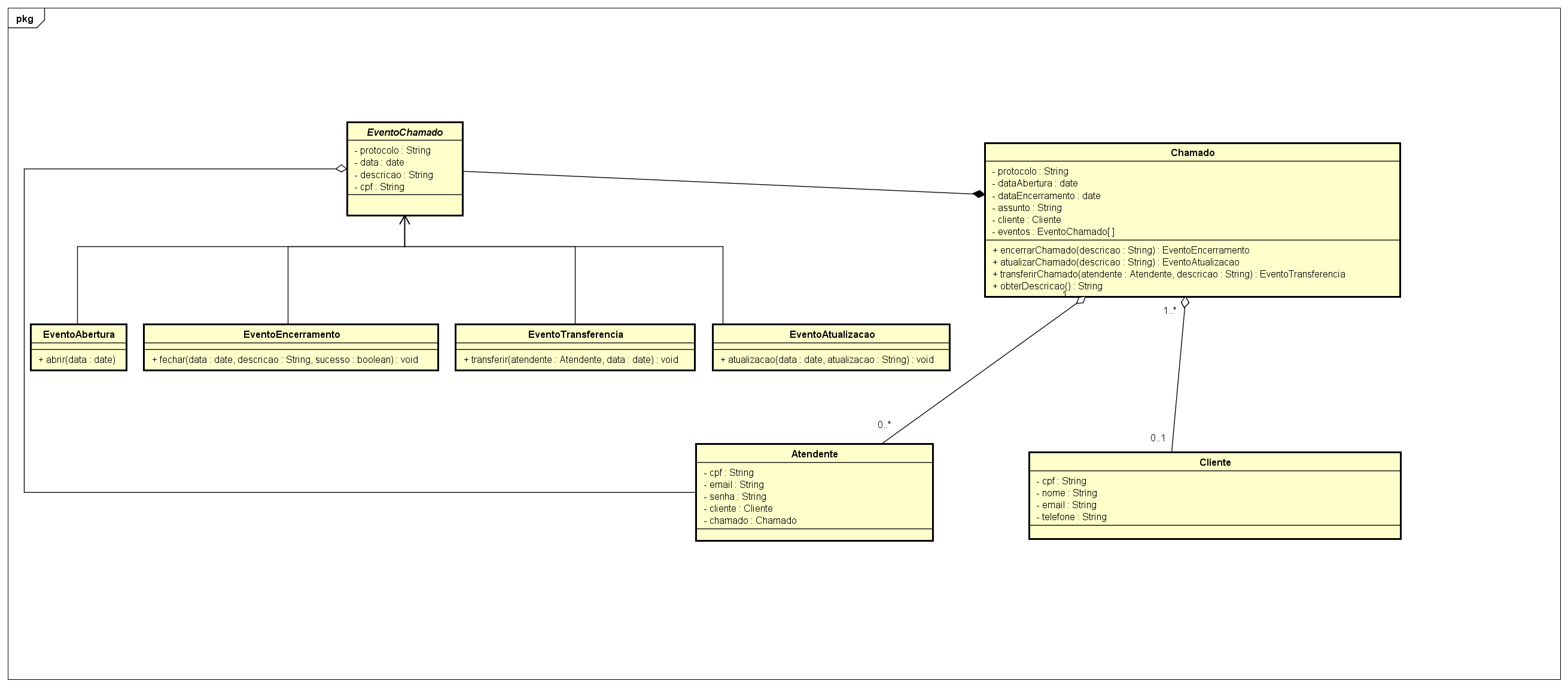
# Para subsidiar essas atividades, a linguagem UML é adotada na manipulação dos diversos diagramas que compõem o desenvolvimento do sistema desde a análise, conforme a imagem abaixo:

# 

## 2.1 Diagrama de casos de uso



## 2.2. Diagrama de classes



## 2.3.Diagramas de comunicação

