

**UNIVERSIDADE DE UBERABA**  
**ENGENHARIA DE SOFTWARE**  
**TRABALHO FINAL**

**ALUNO: VITOR DE AZAMBUJA RIBEIRO FRANCO**  
**R.A: 5153344**

**SISTEMA: SISTEMA DE GERENCIAMENTO DE HOTÉIS**

**ANÁLISE DE REQUISITOS**

**REQUISITOS FUNCIONAIS**

1. Cadastro e Gerenciamento de Hóspedes: O sistema deve permitir o cadastro de novos hóspedes incluindo informações como nome, endereço, contato e documentos. O sistema também deve permitir atualizar ou excluir informações dos hóspedes.
2. Reserva de Quartos: O sistema deve permitir que os usuários façam reservas de quartos com base em datas e outras preferências, visualizem a disponibilidade desses quartos em tempo real e cancelem ou alterem as reservas já realizadas.
3. Check-in e Check-out: O sistema deve permitir que os usuários possam realizar o check-in e sejam atribuídos a um quarto ou check-out, no qual o valor da estadia é automaticamente calculado.
4. Gerenciamento de Quartos: O sistema deve permitir o cadastro e a atualização das informações dos quartos como número, tipo, preço e status e deve permitir a alteração dos status do quarto como disponível, ocupado, manutenção
5. Gerenciamento de Funcionários: O sistema deve permitir o cadastro de funcionários incluindo informações como cargo, nome e horário de trabalho. O sistema também deve gerenciar as permissões de acesso ao próprio sistema.
6. Faturamento e Pagamento: O sistema deve emitir faturas detalhadas para os hóspedes ao fazer check-out e permitir o registro de diferentes formas de pagamento como cartões de crédito, dinheiro, PIX
7. Relatórios: O sistema deve poder gerar relatórios financeiros, de ocupação e de manutenção.
8. Serviços Adicionais: O sistema deve permitir o registro de serviços extras contratados pelo hóspede (como lavanderia, spa, serviço de quarto). O sistema deve adicionar automaticamente os custos adicionais dos serviços na fatura do hóspede.

**REQUISITOS NÃO FUNCIONAIS**

1. Segurança: O sistema deve garantir a segurança dos dados dos hóspedes e dos funcionários usando criptografia e autenticação de dois fatores para usuários com poderes administrativos.
2. Desempenho: O sistema deverá suportar até 500 usuários simultâneos sem perda de desempenho. O sistema também deverá processar solicitações de buscas de quartos em menos de 2 segundos.
3. Escalabilidade: O sistema deverá ser escalável para gerenciar vários hotéis em diversas localidades. O sistema também deve suportar com facilidade a implementação de novos recursos.

4. Compatibilidade: O sistema deverá ser compatível com todos os principais e mais modernos navegadores e deve ser acessível tanto em ambientes desktop quanto em ambientes mobile.
5. Usabilidade: O sistema deverá possuir uma interface intuitiva que possibilita usuários sem treinamento técnico a utilizarem. O sistema também deverá possuir mais de um idioma a ser selecionado (português e inglês ou espanhol)

## **PLANEJAMENTO**

### **CRONOGRAMA DE DESENVOLVIMENTO**

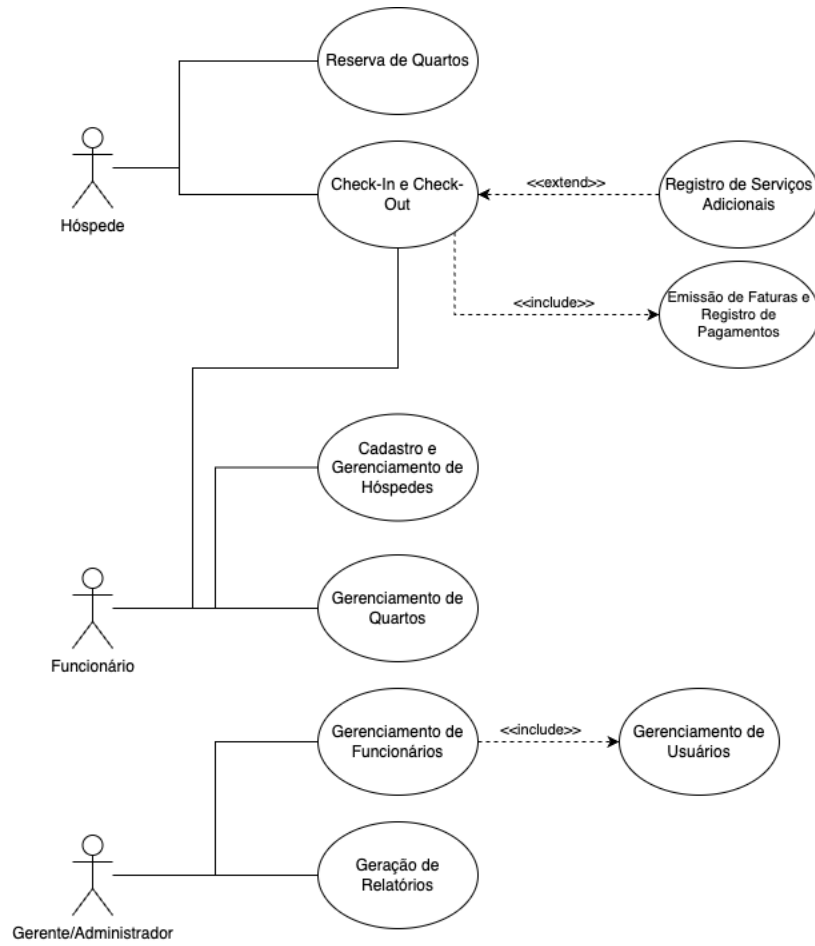
1. Levantamento de Requisitos: 2 semanas
2. Análise e Especificação: 3 semanas
3. Projeto e Modelagem: 4 semanas
4. Implementação: 6 semanas
5. Testes e Depuração: 3 semanas
6. Implantação: 2 semanas

### **RECURSOS NECESSÁRIOS**

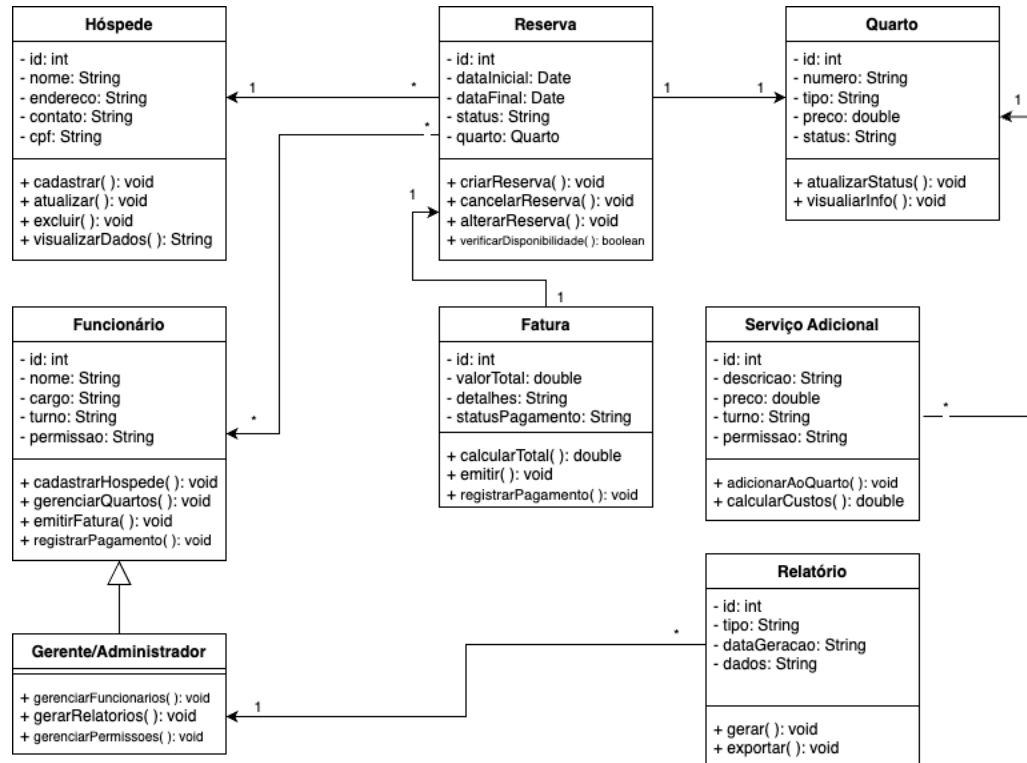
1. Equipe de Desenvolvimento: 4 desenvolvedores, 1 analista de negócios, 1 designer de interface, 1 gerente de projetos e 1 testador QA.
2. Ambiente de Desenvolvimento: IDE, ferramenta de controle de versão, servidor de banco de dados, servidor web e servidor de teste, ferramenta de integração contínua e ferramenta para gerenciamento de dependências.
3. Ferramentas de comunicação: E-mail, Chat de Equipe, Videoconferência, ferramenta de gerenciamento de projeto (Jira, Trello, Asana) e ferramenta de documentação (Confluence ou Google Docs)

# MODELAGEM

## DIAGRAMA DE CASOS DE USO



## DIAGRAMA DE CLASSES



## IMPLEMENTAÇÃO

O sistema será desenvolvido utilizando a linguagem de programação Java e o framework Spring Boot para camada de aplicação e criação da API REST. O banco de dados será implementado usando MySQL, tendo sua comunicação gerenciada pelo Spring Data JPA. A interface de usuário será desenvolvida utilizando HTML, CSS, JavaScript, framework Bootstrap e AJAX. A arquitetura do sistema seguirá o padrão MVC (Model-View-Controller).

## TESTES

Serão realizados testes unitários, de integração, de aceitação, de desempenho e carga e de segurança. Para os testes unitários serão utilizados JUnit e Mockito; para os testes de integração serão utilizados Spring Test e Testcontainers; para os testes de aceitação serão utilizados Selenium e Cucumber; para os testes de desempenho e carga serão utilizados o JMeter e Gatling; para os testes de segurança serão utilizados OWASP ZAP e Burp Suite. Ao utilizar todas as ferramentas em conjunto, tem-se a garantia de um sistema robusto, funcional e seguro antes mesmo da implantação.

## CONCLUSÃO

Em resumo, ao desenvolver um sistema de gerenciamento de hotéis, observa-se o uso de várias práticas comuns na Engenharia de Software. Análise de requisitos, planejamento, modelagem, implementação e testagem foram as etapas trabalhadas com o objetivo de listar, planejar e mapear o desenvolvimento completo de um software. Dos conceitos aprendidos, foram utilizados a definição de requisitos funcionais e não funcionais, a etapa de planejamento para definir prazos

e recursos, a modelagem dos diagramas de caso de uso e de classes, a implementação usada para definir as linguagens, frameworks utilizados e o padrão MVC e por fim a definição dos testes e como eles serão executados para garantir um software de qualidade. Enfim, foram utilizados vários conceitos envolvendo boas práticas com o objetivo de garantir um produto final de qualidade, de qualidade e que atende ao objetivo pelo qual foi proposto.

## REFERÊNCIAS

### Bibliografia

Advantages of Java. Disponível em: <<https://www.ibm.com/docs/hu/aix/7.2?topic=monitoring-advantages-java>>. Acesso em: 4 dez. 2024.

DO DESENVOLVEDOR, C. Como realizar a Análise de Requisitos de Software? Disponível em: <<https://blog.casadodesenvolvedor.com.br/analise-de-requisitos-de-software/>>. Acesso em: 4 dez. 2024.

FILHO, M. Ferramentas de testes JAVA. Disponível em: <<https://www.dio.me/articles/ferramentas-de-testes-java>>. Acesso em: 4 dez. 2024.

LUCID SOFTWARE PORTUGUÊS. Tutorial de diagramas de classes UML. Disponível em: <<https://www.youtube.com/watch?v=rDidOn6KN9k&t=621s&pp=ygUTZGhhZ3JhbWEgZGUgY2xhc3Nlcw%3D%3D>>. Acesso em: 4 dez. 2024a.

LUCID SOFTWARE PORTUGUÊS. Tutorial de Caso de Uso UML. Disponível em: <<https://www.youtube.com/watch?v=ab6eDdwS3rA&pp=ygUXZGhhZ3JhbWEgZGUgY2FzbyBkZSB1c28%3D>>. Acesso em: 4 dez. 2024b.

Requisitos funcionais e não funcionais: o que são? Disponível em: <<https://www.mestresdawe.com.br/tecnologias/requisitos-funcionais-e-nao-funcionais-o-que-sao>>. Acesso em: 4 dez. 2024.