



TADS - Análise e Desenvolvimento de Sistemas

## Projeto Web

Nomes: Lucas Ryu Muraoka, Matheus Ferreira, Gustavo Aquino e  
Matheus Oldoni

Professor: Carlos Alberto Veríssimo Pereira

## Identificação da Equipe

Id da Equipe: GR13 – SpringTools	
Nome	Papel
Matheus Ferreira	Gerente de Projetos / Arquiteto de Software / Desenvolvedor Back-End.
Lucas Ryu	Arquiteto de Software / D.B.A.
Gustavo Aquino	Q.A. / Desenvolvedor Back-End.
Matheus Oldoni	Q.A.

## **Especificações do Sistema**

Na nossa aplicação o usuário entra em nossa comunidade podendo ser registrado para poder criar as publicações e seus comentários na comunidade online. Possuindo um sistema de pontos conforme as publicações e comentários sejam feitos para que os usuários sejam ranqueados e por fim as postagens podem ser buscadas pela sua categoria quando foi criada.

O usuário pode usar as operações CRUD para criar as postagens e os comentários também podendo deletar as postagens e comentários criados pelos próprios e por fim podem buscar certa postagem pela sua categoria.

O sistema possui a arquitetura MVC para organização dos pacotes e REST na construção do back-end(API) implementada pelo framework Spring e estabelecendo a conexão com o Banco de Dados MySQL, mas no repositório se encontra a versão com o H2 Database, pois o projeto ainda continua em desenvolvimento.

## **Cenário**

Imagine um aplicativo de uma comunidade online onde as postagens e os comentários são feitos pelos próprios usuários, permitindo aos usuários interagirem na plataforma online mas para obter o acesso na comunidade é necessário se registrar. E assim com o usuário registrado ele poderá participar do sistema de pontuações onde cada postagem e comentário possui um valor de pontos e assim tendo um ranque de pontos.

Com isso em mente a comunidade se torna um lugar onde as pessoas poderão conversar sobre assuntos diversos pelas postagens e cada um podendo comentar, e se o usuário preferir pode buscar os assuntos preferidos pelas categorias onde cada postagem possui uma.

## **Planejamento**

Semana 1 a 4:

Nas 3 primeiras semanas do projeto, foram decididos basicamente as funções de cada um dos integrantes do grupo, juntamente com as tecnologias que foram usadas. Durante os dailys, foram inseridos os propósitos do sistema, as nossas intenções com esse sistema e o público alvo. Por ser um sistema público, precisávamos tentar visualizar certas dores, assim, como seria o mercado e a concorrência com esse sistema já em atividade.

Nas demais semanas:

Durante as demais semanas, foi iniciado o desenvolvimento do projeto, tais como as telas, a criação de API's e toda a refatoração dos códigos, também veio sendo desenvolvido os desenhos da arquitetura de software, a inserção das tecnologias, aplicação da estrutura MVC, inserindo os princípios de Clean Code, onde foram feitos diversos testes até o momento final do projeto, que foi obtido as expectativas esperadas.

## Situação do Problema

Ao iniciarmos o projeto do blog como parte do nosso trabalho acadêmico, nos deparamos com diversos desafios relacionados ao desenvolvimento de software. Nossa principal preocupação era criar um blog que atendesse às necessidades dos usuários, fosse intuitivo e oferecesse uma experiência agradável.

Um dos desafios era escolher as tecnologias adequadas para o desenvolvimento do blog. Consideramos diferentes opções, levando em conta a nossa familiaridade com as tecnologias, a eficiência, a segurança e a escalabilidade. Além disso, queríamos garantir que o blog pudesse ser facilmente mantido e evoluído no futuro.

Outro desafio estava relacionado à arquitetura do sistema. Queríamos seguir boas práticas de desenvolvimento, como a estruturação do projeto em camadas (MVC) e a aplicação de princípios de design limpo (Clean Code). Essas decisões tinham o objetivo de garantir um código bem organizado, de fácil compreensão e manutenção.

A segurança também era uma preocupação importante. Queríamos implementar um sistema de autenticação e autorização robusto, garantindo que apenas usuários autorizados pudessem acessar certas funcionalidades do blog. Além disso, precisávamos proteger os dados dos usuários e evitar vulnerabilidades que pudessem comprometer a segurança do sistema.

## Solução de Software

Como grupo de programadores, decidimos utilizar tecnologias como Java, Spring Boot, Spring Security e banco de dados H2 para construir o nosso blog como parte do trabalho acadêmico. Essas escolhas nos permitiram desenvolver um sistema robusto, seguro e de fácil manutenção, seguindo a estrutura de projeto MVC e aplicando os princípios de Clean Code.

Iniciamos utilizando Java, uma linguagem de programação com a qual estamos familiarizados e que possui uma ampla comunidade de desenvolvedores. Em seguida, optamos pelo Spring Boot, que simplificou bastante a configuração inicial do projeto, economizando tempo e permitindo que nos concentrássemos mais na lógica de negócio do blog.

Para garantir a segurança do nosso blog, implementamos o Spring Security. Com essa tecnologia, criamos um sistema de autenticação e autorização confiável, restringindo o acesso apenas a usuários autenticados. Dessa forma, somente pessoas autorizadas podem criar, editar e excluir postagens, além de interagir com outros usuários através de comentários.

Para armazenar os dados do blog, escolhemos o banco de dados H2. Essa opção leve e de fácil configuração foi perfeita para o ambiente de desenvolvimento do projeto. No futuro, se necessário, podemos migrar facilmente para um banco de dados mais robusto, como MySQL ou PostgreSQL.

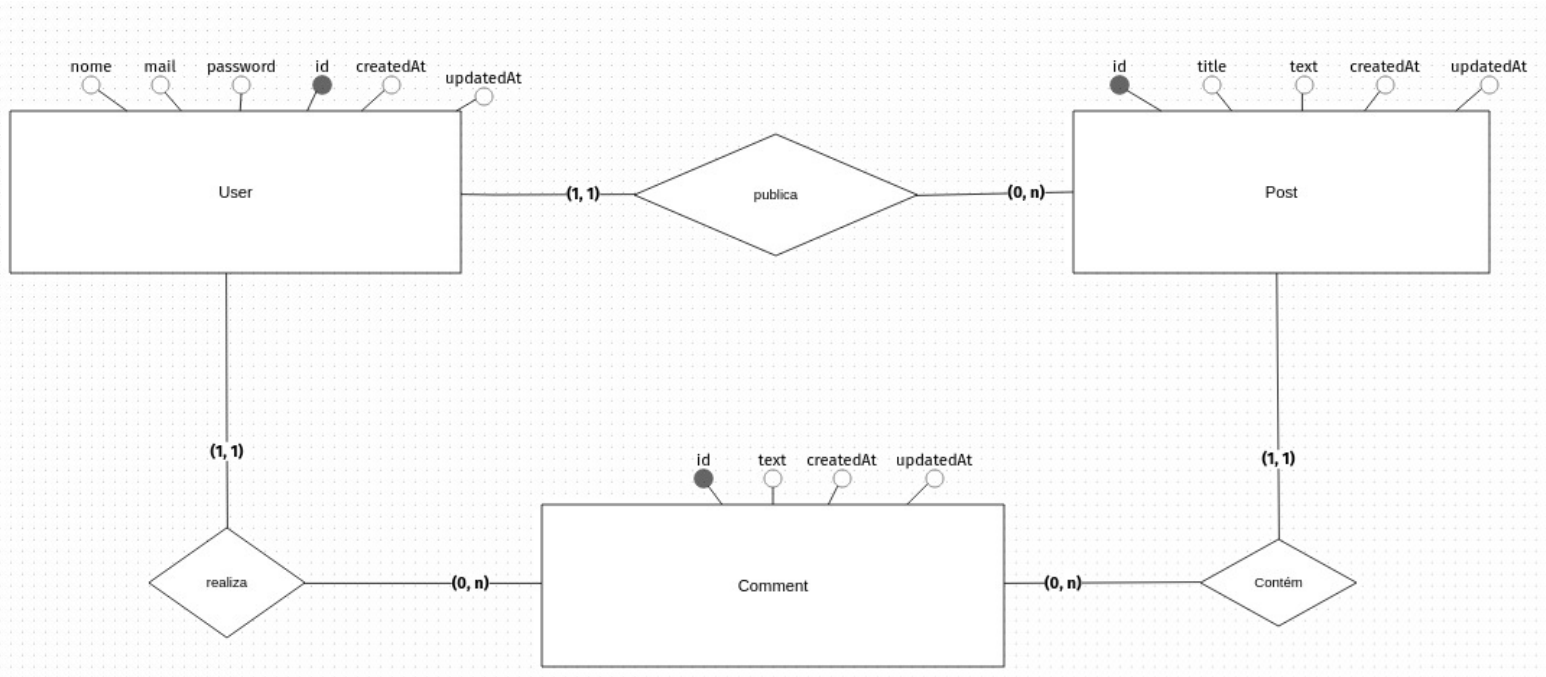
Seguindo a estrutura de projeto MVC, organizamos nosso código em diferentes camadas. O modelo (Model) representou as entidades do domínio, como postagens e comentários. As visualizações (View) cuidaram da apresentação dos dados aos usuários. E, por fim, os controladores (Controller) trataram das requisições dos usuários, coordenando a lógica de negócio e interagindo com o modelo e a visualização.

Durante o desenvolvimento, aplicamos os princípios de Clean Code, escrevendo um código legível, coeso e de fácil manutenção. Utilizamos boas práticas de nomenclatura, adicionamos comentários explicativos e dividimos a lógica do blog em funções e classes concisas. Essas práticas garantiram um código de qualidade, facilitando a compreensão e evolução do projeto ao longo do tempo.

Em resumo, como grupo de programadores, utilizando as tecnologias Java, Spring Boot, Spring Security e banco de dados H2, conseguimos desenvolver um blog robusto, seguro e de fácil manutenção para o nosso trabalho acadêmico. A estrutura de projeto MVC e a aplicação dos princípios de Clean Code proporcionaram uma arquitetura organizada e um código de qualidade. Essas escolhas e práticas de desenvolvimento contribuíram para o sucesso do projeto e para o aprimoramento das nossas habilidades como programadores.



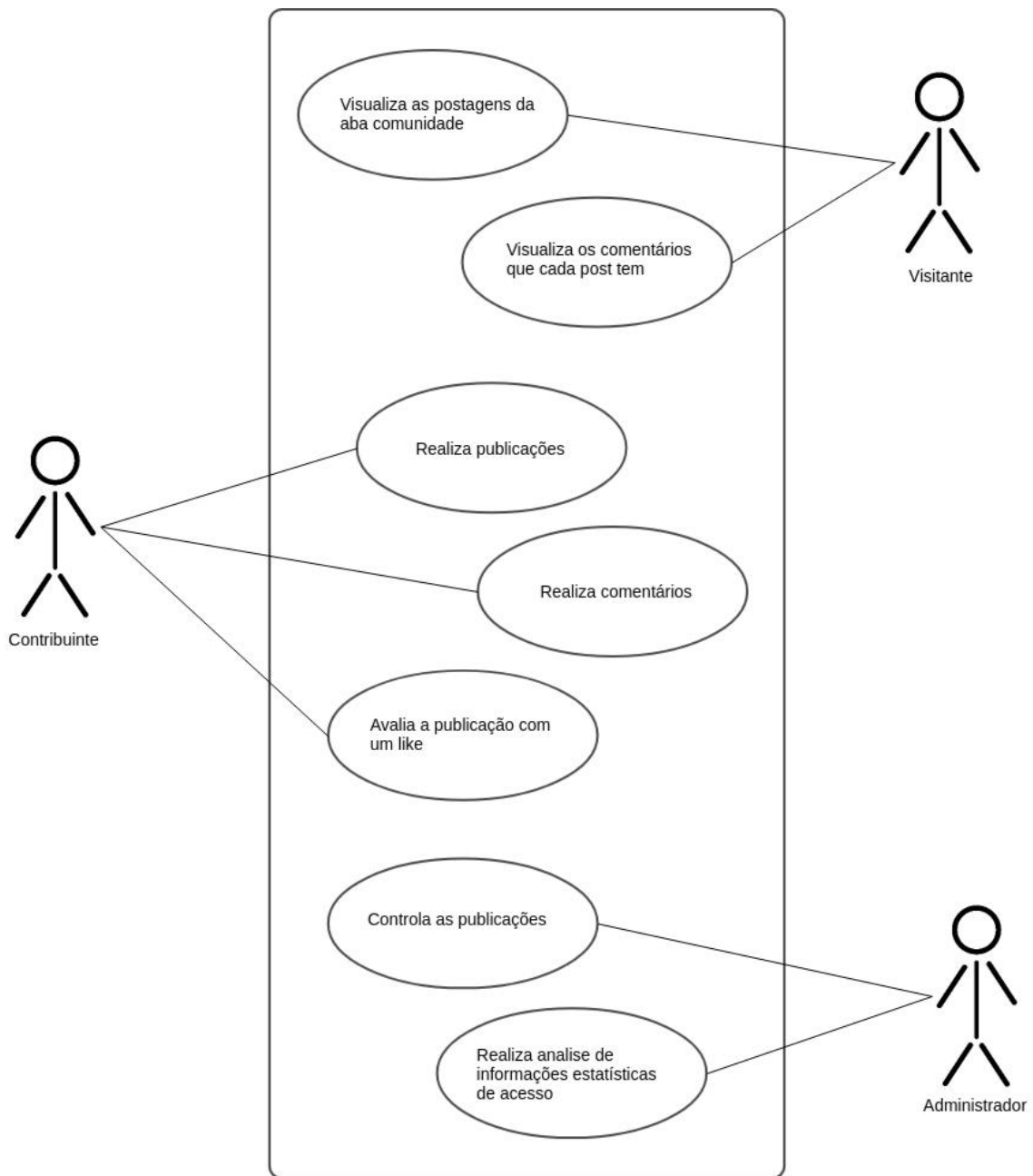
# Modelo de Dados



Disponível também no GitHub para melhor visualização:

<https://github.com/Matheus-FSantos/projeto-java-web/tree/main/Modelagem/Banco%20de%20Dados>

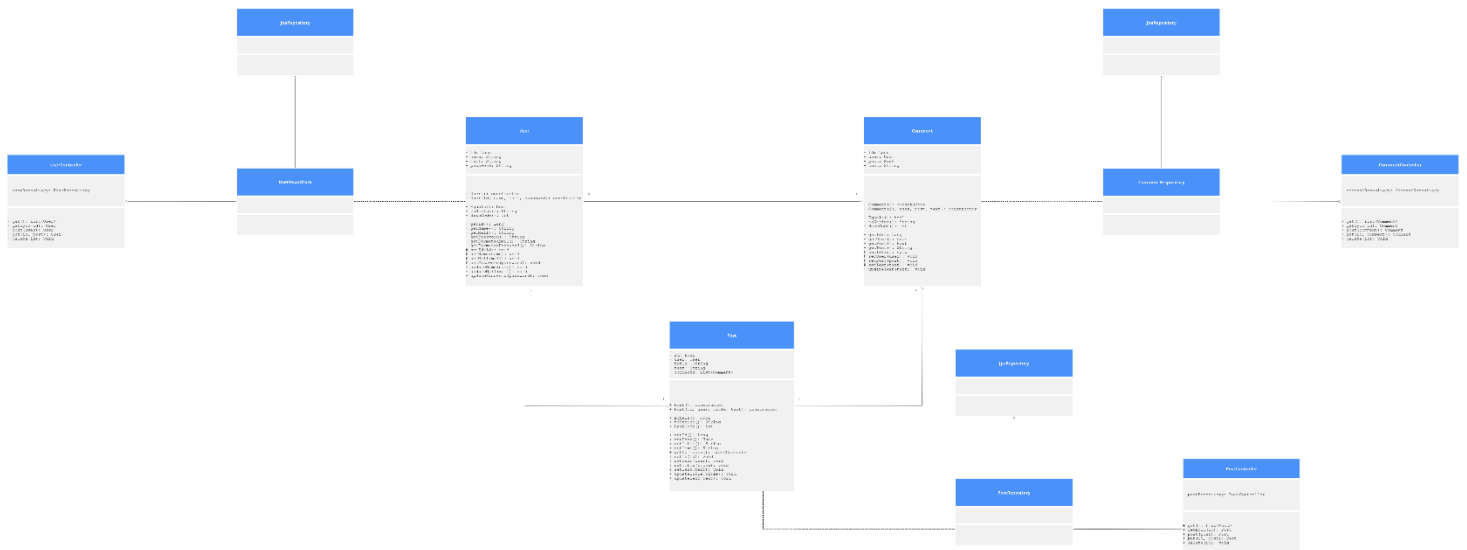
## Caso de Uso



Disponível também no GitHub para melhor visualização:

<https://github.com/Matheus-FSantos/projeto-java-web/tree/main/Modelagem/Use%20Case>

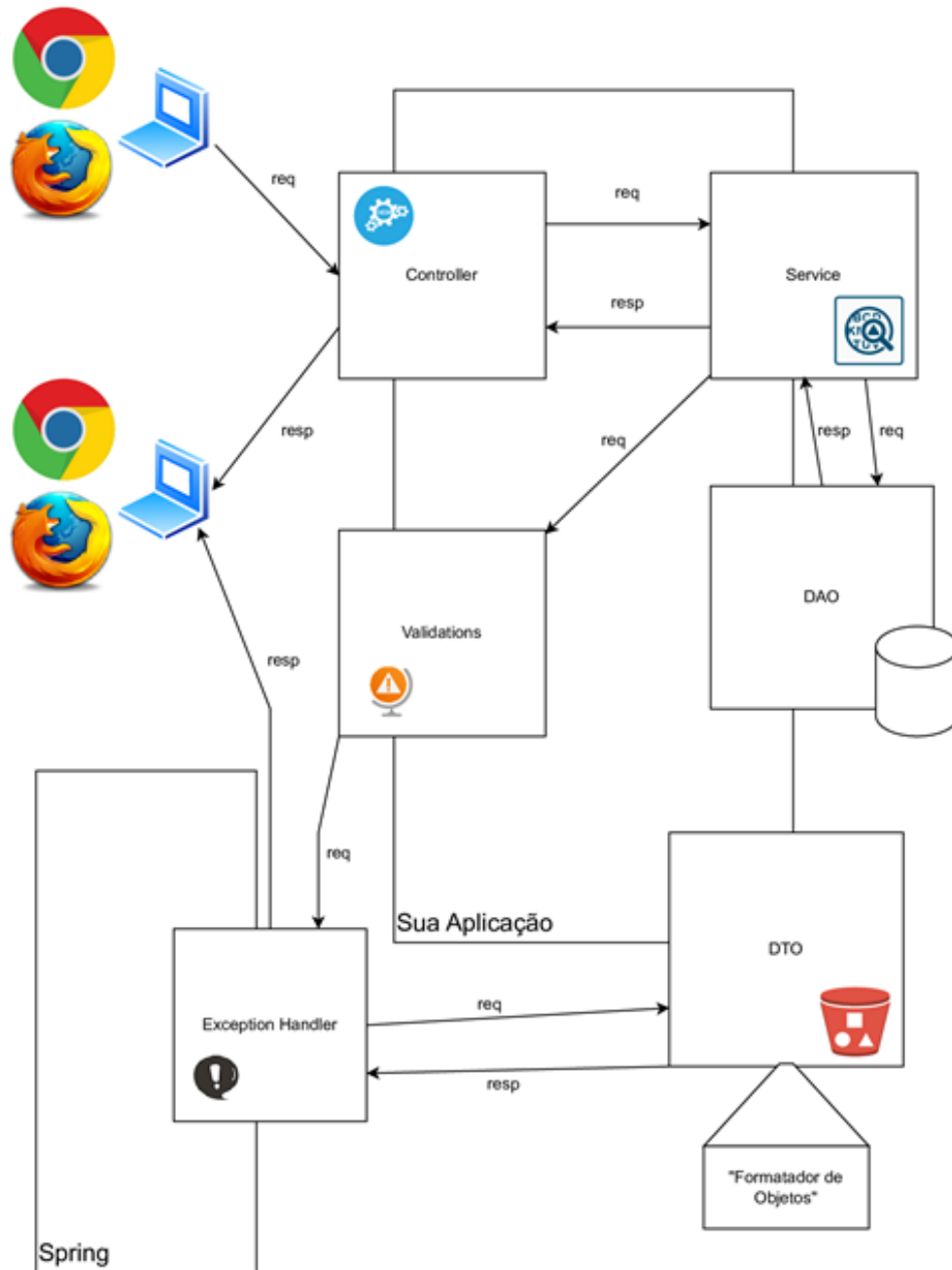
## Diagrama de Classes



**Disponível também no GitHub para melhor visualização:**

<https://github.com/Matheus-FSantos/projeto-java-web/tree/main/Modelagem/UML>

## Desenho da Arquitetura



Disponível também no GitHub para melhor visualização:

<https://github.com/Matheus-FSantos/projeto-java-web/tree/main/Modelagem/Arquitetura>

## Conclusão

Como grupo de programadores, o desenvolvimento do blog como parte do nosso trabalho acadêmico foi uma experiência enriquecedora. Ao utilizar tecnologias como Java, Spring Boot, Spring Security e banco de dados H2, conseguimos criar um sistema robusto, seguro e de fácil manutenção.

Durante o processo, aplicamos a estrutura de projeto MVC e seguimos os princípios de Clean Code, o que resultou em um código organizado, legível e de qualidade. Além disso, a escolha cuidadosa das tecnologias nos permitiu desenvolver um blog com recursos avançados, como autenticação de usuários e interação por meio de comentários.

Ao trabalhar em equipe, aprendemos a importância da colaboração e comunicação eficiente. Cada membro contribuiu com suas habilidades e conhecimentos, permitindo um desenvolvimento ágil e eficaz. Também enfrentamos desafios ao longo do caminho, mas fomos capazes de superá-los por meio do trabalho em equipe e da busca por soluções criativas.

O desenvolvimento do blog nos proporcionou uma oportunidade valiosa de aplicar nossos conhecimentos em programação e desenvolvimento de software em um contexto prático. Aprendemos a tomar decisões fundamentadas sobre as tecnologias a serem utilizadas, a definir uma arquitetura sólida e a implementar recursos avançados para fornecer uma experiência de usuário agradável.

Além disso, o projeto do blog nos permitiu aprimorar nossas habilidades de trabalho em equipe, comunicação. Aprendemos a coordenar nossos esforços, dividir tarefas e resolver problemas de forma eficiente.

No geral, o desenvolvimento do blog para o trabalho acadêmico foi uma experiência enriquecedora. Aprendemos não apenas sobre as tecnologias e práticas de desenvolvimento de software, mas também sobre a importância do trabalho em equipe. Estamos orgulhosos do resultado alcançado com nossos esforços.