Projeto e Implementação de um Serviço Web RESTful com Técnicas de Segurança

Claudiomar Araújo

Orientador: Raoni Kulesza

João Pessoa, 12 de novembro de 2018

Sumário

Introdução

Comunicação para transferir informação

Internet

Web

Engenharia de software no desenvolvimento de sistemas

Tema

Software tem sido alvo de ataques em diversos setores

Sistemas comprometidos podem afetar definitivamente organizações e seus clientes

Tema

Em 2013, foi assinada uma ordem nos EUA para aprimorar a segurança contra ataques cibernéticos

Existe incentivo para aprimorar segurança? Dados foram reunidos e analisados

Romanosky, 2016
Journal of Cybersecurity, Oxford University

Tema

Dados

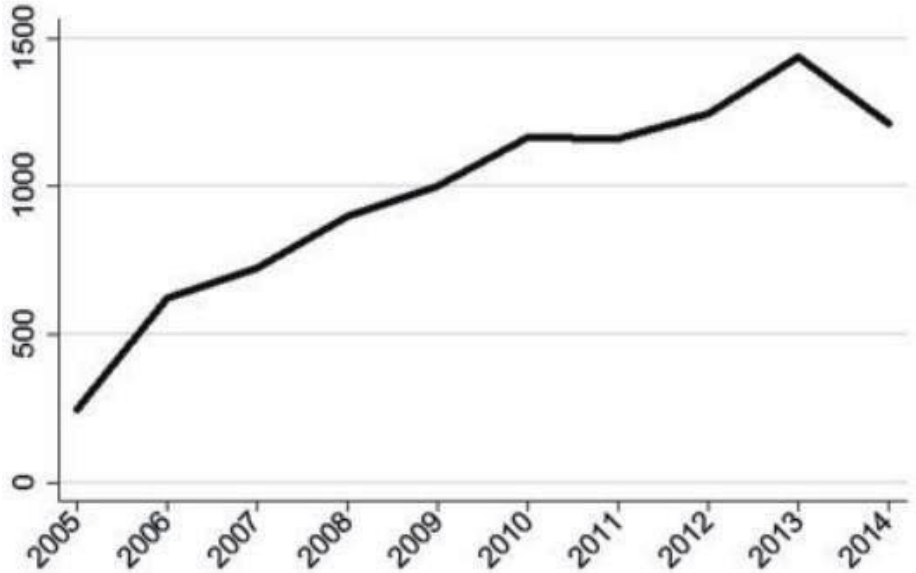
12000 eventos cibernéticos divulgados publicamente, separados por tipos

(Romanosky, 2016)

“Cibernética é a ciência que estuda como informação é comunicada em máquinas e dispositivos eletrônicos.”

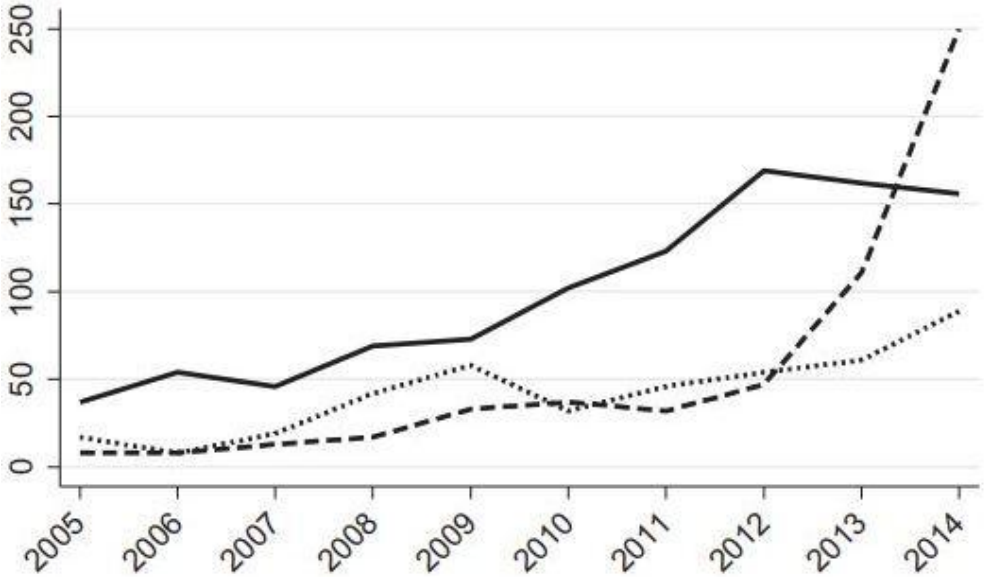
(Cambridge, 2018)

Tema



Vazamento de dados

Tema



Incidentes de segurança

Problema

Sistemas Web

É preciso tratar segurança no cliente e no servidor. Existe necessidade de desenvolvimento de tecnologias e contramedidas de segurança. (Groef, 2016)

Projeto de Software

Requisitos de segurança podem ser atendidos de forma mais eficiente durante o projeto de software. Desenvolvedores também são responsáveis, não só especialistas. (OWASP, 2018)

Mentalidade de Segurança

É preciso adquirir uma mentalidade de segurança para melhor percepção e compreensão. (Hibishi, 2016)

Objetivo Geral

Utilizar técnicas de segurança para avaliar requisitos em sistemas Web do ponto de vista de projeto e implementação de software.

Objetivos Específicos

1. Pesquisar o histórico de sistemas Web para compreender sua evolução até a atualidade.
2. Estudar o framework Spring para utilizá-lo como principal tecnologia Web no desenvolvimento de um sistema.
3. Apresentar conceitos e a importância da segurança de sistemas computacionais, com foco em sistemas Web.
4. Projetar e implementar um sistema Web e avaliar seus requisitos com técnicas de segurança.

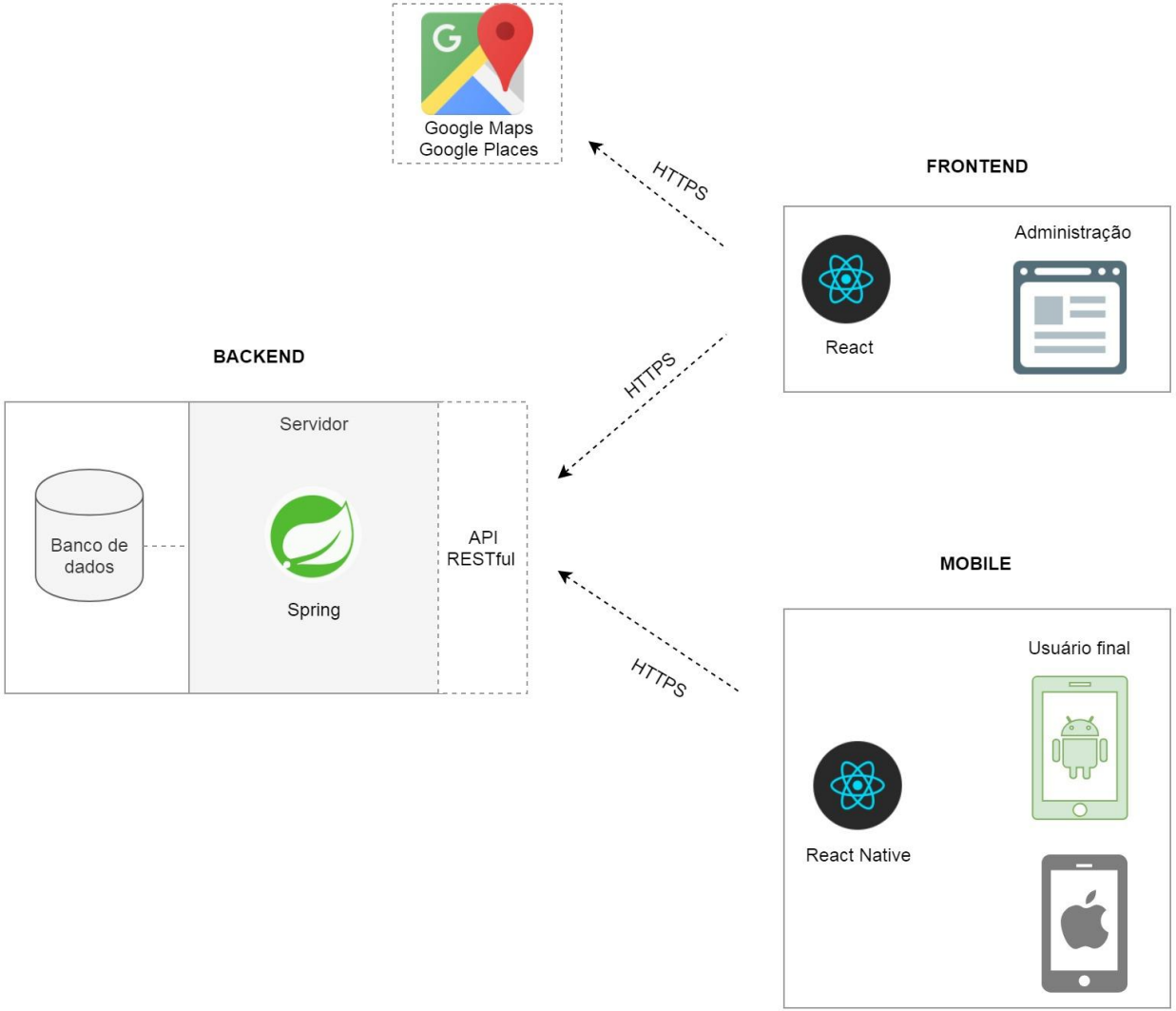
Estudo de Caso: Você Digital

LAViD **TCE-PB**

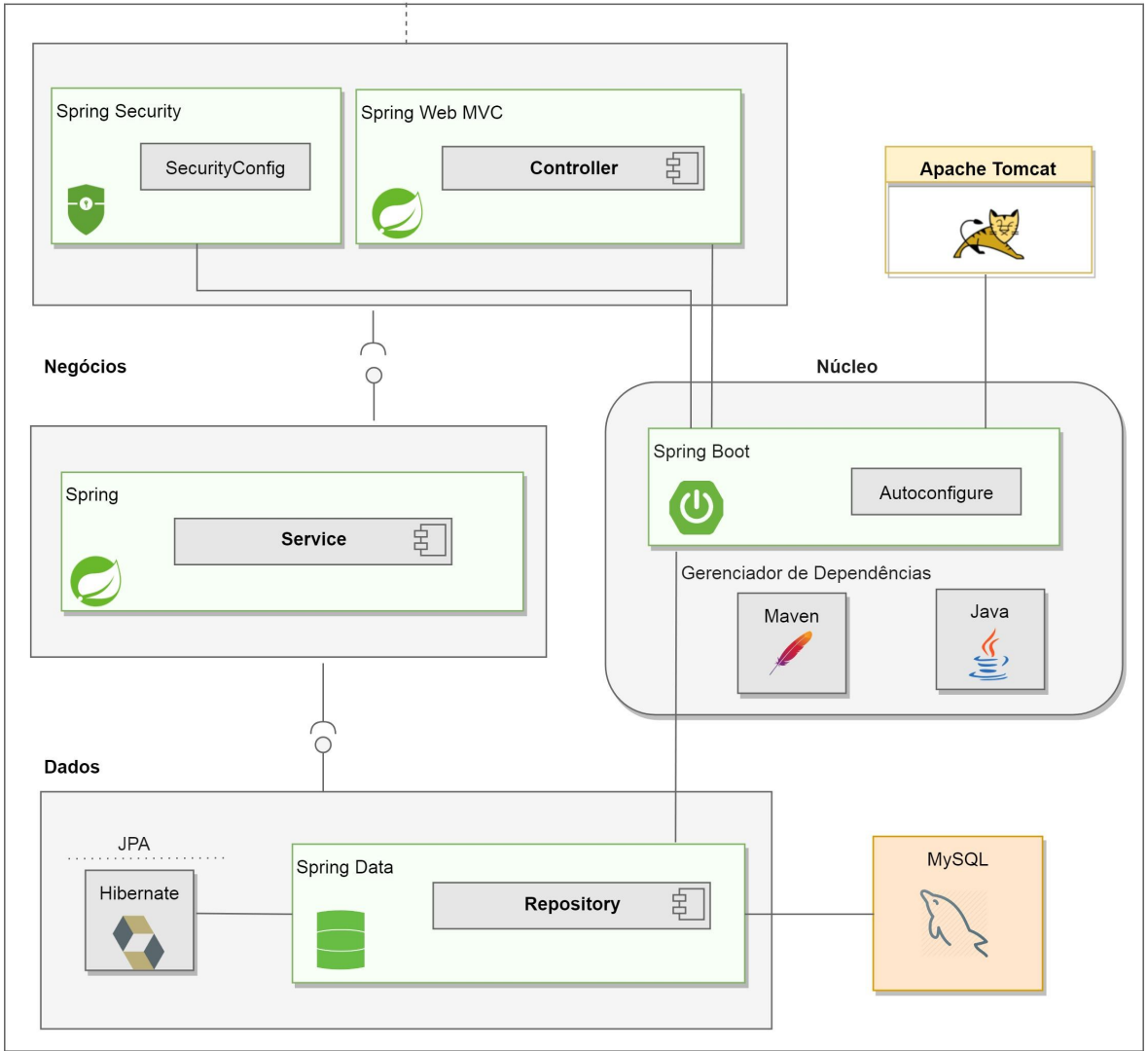
Sistema de escuta popular para melhor interação e comunicação entre a sociedade e gestores públicos.

Identificar problemas em áreas de gestão pública: educação, saúde e segurança.

Pessoas poderão avaliar órgãos públicos com uma nota quantitativa, um comentário, e enviar imagem, áudio e vídeo.



Arquitetura de alto nível Você Digital



Arquitetura
de baixo nível
backend
Você Digital

Avaliação

Você Digital

Requisitos são avaliados e implementados com técnicas de segurança

Requisitos de segurança

Provenientes de padrões industriais, leis e histórico de vulnerabilidades.
Permitem reuso de padrões e suas melhores práticas

Avaliação

Open Web Application Security Project

Projeto de segurança para software aberto à comunidade

OWASP

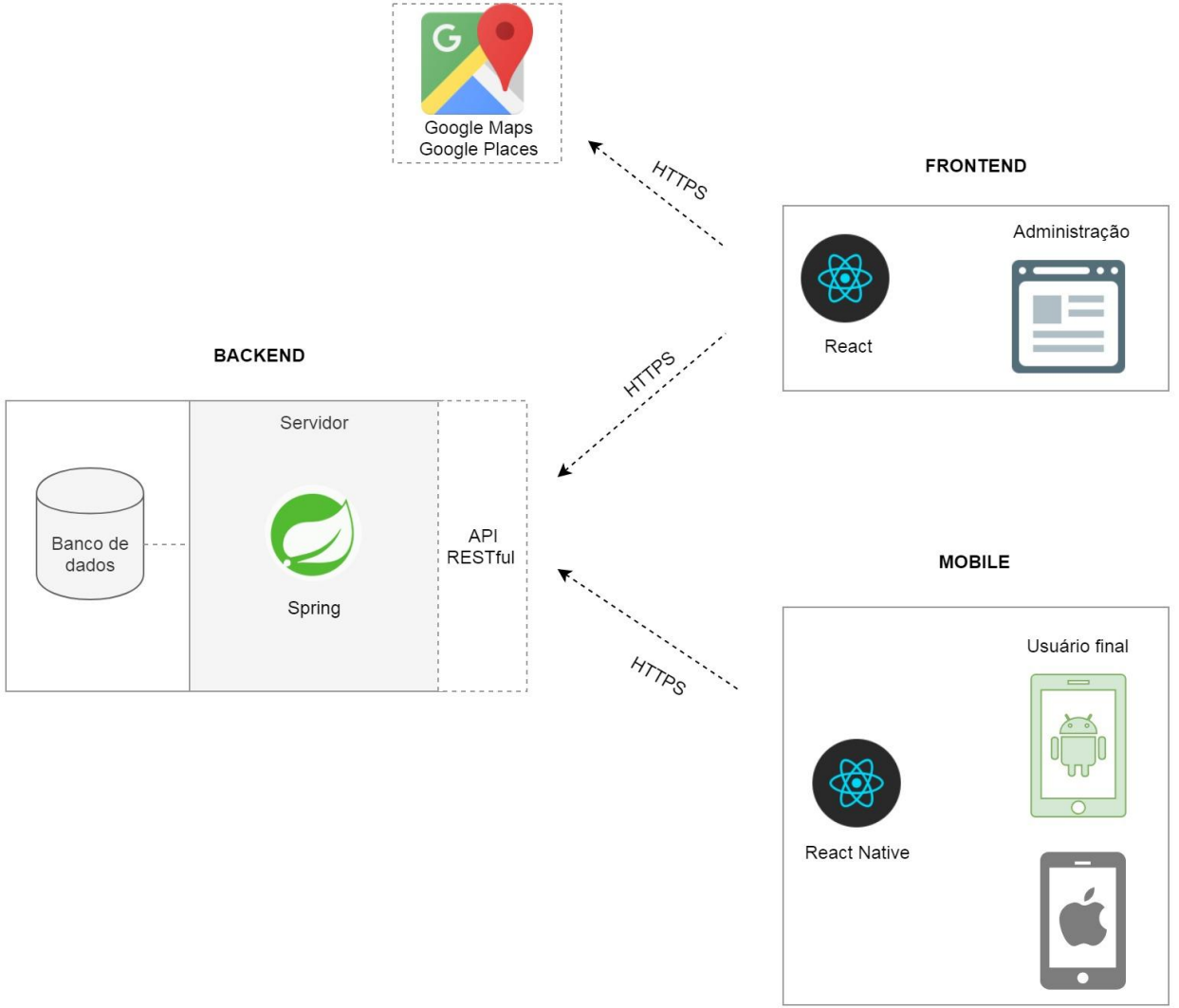
Controles Proativos

Técnicas de segurança para desenvolvedores com controles proativos

OWASP, Pro Active Controls For Developers 2018

Identidade Digital

Autenticação



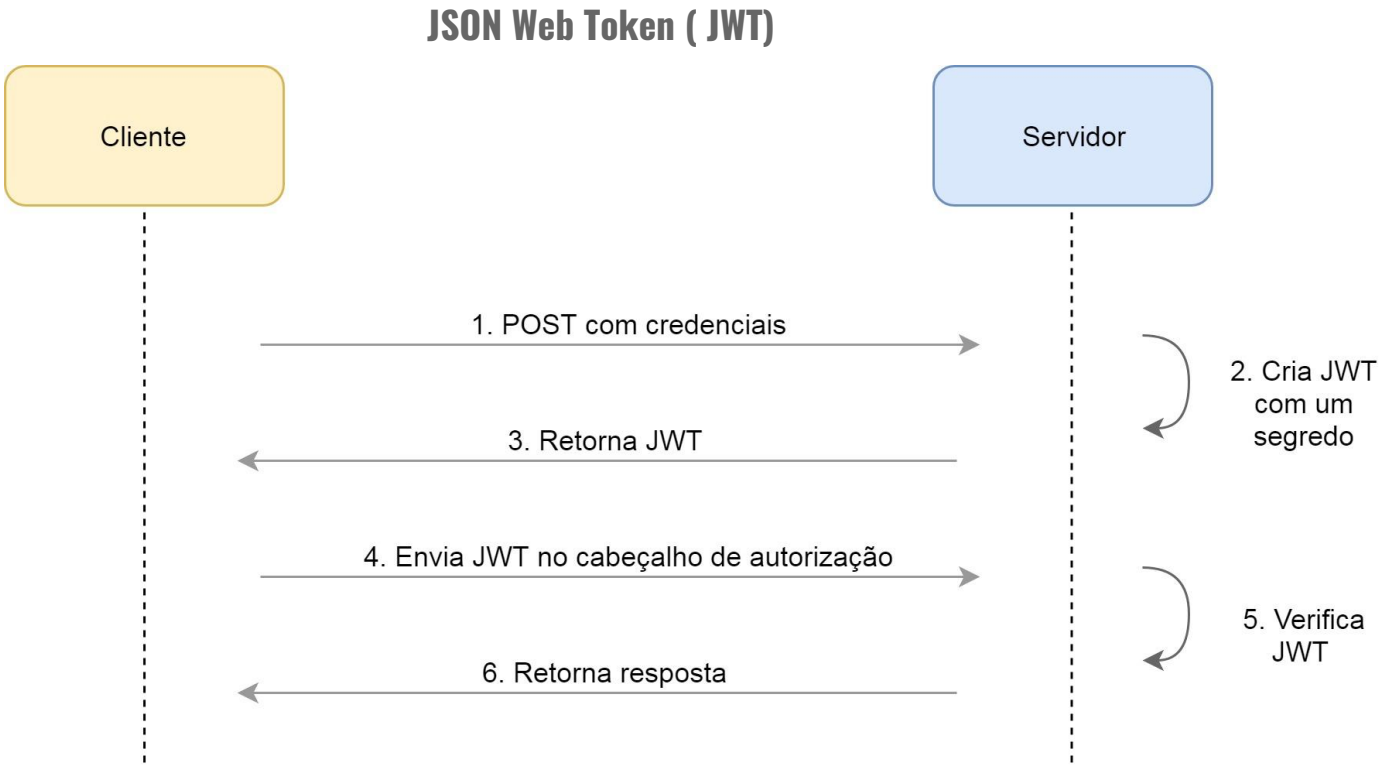
Identidade Digital

JSON Web Token (JWT)

Criptografia hash

Assinado digitalmente

Identidade Digital



Identidade Digital

JSON Web Token (JWT)

POST

```
{
  "email": "email@email.com",
  "senha": "0000000000"
}
```

Identidade Digital

JSON Web Token (JWT)

Resposta


```
{
    "accessToken" :
    "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiIxMzMiLCJpYXQiOjE1NDA5MjYzM
    TIsImV4cCI6MTU0MTUzMTExMn0.k4lFKgA5Nk9-jHAk4qyPx2JZfVeGumeC
    rPXdohA30gxT0J1RfC45LF5dSm8ihbjYVTBpY8cQ7Y72qNxXGuqYnQ",
    "tokenType" : "Bearer"
}
```

Controle de Acesso (Autorização)

Papéis de usuários: Administrador e usuário comum

Spring Security permite assegurar por URL, classe e **método**

Exemplo:

1. Spring possui uma instância da identidade do usuário que fez a requisição.
2. No método, uma anotação define os papéis permitidos.
3. Se o usuário não possuir um dos papéis permitidos, o acesso é negado.

Controle de Acesso

usuario	
id	...
1	...
2	...

papel	
id	nome
1	Administrador
2	Usuário

papeis_usuario	
id_usuario	id_papel
1	1
2	2

Papéis de usuários em modelo relacional

Proteger Dados

Comunicação

Criptografar dados em trânsito

Transport Layer Security (TLS)

Hand Shake Protocol

Código de Autenticação de Mensagem (MAC)

HTTP sobre TLS: HTTP Secure

Componentes criptografados:

1. URL do conteúdo
2. conteúdo requisitado
3. formulários
4. cookies compartilhados entre cliente e servidor
5. conteúdos de cabeçalho HTTP

Proteger Dados

Senha

Dado sensível armazenado no banco de dados com criptografia *hash*

BCrypt

Possui um parâmetro para personalizar o fator de trabalho

Faz uso de *sal*

Proteger Dados

Senha	Hash
pass123456	$2a$10$LqbDQfYjMBNc.lEGSouwV.p0acvxlQoUB2UVDju.cvxnLlYl2Yw9K
pass123456	$2a$10$9IqBzNUSSL6UkwrVQiFcW.liSl9mmMWzreYP8j2yRNFaJ3woTXFHK

Resultado de *hashes* gerados com BCrypt

Acesso ao Banco de Dados

Configuração

Comunicação

Consultas

```
String query = "SELECT * FROM conta
WHERE id_cliente='" +
request.getParameter("id") + "'";
```

```
'or '1'='1
```

```
String query = "SELECT * FROM conta
WHERE id_cliente='' or '1'='1'";
```

Outros Controles

Validar entradas

Sintaxe e semântica

Tratar erros e exceções

Evitar publicar informações internas do sistema com erros de pilha

Devolver respostas adequadas ao usuário

Conclusões

A **principal contribuição** foi o desenvolvimento de um **serviço Web** que considerou mecanismos de segurança.

Foram apresentadas soluções de segurança existentes e utilizadas pela comunidade, aplicando especificamente para os requisitos do sistema estudado.

Foi possível perceber a importância da seleção de tecnologias durante a fase de projeto de software, tendo feito reuso das funcionalidades disponibilizadas pelo framework Spring.

Percebe-se como a segurança se relaciona com diferentes partes de software, colaborando com sua segurança como um todo.

Conclusões

São proposta as seguintes **melhorias**:

Registros de segurança, inclusive com monitoramento em tempo real.

Autenticação multifator pode ser considerada para maior segurança.

Testes de segurança automatizados de fora da aplicação podem ser realizados para melhor validação do sistema, inclusive com ferramentas existentes próprias para isso.

Trabalhos Futuros

Compreensão de **sistemas Web atuais** e suas **técnicas de segurança**, mesmo para outras arquiteturas e tecnologias.

O modelo de **autenticação** apresentado pode ser utilizado como base para **outros protocolos** entre aplicações, inclusive comunicação em arquitetura de microserviços.

São sugeridos novos estudos para definir uma metodologia, com o apoio de **ferramentas de automação**, para avaliar o **nível de segurança** de sistemas Web.

Projeto e Implementação de um Serviço Web RESTful com Técnicas de Segurança