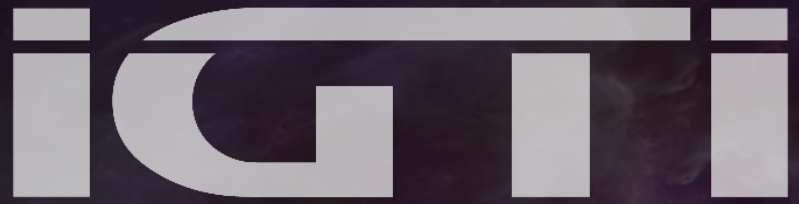


A aula interativa de **Desenvolvimento de APIs** começará em breve!

Atenção:

1) Acesse a aula com NOME COMPLETO, para que sua frequência seja computada.
2) Mantenha o microfone DESLIGADO, abrindo-o apenas em momentos de interatividade.
3) Mantenha seu vídeo sempre ATIVADO.

Desenvolvimento de APIs

PRIMEIRA AULA INTERATIVA

PROF. GUILHERME ASSIS

Nesta aula

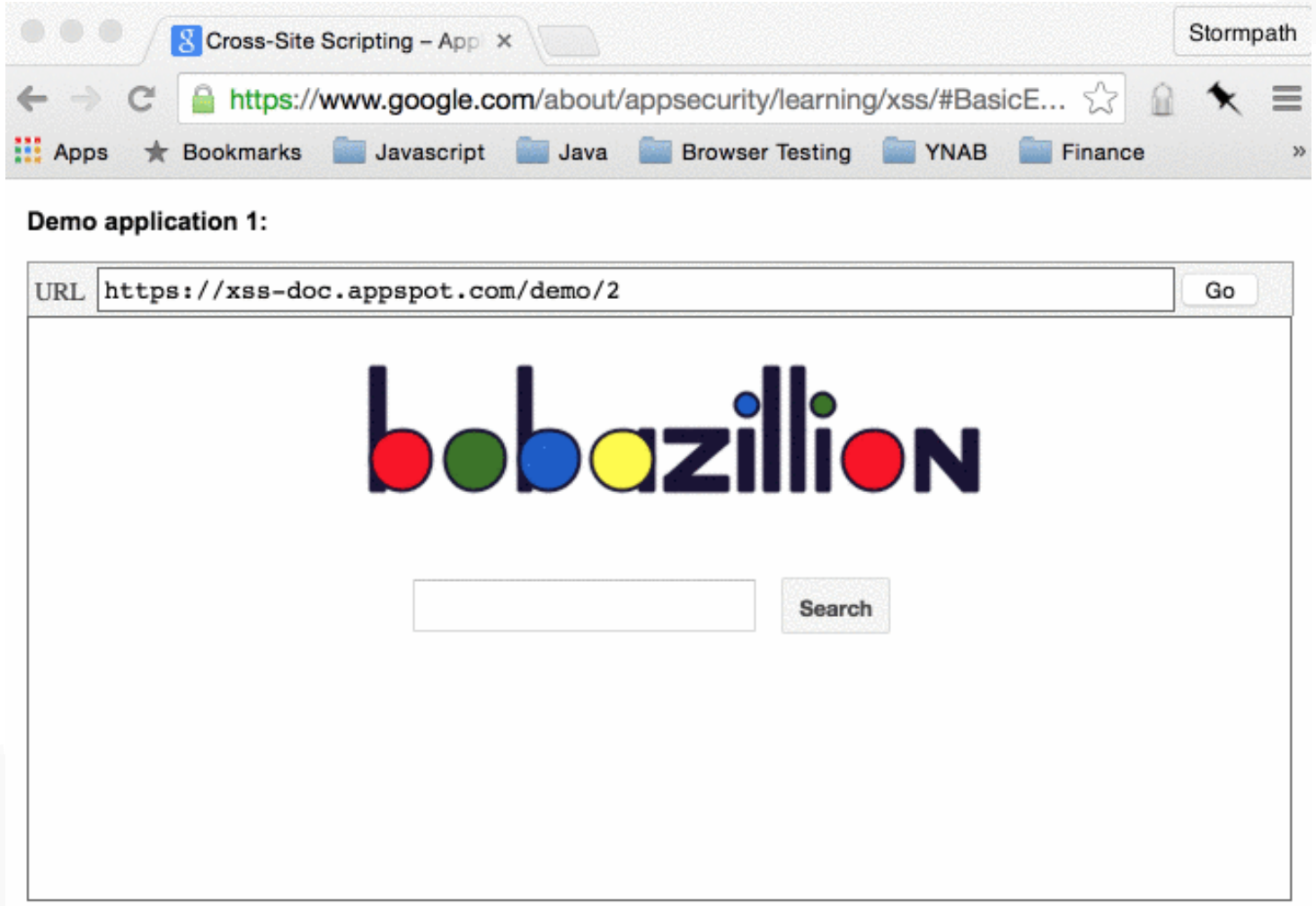
- Segurança API Rest.
- Autenticação API Rest.
- Exemplo de autenticação com JWT.
- CORS.
- Exemplo de CORS.

API REST – Segurança

APIs REST expostas na internet devem ter um cuidado maior com a segurança.

- Métodos HTTP devem ser protegidos:
  - Nem todos usuários podem acessar todos os métodos, deve-se verificar se quem está utilizando tem permissão.
- Proteger contra Cross-site Request Forgery (CSRF):
  - Enviar token em todas as requisições e validar o proprietário.
- Proteger contra Cross-site Scripting (XSS):
  - Validar entradas e saídas de dados do sistema.
  - Utilizar no DOM value, innerText e textContent ao invés de innerHTML.

API REST – Segurança



API REST – Segurança

- Não permitir acesso direto a partir da chave da entidade:
  - Exemplo: https://exemplo.com/pedido/2362365
- Limitar quantidade de requisições por dia/hora.
- Utilizar SSL/TLS para comunicação com o HTTPS para evitar o ataque do tipo Man in the Middle.
  - A não ser que o site seja somente de consulta.
- Responder as requisições com os status HTTP corretos.
  - Facilitar a validação das requisições e identificação de riscos em potencial.

API REST – Segurança

- **200 OK**: ação executada com sucesso.
- **400 Bad Request**: a requisição está mal formada, como por exemplo o formato do body incorreto.
- **401 Unauthorized**: autenticação inválida ou não informada.
- **403 Forbidden**: a autenticação foi realizada porém o usuário não tem permissão de acesso ao recurso.

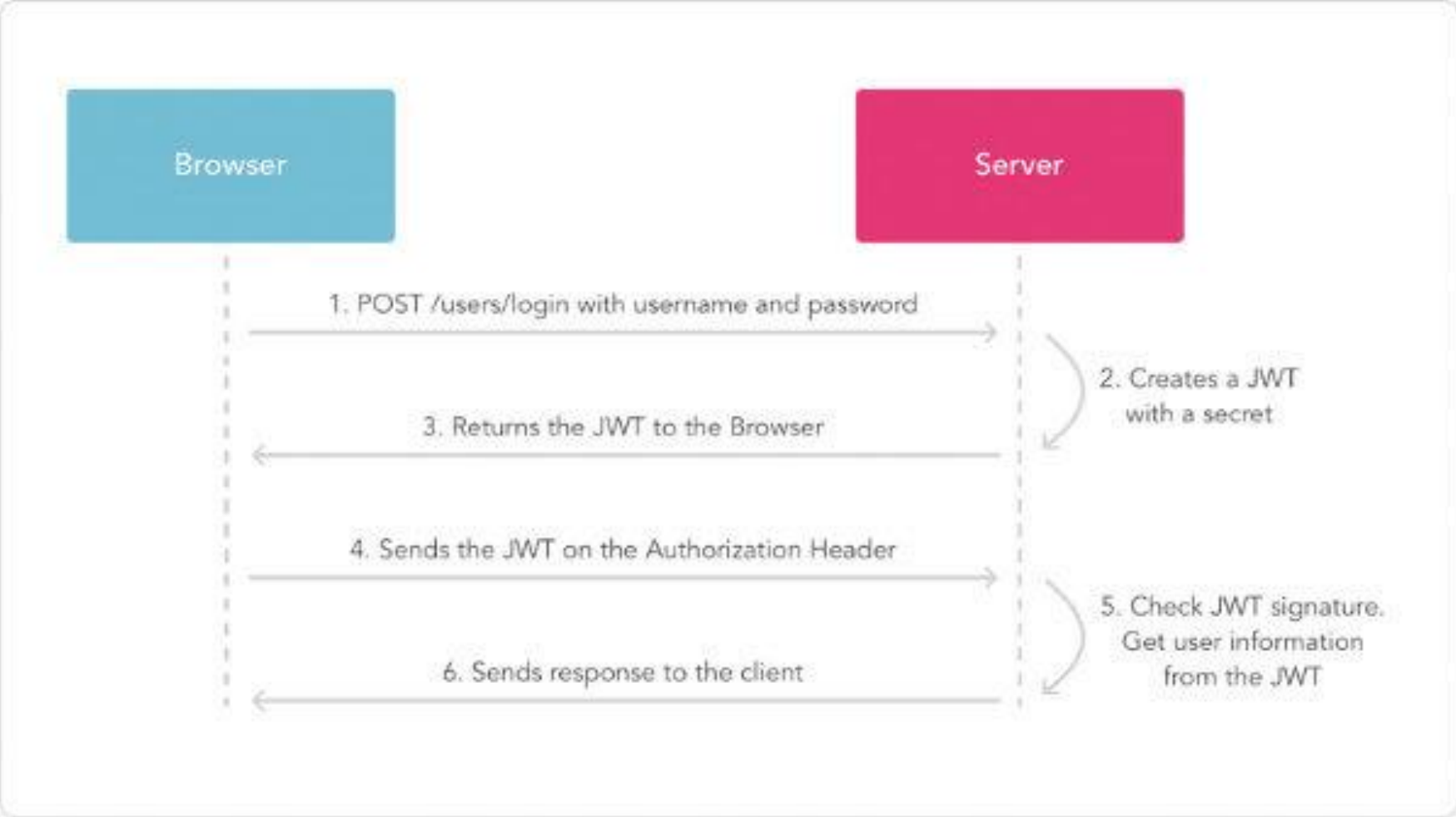
API REST – Segurança

- **404 Not Found:** recurso solicitado não existe.
- **405 Method Not Allowed:** método inválido. Exemplo: esperava-se POST e obteve GET.
- **429 Too Many Requests:** erro reportado quando um ataque DOS é verificado ou quando a quantidade de requisições foi limitada.

API REST – Autenticação – Session ID

- **Vantagens:**
  - Fácil de implementar no cliente e no servidor.
  - Fácil de destruir uma sessão quando um usuário faz logout.
- **Desvantagens:**
  - O servidor tem que limpar periodicamente as sessões que não foram finalizadas.
  - Toda requisição HTTP exige uma consulta ao banco de dados.
  - Armazenamento cresce a medida que aumenta o número de sessões.
  - Caso existam vários servidores pode gerar um gargalo.

API REST – Autenticação – JWT



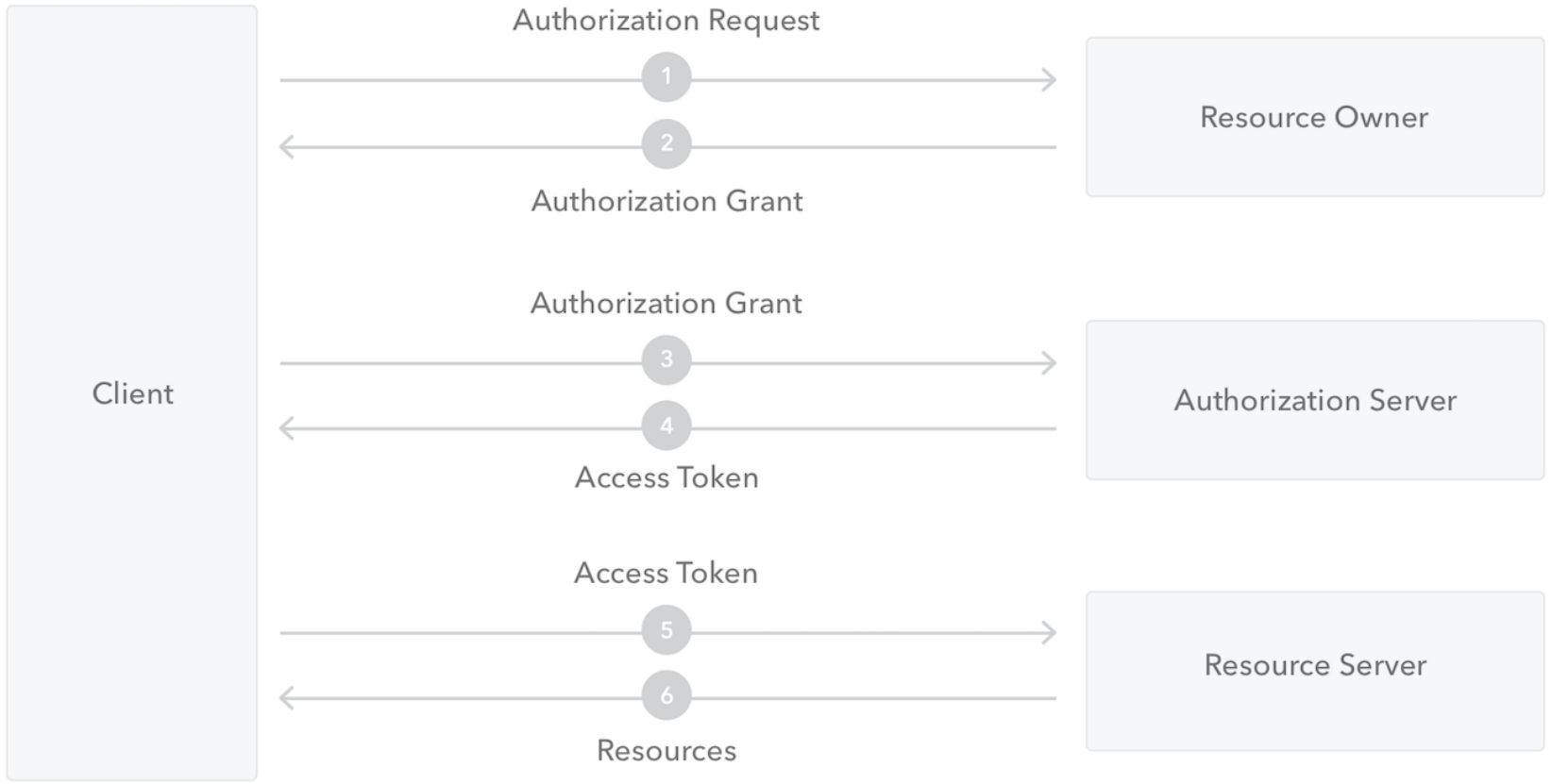
API REST – Autenticação – JWT

- **Vantagens:**
  - Não há armazenamento no servidor.
  - Código no cliente é simples.

API REST – Autenticação – JWT

- **Desvantagens:**
  - Tamanho do JWT pode ser maior que um session id, podendo afetar a performance.
  - Servidor precisa implementar a geração, validação e leitura do JWT.
  - Caso a chave de assinatura seja roubada, pode-se gerar JWTs sem que você tome conhecimento.
  - Para anular um JWT antes que ele expire é necessário ter uma lista de tokens ou alterar a chave para anular todos.

API REST – Autenticação – OAuth



API REST – Autenticação – OAuth

- **Vantagens:**
  - Não é preciso gerenciar cadastro e reset de senhas.
  - Não é preciso enviar link de confirmação de email.
  - Usuários não precisam relembrar seu usuário e senha.
  - A aplicação pode buscar dados do usuário sem que ele precise cadastrar.

API REST – Autenticação – OAuth

- **Desvantagens:**
  - Dependência de um terceiro para que usuários acessem a aplicação.
  - Problemas de privacidade, pois o terceiro conhece os usuários que utilizam sua aplicação.
  - Código escrito para cada provedor.
  - Confiança na implementação do terceiro. Ele pode enviar tokens trocados.

CORS

- **CORS:** Cross-origin Resource Sharing ou compartilhamento de recursos de origem cruzada.
- Política de segurança dos navegadores.
- Permite ao servidor definir como será o acesso a seus recursos a partir de outros domínios.
- Access-Control-Allow-Origin.

Conclusão

- ✓ Segurança API Rest.
- ✓ Autenticação API Rest.
- ✓ Exemplo de autenticação com JWT.
- ✓ CORS.
- ✓ Exemplo de CORS.