



Teste de Aplicações Web

Eduardo Boccato Pires de Camargo - 194286
Giovanne Lucas Dias Pereira Mariano - 173317
Natan Beltrão da Cunha Pevidor Carvalho - 184972



Características de aplicações Web

- **URL e estado:** Aplicações web são navegadas através de URLs e possuem a capacidade de retorno e avanço no navegador, o que permite um maior controle sobre o estado da aplicação pelo usuário.
- **Responsividade:** Uma página web pode ser acessada por diversos navegadores, dispositivos, sistemas operacionais e conexões de internet e precisa ser capaz de se adaptar às diferentes circunstâncias em que ela será acessada.
- **Tecnologias:** Aplicações Web são desenvolvidas usando diversas tecnologias. Enquanto a base do lado do cliente é composta por HTML, CSS e JavaScript, existem diversas outras ferramentas e frameworks tanto para o lado do cliente quanto para o lado do servidor, que adicionam à complexidade da arquitetura.
- **Rápida mudança:** Devido ao fato de que o deploy de novas versões é mais simples do que aplicações desktop ou aplicativos móveis, aplicações webs são alteradas com mais frequência.



Desafios para os testes

- **Estado da Navegação:** Difícil visualizar as mudanças antes e depois de eventos, pois a tela não renderiza completamente, a barra de endereço permanece a mesma.
- **Transição da Navegação:** Geralmente testado através de análise estática o que dificulta identificar erros em cenários complexos, que ocorrem em tempo de execução.
- **Mensagens com informação parcial:** Muitas das mensagens só fazem sentido após serem processadas pelo lado do cliente e inseridas na “DOM”.
- **Comportamento Assíncrono:** Gera um comportamento não determinístico na aplicação, que pode ser causado por atrasos na rede e tratamento não sequencial das mensagens.
- **Comportamento orientado a estado:** O maior desafio é salvar os estados da aplicação e permitir a navegação no histórico da aplicação, que geralmente quebra ao se utilizar requisições “AJAX”.



Técnicas para testes de aplicações Web

- **Testes de Caixa Branca:** O foco dos testes de caixa branca é testar o software com base na estrutura interna do programa.
- **Testes de Caixa Preta:** Os testes de caixa preta, por outro lado, focam na testagem baseada em funcionalidades, sendo agnóstica a implementações, de modo que a estrutura interna do código é irrelevante.



Técnicas de Caixa Branca

As principais abordagens para testes de caixa branca em Web são baseadas em modelos. Técnicas fortemente embasadas em engenharia reversa e raspagem de páginas web.

- **Baseada em navegação:** A abordagem baseada em navegação utiliza modelagem de grafos para gerar casos de teste que cubram as possibilidades de navegação. Nesta técnica, é possível cobrir um grande número de interações, mas não é possível testar comportamentos assíncronos ou analisar dinamicamente o comportamento da aplicação.
- **Baseado em cobertura de código:** Essa abordagem utiliza as técnicas de modelagem e testagem de fluxo de dados e fluxo de controle para garantir boa cobertura. É fortemente baseada em engenharia reversa.



Técnicas de Caixa Preta

- **FSMs:** A principal dificuldade encontrada é a modelagem do comportamento esperado da aplicação, de modo que uma das principais soluções para esse tipo de teste é a utilização de máquinas de estado finitas para modelagem de problema.
- **Sessões de usuários:** Uma segunda abordagem para testes de caixa preta em aplicações Web é explorar a geração de caminhos de teste através de sessões de usuários. Nesta prática, a utilização do sistema por usuários reais é captada e os usos mais frequentes são transformados em caminhos de teste, de modo que existe os testes tenham sempre boa cobertura com relação aos casos de uso mais frequentes.



Falhas comuns encontradas ou defeitos neste tipo de aplicação

- **URL inválida e páginas de erro:** Uma aplicação web precisa ser capaz de lidar com URLs não válidas da forma adequada.
- **Responsividade:** Uma página web pode não ser compatível com as configurações de um usuário (dispositivo, navegador, SO, etc.).
- **Cookies, variáveis de sessão e cache:** A gerência de estado no navegador possui várias camadas além das variáveis criadas no próprio código JavaScript, o que pode gerar inconsistências, que ocasionam em erros.



Bibliografia

ARORA, A.; SINHA, M. Web Application Testing: A Review on Techniques, Tools and State of Art. International Journal of Scientific & Engineering Research, v. 3, n. 2, 2012.