

## Abstrak

*Software testing* harus dapat dengan efektif menemukan cacat yang ada, dan banyak metode yang dapat digunakan untuk merancang *test-case*. Tidak hanya efektif, *software testing* juga harus dengan efisien melakukan *test* secepat dan semurah mungkin, untuk itu terdapat *tool* yang dapat meng-automasi *testing*.

*Software testing* yang ter-automasi dapat dengan signifikan mengurangi usaha yang dibutuhkan dalam *testing* yang menyeluruh, atau dengan signifikan meningkatkan jumlah *testing* yang dapat dilakukan dalam waktu yang terbatas. Terdapat beberapa *tool* untuk meng-automasi test khususnya aplikasi berbasis web yang dapat dijalankan dalam *browser*, salah satunya adalah Selenium.

Buku ini menghadirkan analisis untuk dua metode yang dapat diaplikasikan pada level *functional test* menggunakan *tool* Selenium, *Cause-Effect graphing* dan *State-Transition analysis*, mulai dari permasalahan implementasi dan ongkos uji.

Dari percobaan, walaupun masing-masing metode memiliki keunggulan untuk fungsionalitas tertentu, namun kedua metode tersebut dapat memodelkan spesifikasi yang sama tanpa kehilangan cakupan uji, sehingga tiap metode dapat mendeteksi semua cacat yang ada.

**Kata kunci:** *functional test, Selenium, Cause-Effect, State-Transition.*