

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

*Quality assurance* berpengaruh penting dalam pengembangan sebuah aplikasi. Adanya *software testing* (pengujian perangkat lunak) mampu menjadi bagian dalam *quality assurance*. *Software testing* merupakan hal yang dasar dan penting sebagai bagian dalam proses pengembangan *software*. Secara umum *software testing* dibagi menjadi dua kategori, yakni *Manual testing* dan *Automated software testing*. [1][2] dengan mengintegrasikan *testing* secara otomatis kedalam program pengembangan *software*, pengguna akan mendapatkan beberapa kelebihan pada penghematan biaya dan upaya serta juga mengurangi waktu.

Sebagai salah satu objek penelitian penulis, perusahaan Commsult Indonesia yang merupakan perusahaan yang berfokus pada pengembangan software, salah satu aplikasi yang dikembangkan sekarang, Ontego SAP Mobility merupakan aplikasi web mobile yang berfokus pada manajemen *warehouse*. Memberikan bantuan logistik dan jasa lewat mobile device. Aplikasi ini akan membantu pekerja dengan akses data secara daring maupun luring dengan akses SAP secara aman dan cepat. Proses implementasi logika ERP yang mencegah kesalahan *entry* dan mendukung *input form* secara majemuk. Salah satu fitur seperti *smart scan* untuk mempermudah *user experience*.

Secara garis besar, aplikasi Ontego SAP Mobility memiliki beberapa fitur penting seperti manajemen aplikasi, *Signing Key*, dan *Special Function* yang menunjang aplikasi dalam *warehouse management, sales and distribution*,

*production planning* , *extended warehouse management*, dan *material management* yang masing masing fitur tersebut perlu dilakukan sebuah pembuatan, edit, dan hapus yang biasa disebut sebagai *create,read,update,delete*.

Fitur *warehouse logistics* yang berfokus pada relokasi, pemesanan transportasi untuk mengangkut barang dan *quality control* untuk pengiriman, dengan fitur *OCR text recognition*. Fitur *production planning* yang merupakan bagian dari proses manufaktur dengan konfirmasi produksi, mengetahui kapan material akan dikirim dan dibuat. Fitur pada *inventory* yang memberikan kenyamanan pada kemampuan sistem untuk menyediakan *storage management*. Serta penggunaan aplikasi yang dapat digunakan pada *smartphone* atau tablet memberikan kemudahan akses untuk para pekerja perusahaan.

Proses perubahan yang terus terjadi, pengembangan fitur dan penambahan fitur baru yang dilakukan terus ,tuntutan *deadline* yang mengharuskan para pengembang menyelesaikan aplikasi dengan efektif dan efisien merupakan tantangan yang harus dihadapi. Aplikasi yang dibuat nantinya akan melewati tahapan pengujian terlebih dahulu sebelum diluncurkan dan dipakai oleh pengguna. Aplikasi yang baru dibuat tidak lepas dari beberapa kendala seperti *bug* dan *error*. Sehingga jika tidak ditangani dengan baik, maka akan menjadi masalah yang cukup fatal bagi aplikasi yang sedang dikembangkan.

Saat ini Commsult menggunakan prinsip kerja CI/CD (Continuous Integrations & Continuous Delivery) atau juga yang dikenal sebagai DevOps yang merupakan proses pengujian dan pembuatan software secara otomatis pada aplikasi yang telah terintegrasi pada *repository*, dilakukan proses penyampaian aplikasi dalam proses CI yang akan dilakukan secara terus menerus setiap adanya perubahan pada fitur aplikasi atau sistem, sehingga proses pembuatan aplikasi akan menjadi lebih cepat. Proses CICD ini secara umum membutuhkan pengujian secara otomatis untuk menangani siklus deploy, pengecekan kode dan menjalankan pengujian, sehingga manual test dinilai kurang relevan untuk digunakan pada prinsip CICD yang membutuhkan waktu yang cepat.[3]

Salah satu kendala yang terjadi pada perusahaan Commsult Indonesia adalah melakukan proses pengujian yang masih dilakukan secara manual dan tidak

sejalan dengan proses kerja CICD, yakni membutuhkan waktu yang cepat dan proses pengembangan yang terus menerus dan berulang - ulang. Jika dilakukan secara manual, proses *commit* kode tidak dilakukan secara terus menerus, sehingga akan lebih memakan waktu pada integrasi, mencari error, sehingga menunggu sampai proses *build* selesai sebelum ke proses pengujian. Sehingga dari segi waktu dan efisiensi sangat berperan penting bagi siklus pengembangan aplikasi secara berkelanjutan, dan adanya perubahan serta fitur baru yang membutuhkan waktu cepat pada pembuatan aplikasi perusahaan sampai dilakukannya proses *deploy*.

Sehingga untuk mengatasi permasalahan ini, penulis akan mengimplementasikan pengujian secara otomatis yang nantinya akan diaplikasikan oleh perusahaan untuk menjalankan metode CICD. Proses pengujian secara otomatis akan dilakukan pada aplikasi terkait, aplikasi yang dikembangkan nantinya akan selalu dalam proses pengembangan dan akan menghadapi beberapa perubahan seperti pemeliharaan(*maintenance*), dan *update*.

Proses pengujian ini akan dilakukan dengan menggunakan metode *black box* dimana pengujian ini berfokus pada fungsionalitas aplikasi dengan mengimplementasikan beberapa *test case* yang akan disesuaikan dengan fitur aplikasi ini. Dengan aplikasi yang sudah dapat dipakai oleh pengguna, maka data yang diterima atau digunakan juga akan bersifat masif, sehingga proses pengujian dengan set data yang banyak dengan variasi yang bermacam macam akan mengakibatkan pengujian menjadi tidak efektif. dua teknik yang akan digunakan dalam penelitian ini yaitu *Equivalence Partitioning*, dan *Boundary Value Analysis*.

Setiap perubahan dan pemeliharaan yang akan selalu dilakukan oleh perusahaan sehingga pembuatan *test case* yang digunakan untuk menguji aplikasi memiliki kemampuan untuk digunakan kembali. Tidak hanya fungsi *script* dan data, namun juga langkah pengujian. Maka dari itu, pengujian secara otomatis mampu dan penting untuk mencapai kasus pengujian yang *reusable*, *maintanable*, *accurate* dan mengurangi pekerjaan dalam pembuatan kasus

pengujian terhadap aplikasi yang dikembangkan, khususnya pada implementasi pengembangan aplikasi yang bersifat CICD

Penelitian ini akan mencakup pada fitur fitur aplikasi Ontego SAP Mobility dengan menguji fungsi dan fitur secara keseluruhan (end-to-end) dimana penulis akan meneliti pengujian secara otomatis pada aplikasi dengan menggunakan metode Black Box yang berfokus pada fungsionalitas sistem, dan teknik khusus input pada fitur, serta menganalisis perbandingan antara waktu uji manual dan otomatis untuk menganalisis kecepatan dan efisiensi waktu pengujian pada aplikasi.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dipaparkan di atas, rumusan masalah pada penelitian ini terdiri dari beberapa poin yaitu:

- 1.2.1. Apakah metode uji *Black Box* dengan teknik *Boundary Value Analysis* dan *Equivalence Partitioning* dapat mengcover seluruh kasus uji yang dibutuhkan?
- 1.2.2. Berapa waktu yang dibutuhkan untuk melakukan pengujian otomatis pada aplikasi *Ontego SAP Mobility* ?
- 1.2.3. Bagaimana perbandingan waktu uji otomatis dengan uji manual pada fitur aplikasi *Ontego SAP Mobility* agar sejalan dengan prinsip kerja CI/CD yang memerlukan waktu yang singkat?

## 1.3 Batasan Masalah

Batasan masalah dari penelitian ini adalah sebagai berikut:

- 1.3.1 Sistem yang dirancang akan berfokus pada software testing otomatis dengan menggunakan *tools* WebdriverIO
- 1.3.2 Objek pengujian adalah fitur dari aplikasi Ontego SAP Mobility

- 1.3.3 Sistem yang dijalankan akan melalui penulisan *script* dengan bahasa pemrograman JavaScript
- 1.3.4 Sistem pengujian akan menghasilkan laporan atau *report* menggunakan *spec reporter* dengan menunjukkan *test case*, *time*, dan , parameter keberhasilan atau kegagalan sebuah pengujian
- 1.3.5 Sistem pengujian yang dirancang akan menganalisis pengujian secara otomatis dari segi waktu yang dibutuhkan.
- 1.3.6 Eksekusi akan dijalankan melalui input *command* pada terminal untuk menjalankan pengujian secara otomatis
- 1.3.7 Sistem yang dirancang mencakup cara kerja WebdriverIO dalam menangani pengujian aplikasi website secara otomatis dengan memanfaatkan framework mocha.js

#### **1.4 Tujuan Penelitian**

- 1.4.1 Merancang test case untuk fitur-fitur pada aplikasi berbasis web dan melakukan pengujian fungsional secara otomatis
- 1.4.2 Mengetahui keberhasilan hasil pengujian fungsional dari *test case* yang dibuat dan diuji berdasarkan pengujian sistem dan uji *end-to-end*
- 1.4.3 Mengetahui hasil pengujian terutama pada parameter waktu yang dibutuhkan dalam melakukan pengujian pada setiap fitur bagian tertentu.
- 1.4.4 Mengetahui perbandingan waktu dan kecepatan dari pengujian manual dan pengujian otomatis
- 1.4.5 Mengetahui keefektifan waktu dari perbandingan hasil pengujian otomatis dengan pengujian manual
- 1.4.6 Menjadikan *script* pengujian otomatis untuk diaplikasikan pada Commsult Indonesia yang menggunakan prinsip CICD

## 1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

- 1.5.1 Mempermudah proses pengujian aplikasi berbasis web dengan memberikan pengetahuan terkait pengujian otomatis secara fungsional
- 1.5.2 Mengetahui perbandingan waktu pengujian yang efektif pada prinsip kerja tidak hanya khusus CICD , namun juga prinsip lainnya
- 1.5.3 Mempermudah dan mempersingkat kinerja menjadi lebih efektif dan efisien dalam waktu dan upaya
- 1.5.4 Mempelajari penggunaan *test automation framework* khususnya untuk aplikasi berbasis web
- 1.5.5 Mempelajari rancangan *test case* untuk pengujian secara otomatis