**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

**NAMA : Cahya Putra Hikmawan**

**NRP : 05111540000119**

**DOSEN WALI : Wijayanti Nurul Khotimah, S.Kom, M.Sc.**

**DOSEN PEMBIMBING : 1. Royyana Muslim I, S.Kom., M.Kom., Ph.D  
 2.**

# JUDUL TUGAS AKHIR

“Implementasi *Load Testing* dengan Menggunakan *Headless Browser* untuk Aplikasi di ITS.”

# LATAR BELAKANG

*Headless Testing* merupakan suatu teknik untuk melakukan pengujian tampilan aplikasi khususnya *Document Object Model (DOM)* pada *browser* tanpa menampilan *GUI browser,* pada umumnya fitur yang digunakan adalah *Headless Browser.* Teknik ini menyediakan akses kontrol yang sama seperti *browser* yang menggunakan *GUI,* namun yang membedakan adalah ekseskusi dijalankan melalui *Command Line Interface (CLI)*.

*Headless Browser* banyak digunakan *developer* untuk melakukan suatu pengujian tampilan, contohnya yaitu Headless Chrome, PhantomJS, Firefox Headless Mode, Selenium WebDriver.

Saat ini ITS mempunyai banyak aplikasi yang diluncurkan, namun belum ada pengujian tampilan yang menggunakan skenario pengujian sebelum diluncurkan. Oleh karena itu dibutuhkannya skenario pengujian agar aplikasi yang diluncurkan tidak bermasalah saat digunakan penggunanya dan tanpa harus melakukan pengujian manual satu-persatu. Untuk setiap pengujian diperlukannya *container* atau *virtual machine* sebagai *host* maupun *client* untuk menentukan jumlah *request* yang akan dijalankan untuk melakukan skenario pengujian.

# RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana cara melakukan pengujian tampilan untuk mendapatkan *Document Model Object (DOM)* tanpa menggunakan *GUI?*
2. Bagaimana cara membuat skenario untuk pengujian dengan menggunakan *container* atau *virtual machine?*

# BATASAN MASALAH

Permasalahan yang dibahas dalam tugas akhir ini memiliki batasan antara lain:

1. *Headless browser* yang digunakan adalah *headless chrome* dan *headless firefox mode.*
2. Kontainer yang digunakan adalah *docker*.
3. Uji coba aplikasi akan menggunakan API.

# TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan pembuatan tugas akhir ini antara lain:

1. Membuat sistem manajemen pengujian tampilan aplikasi khususnya *DOM* secara *headless*.
2. Mengimplementasikan pengujian menggunakan skenario yang sudah disiapkan sebelumnya.

# MANFAAT TUGAS AKHIR

Manfaat dari pembuatan tugas akhir ini adalah:

1. Mempelajari penggunaan Headless Browser untuk pengujian suatu aplikasi.
2. Mempermudah melakukan pengujian tampilan aplikasi secara *Headless* untuk mendapatkan *DOM*
3. Meminimalisir adanya *error* *loading* suatu aplikasi di ITS ketika sudah diluncurkan.

# TINJAUAN PUSTAKA

Dalam mengerjakan tugas akhir ini, terdapat beberapa tinjauan pustaka yang digunakan yaitu:

1. **Headless** **Browser**

*Headless Browser* [1] mengambil segala hak akses yang ada pada *Web Browser* normal, namun menghilangkan antarmuka pengguna. Hanya menyisakan mesin dan lingkungan *javascript*. Hal ini mengakibatkan penggunaan *Headless Browser* dilakukan menggunakan baris perintah atau *Command Line Interface*(*CLI)*.

1. **Puppeteer**

*Puppeteer* [2] adalah sebuah *Node Library* yang menyediakan *API* untuk mengontrol Chrome atau Chromium pada protokol pengembangan yang disediakan oleh Google untuk mengakses segala yang ada pada Chrome atau Chromium. Biasanya *puppeteer* berjalan secara *headless* tapi bisa juga dikonfigurasi menampilkan tampilan penuh untuk Chrome dan Chromium.

1. **Docker**

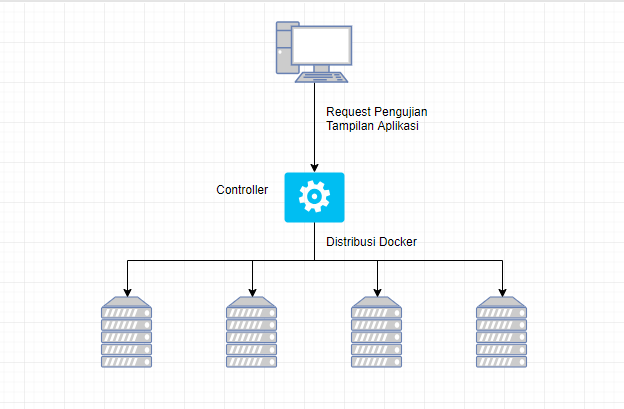
*Docker* [3] adalah sebuah *open* platform untuk membangun, is an open platform for developing, membungkus, dan menjalankan aplikasi. *Docker* memungkinkan untuk memisahkankan aplikasi dari infrastruktur supaya software dapat di jalankan dengan lebih cepat. Dengan *Docker*, dimungkinan untuk mengelola infrastruktur yang ada dengan cara yang sama untuk mengelola aplikasi. Dengan mengambil keuntungan dari metodologi *docker* untuk membungkus, melakukan tes dan melakukan *deploy* kode dengan cepat, *delay* antara penulisan kode dan menjalankannnya pada saat produksi dan dikurangi dengan signifikan.

# RINGKASAN ISI TUGAS AKHIR

Tugas akhir ini disusun untuk mengani masalah pengujian tampilan aplikasi di ITS yang menggunakan teknik secara *Headless* dengan cara membuat skenario yang akan dihubungkan oleh *controller* kepada salah satu node *docker*.

Proses pengujian akan diwalai adanya *request* dari *client* kemudian controller akan menerjemahkan apakah *client* membuat *request* pengujian secara manual atau menggunakan skenario, *request* akan dikirimkan ke docker untuk proses pengujian tampilan aplikasi menggunakan *Headless Browser.* Pengujian pada docker akan mengolah dokumen yang didapatkan dari website untuk dicari bagian yang masih mengalami kegagalan saat dimuat, hasil output bisa berupa screenshot untuk melihat hasil tampilan dan untuk javascript bisa dilihat dari hasil *DOM.*

Skema model dari sistem ini dapat dilihat seperti bagan dibawah ini:

 Gambar 1. Arsitektur

Setelah pengujian selesai maka aka nada hasil log untuk melihat apakah aplikasi masih memiliki bug atau error atau tidak.

# METODOLOGI

## Penyusunan proposal tugas akhir

Proposal tugas akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan *masalah* untuk tugas akhir, tujuan dari pembuatan tugas akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan tugas akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan tugas akhir mulai dari penyusunan proposal hingga penyusunan buku tugas akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan tugas akhir.

## Studi literatur

Pada studi literatur ini, akan dipelajari beberapa sejumlah referensi yang akan diperlukan untuk membangun aplikasi yaitu mengenai *Headless Browser* dan *Docker*.

## Analisis dan desain perangkat lunak

Aktor dari aplikasi ini adalah pengguna yang akan melakukan request untuk pengujian aplikasi pada website.

## Implementasi perangkat lunak

Aplikasi ini akan dibangun menggunakan docker sebagai container penyedia layanan yang akan dipasang di server dan Bahasa pemrograman Python.

## Pengujian dan evaluasi

Pengujian dalam aplikasi ini akan dilakukan dalam beberapa cara antara lain:

1. Pengujian *Request* dari Pengguna

Pengujian ini berfokus pada penerjemahan *request* dari pengguna agar bisa terhubung dengan *Headless Browser* untuk melakukan pengujian tampilan aplikasi.

1. Pengujian Dengan Skenario

Pengujian ini akan berfokus pada bagaimana pengujian tampilan aplikasi menggunakan skenario yang sudah disiapkan sebelumnya.

## Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
   1. Latar Belakang
   2. Rumusan Masalah
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

# JADWAL KEGIATAN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tahapan** | **2018** | | | | **2019** | | | | | | | | | | | | | | | | | | | | |
| **Desember** | | | | **Januari** | | | | **Februari** | | | | **Maret** | | | | **April** | | | | **Mei** | | | | |
| Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| Pengujian dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| Penyusunan Buku |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | R. Roemer, Backbone.js Testing, Birmingham B3 2PB, UK: Packt Publishing Ltd, 2013. |
| [2] | "Puppeteer," [Online]. Available: https://pptr.dev/. [Accessed 1 12 2018]. |
| [3] | "WHAT IS DOCKER?," [Online]. Available: https://www.docker.com/. [Accessed 1 12 2018]. |