

**LAPORAN TUGAS AKHIR**

***AUTOMATED TEST* MENGGUNAKAN FRISBY.JS UNTUK  
*TESTING REST API* ALIPAY DI PT. SEPULSA TEKNOLOGI  
INDONESIA**



**FERDINANDUS EKA P.S**

**201533006**

**PROGRAM STUDI D-III MANAJEMEN INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS KATOLIK WIDYA KARYA  
2018**

**LAPORAN TUGAS AKHIR**

***AUTOMATED TEST MENGGUNAKAN FRISBY.JS UNTUK  
TESTING REST API ALIPAY DI PT. SEPULSA TEKNOLOGI  
INDONESIA***



**FERDINANDUS EKA P.S**

**201533006**

**PROGRAM STUDI D-III MANAJEMEN INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS KATOLIK WIDYA KARYA**

**2018**

## LEMBAR PENGESAHAN

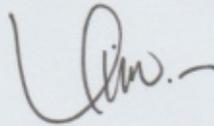
Nama : Ferdinandus Eka Putra Santosa  
NIM : 201533006  
Fakultas : Teknik  
Jurusan : D3 / Manajemen Informatika  
Judul : **AUTOMATED TEST MENGGUNAKAN FRISBY.JS UNTUK TESTING REST API ALIPAY DI PT. SEPULSA TEKNOLOGI INDONESIA**

Malang, 04 Juli 2018

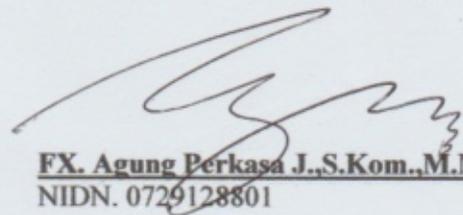
Disetujui oleh,

Dosen Pembimbing,

Dosen Penguji,



Vinno Christmantara, S.Kom., M.Kom.  
NIDN.0707068602



FX. Agung Perkasa J., S.Kom., M.M.  
NIDN. 0729128801

Mengetahui,

Dekan Fakultas Teknik,

Ketua Program DIII  
Manajemen Informatika,



## SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIASI

Saya yang bertanda tangan dibawah ini menerangkan dengan sesungguhnya bahwa Tugas Akhir yang berjudul "*Automated Test Menggunakan Frisby.JS Untuk Testing Rest API ALIPAY Di PT. Sepulsa Teknologi Indonesia*" merupakan karya tulis asli,

Nama : Ferdinandus Eka Putra Santosa

NIM : 201533006

Fakultas : Teknik

Jurusan : DIII Manajemen Informatika

Universitas : Universitas Katolik Widya Karya Malang

Dan bukan karya plagiat, baik secara sebagian maupun seluruhnya.

Demikian surat keterangan ini saya buat dengan sebenar-benarnya dan apabila terdapat kekeliruan, saya bersedia menerima sanksi sesuai dengan aturan yang berlaku.

Malang, 6 Juli 2018



Ferdinandus Eka Putra Santosa

## KATA PENGANTAR

Puji dan syukur penulis ucapkan kepada Tuhan yang Maha Esa yang telah memberikan rahmat dan karunianya berupa kesehatan dan kesempatan, sehingga penulis mampu menyelesaikan Laporan Kerja Lapangan ini.

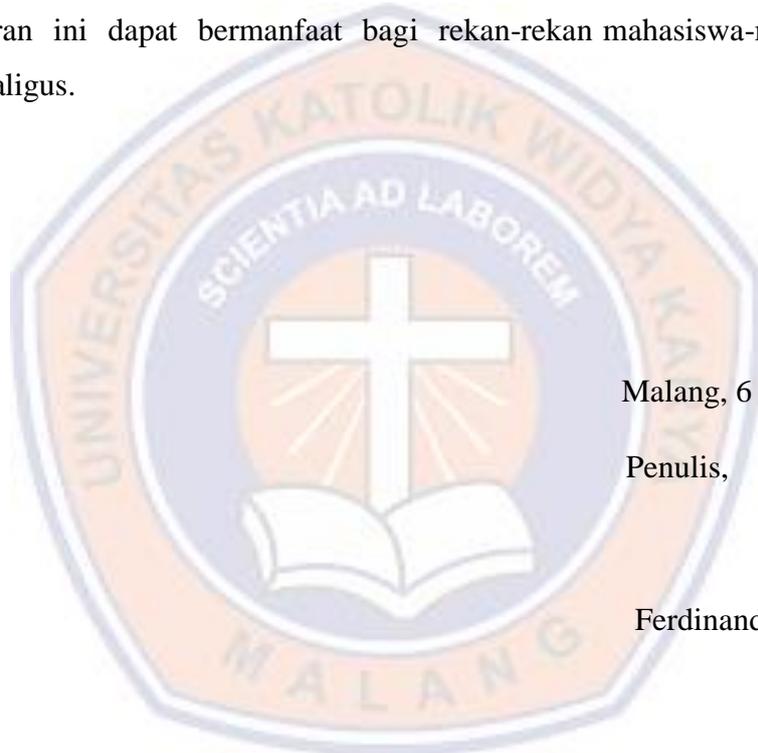
Laporan tugas akhir ini berjudul “*Automated Test Untuk Testing REST API Di PT.Sepulsa Teknologi Indonesia*”. Laporan tugas akhir ini merupakan tugas yang harus diselesaikan oleh Mahasiswa Jurusan D3 Sistem Informasi di Universitas Katolik Widya Karya Malang. Tujuan utama dari tugas akhir ini adalah untuk memantapkan teori dan praktek yang telah dipelajari di kampus dan dapat diselesaikan dengan serta diaplikasikan di lapangan. Laporan ini disusun guna melengkapi prasyarat dalam menyelesaikan kegiatan tugas akhir bagi mahasiswa Fakultas Teknik Program Diploma III Sistem Informasi dalam meningkatkan peran serta mahasiswa pada dunia kerja.

Dalam penyusunan laporan, penulis banyak mendapatkan bimbingan dan dukungan dari berbagai pihak, maka pada kesempatan ini penulis menyampaikan rasa terimakasih yang ditujukan kepada :

1. Bapak Benedictus Sonny Y, S.Pd., M.T. selaku Dekan Fakultas Teknik Universitas Katolik Widya Karya Malang.
2. Ibu Lianita Febrihani, S.Kom., M.T. selaku Ketua Jurusan D3 Sistem Informasi Universitas Katolik Widya Karya Malang.
3. Bapak Vinno Chrismantara, S.Kom., M.Kom. selaku Dosen Pembimbing yang telah meluangkan waktunya untuk membimbing, mengarahkan, memberi semangat dan dorongan kepada penulis dalam penyusunan laporan hingga selesai.
4. Seluruh Dosen Pengajar D3 Sistem Informasi yang telah memberikan ilmu yang bermanfaat bagi penulis.
5. Keluarga tercinta yang telah memberikan semangat, dorongan serta doa yang bermanfaat bagi penulis.

6. Teman – teman angkatan 2015 yang banyak membantu dalam proses penyusunan laporan tugas akhir sampai selesai.
7. Semua pihak yang banyak membantu terselesainya laporan tugas akhir yang tidak disebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan dengan segala kekurangannya. Untuk itu penulis mengharapkan adanya kritik dan saran dari semua pihak demi kesempurnaan dari laporan tugas akhir ini. Penulis berharap, semoga laporan ini dapat bermanfaat bagi rekan-rekan mahasiswa-mahasiswi dan pembaca sekaligus.



Malang, 6 Novembe 2017

Penulis,

Ferdinandus Eka P.S.

**DAFTAR ISI**

|  | Halaman     |
|--|-------------|
| <b>Halaman judul .....</b>               | <b>ii</b>   |
| <b>LEMBAR PENGESAHAN .....</b>           | <b>iii</b>  |
| <b>KATA PENGANTAR .....</b>              | <b>iv</b>   |
| <b>DAFTAR ISI .....</b>                  | <b>vi</b>   |
| <b>DAFTAR TABEL .....</b>                | <b>vii</b>  |
| <b>DAFTAR GAMBAR .....</b>               | <b>viii</b> |
| <b>SEGMENT KODE .....</b>                | <b>ix</b>   |
| <b>BAB I .....</b>                       | <b>1</b>    |
| 1.1 Latar Belakang .....                 | 1           |
| 1.2 Rumusan masalah .....                | 3           |
| 1.3 Tujuan .....                         | 3           |
| 1.4 Manfaat .....                        | 3           |
| 1.5 Batasan Masalah .....                | 4           |
| 1.6 Metode Pengambilan Data.....         | 4           |
| <b>BAB II.....</b>                       | <b>5</b>    |
| 2.1 Gambaran Situasi .....               | 5           |
| 2.2 Tinjauan Pustaka.....                | 7           |
| 2.3 Penelitian Sebelumnya.....           | 12          |
| <b>BAB III .....</b>                     | <b>15</b>   |
| 3.1 Analisa Sistem .....                 | 15          |
| 3.2 Perancangan Sistem .....             | 18          |
| <b>BAB IV.....</b>                       | <b>36</b>   |
| 4.1 Perangkat Lunak yang dibutuhkan..... | 36          |
| 4.2 Implementasi.....                    | 37          |
| <b>BAB V .....</b>                       | <b>51</b>   |
| 5.1 Kesimpulan .....                     | 51          |
| 5.2 Saran .....                          | 51          |
| <b>DAFTAR PUSTAKA .....</b>              | <b>x</b>    |

**DAFTAR TABEL**

|  | Halaman |
|--|---------|
| Tabel 3.1 Data hasil manual test.....                              | 15      |
| Tabel 3.2 Data hasil automated test.....                           | 16      |
| Tabel 4.1 Data hasil automated test tanpa bantuan runner.sh .....  | 49      |
| Tabel 4.2 Dara hasil automated test dengan bantuan runner.sh ..... | 48      |



**DAFTAR GAMBAR**

|  | Halaman |
|--|---------|
| Gambar 3.1 Hasil dari test frisby.js .....                     | 17      |
| Gambar 3.2 Sequence Diagram dari sistem API.....               | 19      |
| Gambar 4. 1 menjalankan runner.sh melalui terminal.....        | 37      |
| Gambar 4. 2 Tampilan Menu Informasi .....                      | 39      |
| Gambar 4. 3 Tampilan konfirmasi Exit Game .....                | 40      |
| Gambar 4. 7 Tampilan hasil report istanbul .....               | 46      |
| Gambar 4. 8 Tampilan jika memilih salah satu file test .....   | 47      |
| Gambar 4. 9 Tampilan detail dari hasil code coverage test..... | 48      |



**SEGMENT KODE**

|   | Halaman |
|---|---------|
| Segment Kode 4. 1 Kode runner.sh .....                              | 38      |
| Segment Kode 4. 2 fungsi konversi file report .....                 | 42      |
| Segment Kode 4. 3 fungsi gulp untuk menjalankan automated test..... | 43      |
| Segment Kode 4. 4 Inisialisasi konfigurasi.....                     | 45      |
| Segment Kode 4. 5 fungsi untuk menjalankan eslint .....             | 45      |
| Segment Kode 4. 6 konversi hasil ke dalam format html.....          | 47      |



## BAB I PENDAHULUAN

### 1.1 Latar Belakang

*Software testing* adalah upaya pengecekan yang dilakukan untuk memberikan informasi tentang kualitas dari sistem yang sedang berjalan maupun sedang dirancang. Dari hasil *software testing* para *developer* maupun *product manager* dapat mengetahui apakah resiko yang akan didapat jika sistem tersebut diimplementasikan. Langkah – langkah yang dilakukan ketika melakukan *software testing* adalah dengan menjalankan sistem lalu mencari *bug* (*error* dan masalah lain), dan mengecek apakah sistem layak untuk diimplementasikan.

Dalam *software testing* ada beberapa hal yang diinginkan seperti, apakah sistem sudah memenuhi kebutuhan spesifikasi yang ada, apakah sistem sudah sesuai dengan keinginan dari *product manager*. Dalam melakukan *software testing* dapat dilakukan menggunakan dua cara yaitu dengan *manual testing* dan *automated testing*. *Manual testing* Dalam melakukan test diawasi oleh manusia, karena biasanya *assertion*-nya masih menggunakan bagian mata dan pikiran manusia, *manual testing* merupakan Jenis tes yang paling umum dilakukan. Awalnya instruksi test-case ini dilakukan secara manual oleh tester tersebut (dengan mouse/keyboard). Sedangkan *automated test* adalah meng-otomasi *test case* yang tadinya dikerjakan secara manual dan berulang-ulang. Yang tadinya harus dan diimput dan -klik satu-satu jadi cukup klik sekali. Tujuan utama dari *automated testing* adalah untuk meningkatkan efektivitas, efisiensi, dan *coverage* dalam testing.

Dalam pembuatan sistem yang baik, pasti diperlukan suatu proses uji coba dalam memastikan bahwa sistem tersebut berjalan dengan baik. Uji coba sistem

dapat dilakukan dengan manual testing yaitu melakukan input dan running sistem secara manual. Hal seperti ini biasanya dilakukan hanya sekali atau dua kali pengulangan, Namun ketika sistem yang ada sudah sangat kompleks dan *workflow developer* sistem sudah sangat tinggi, maka semakin banyak resiko sistem tersebut memiliki *error* maupun *bug* yang tidak disadari. Jika proses pengecekan menggunakan *manual test* maka akan membutuhkan waktu yang lama dan tidak efisien.

Untuk itu diperlukan adanya automated test yang dapat bermanfaat dalam meningkatkan efektivitas dan efisiensi waktu dalam pengembangan sebuah sistem. Berbeda dengan *manual test*, dalam *automated test* kita perlu menulis *script*. *Automated test* sebenarnya menjalankan test case sama persis dengan yang dilakukan oleh *manual tester* Hanya saja, semua proses dan validasi yang berkaitan dilakukan oleh komputer. Biasanya *script automation* ini dibuat berdasarkan *test case* yang ada. 1 *script* bisa dibuat untuk 1 *test case* saja atau banyak *test case* sekaligus, tergantung kebutuhannya. *Script* ini nantinya bisa dijalankan lagi kapanpun, berapa kali pun kita mau. Bahkan bisa di-*manage* dan di-*schedule* bila perlu. Selanjutnya, kita hanya perlu menunggu dan melihat apakah hasil Test tersebut *passed/failed*.

PT. Sepulsa Teknologi Indonesia merupakan perusahaan yang bergerak di bidang teknologi, yang memiliki *workflow* yang sangat padat. Sistem payment melalui alipay adalah fitur baru yang akan diimplementasikan pada sistem mereka, karena sistem yang berjalan sudah sangat kompleks, maka kemungkinan untuk adanya bug dan error akan semakin tinggi.

Untuk itu diperlukan adanya automated test dengan menggunakan *frisby.js*, yang merupakan salah satu framework yang bisa digunakan untuk melakukan automated test, karena *frisby.js* dianggap menghasilkan report yang lebih baik dari framework-framework lain. Dengan dibuatnya automated test ini memang dapat mempermudah dan memperlancar implementasi sistem yang sedang berjalan, serta

dapat menghemat waktu sumber daya. Namun *automated test* yang sudah dibuat dapat dikembangkan lagi supaya menjadi lebih efisien, baik itu dari segi waktu dan *cost* dan *effort* yang dikeluarkan serta mampu memberikan *report/outcome* yang dapat bermanfaat untuk manajemen sistem yang sedang berjalan maupun sedang diimplementasikan.

## 1.2 Rumusan masalah

Berdasarkan latar belakang di atas, rumusan masalah yang dapat diidentifikasi adalah sebagai berikut :

1. Bagaimana cara membuat *automated test* menggunakan *frisby.js* untuk testing REST API Alipay di PT. Sepulsa Teknologi Indonesia dengan metode *runner.js*?
2. Bagaimana cara implementasi *automated test* pada sistem REST API Alipay yang sedang berjalan di PT. Sepulsa Teknologi Indonesia?

## 1.3 Tujuan

Adapun tujuan dari pemilihan topik penelitian ini sebagai Tugas Akhir, antara lain :

1. Mengetahui cara membuat *automated test* yang sudah ada menjadi lebih efisien.
2. Mampu mengimplementasikan *automated test* yang sudah lebih efisien pada sistem sedang berjalan.
3. Dapat membuat *automated test* yang sudah ada menjadi lebih efisien.

## 1.4 Manfaat

Manfaat dari penelitian dalam tugas akhir yakni :

1. Dapat lebih cepat dan mudah dalam menemukan error dalam sistem yang sedang berjalan, maupun yang masih dalam tahap percobaan.
2. Dengan *automated test* developer dapat menghemat waktu dan terbantu dalam mengembangkan sistem tersebut.
3. Mempersingkat waktu *tester* dalam melakukan *testing* dan dapat lebih mudah dalam melakukan *reporting*.

4. Hasil dari automated test dapat digunakan perusahaan sebagai bahan penentu keputusan.
5. Penelitian ini dapat bermanfaat sebagai acuan bagi peneliti lain yang akan melakukan pengembangan ataupun pembuatan penelitian yang sama.

## 1.5 Batasan Masalah

Adapun batasan-batasan masalah pada laporan ini sebagai berikut :

1. Yang dibahas hanya metode automated test tidak sampai pada pembuatan sistem Alipay.
2. Tidak membahas tentang manual test secara terperinci lebih kepada bagaimana menyusun automated test dan bagaimana cara kerjanya.
3. Automated test yang disusun hanya sampai pada menghasilkan report.

## 1.6 Metode Pengambilan Data

Untuk mendapatkan data-data yang akurat, penulis meninjau dan mengamati kegiatan dengan beberapa metode pengambilan data sebagai berikut :

### 1. Metode Observasi dan Wawancara

Metode observasi adalah salah satu metode pengumpulan data dengan cara melakukan pengamatan secara langsung yang meliputi kegiatan pemusatan perhatian terhadap suatu objek dengan menggunakan seluruh alat indera. Dalam hal ini, penulis mengamati kegiatan dan para *software engineer* serta QA, alur kerja, dan masalah atau kendala apa yang dihadapi dalam membangun sebuah sistem. Metode wawancara adalah salah satu metode pengumpulan data dengan cara melakukan pertanyaan baik terstruktur maupun tidak terstruktur. Dalam hal ini penulis melakukan pengumpulan data tentang bagaimana sistem yang berjalan dan mengenai sistem yang sedang penulis kerjakan, serta bagaimana membuat automated test yang baik dan efisien.

### 2. Studi Pustaka

Merupakan proses pengambilan data dengan cara mendapatkan informasi dari berbagai sumber seperti buku, karya ilmiah, laporan penelitian dan sumber lainnya.