**Nama Projek :** https://shopee.co.id/ **(Frontend)**

1. **Gambaran**

 Sebagai bagian dari projek, maka pada projek ini diminta untuk menguji beberapa fungsi dari aplikasi <https://shopee.co.id/>.

 Tujuan penulisan dokumen ini adalah sebagai perancangan pengujian tingkat tinggi dengan rincian

* + Lingkup Proyek
  + Strategi Pengujian
  + Jadwal Pengujian, dan
  + *Resource requirements*
  + *test deliverables* dan *schedule*

## Scope

Ruang lingkup projek “https://shopee.co.id/” mencakup pengujian fitur – fitur berikut

* Register
* Login & Logout
* Forgot Password
* Search
* Product Compare
* Product Display Page
* Add to Cart
* Wish List
* Shopping Cart
* Currencies
* Home Page
* Checkout Page
* My Account Page
* Order History Page
* Downloads Page
* Contact Us Page
* Menu Options
* Footer Options
* Category Pages

## Test Environment

* *Windows 10 – Chrome, Firefox and Edge*

# Pengecualian

* Semua fitur kecuali yang telah disbutkan pada bagian 2
* *Test Automation*

## Test Strategy

Sebagai bagian dari pengujian fungsional , saya akan mengikuti pendekatan pengujian berikut ini

# Langkah pertama:

Pembuatan scenario uji dan Kasus uji untuk berbagai fitur pada *Scope*

* + Saya akan menerapkan beberapa tehnik desain tes saat membuat kasus uji
    - *Equivalence Class Partition*
    - *Boundary Value Analysis*
    - *Decision Table Testing*
    - *State Transition Testing*
    - *Use Case Testing*
  + Saya juga menerapkan keahlian dalam membuat kasus uji dengan menerapkan
    - *Error Guessing*
    - *Exploratory Testing*
  + Saya memprioritaskan kasus uji

# Langkah kedua:

Proses pengujian saya,

* + Pertama, kami akan melakukan ***smoke testing*** untuk memeriksa apakah perbedaan dan fungsi penting dari aplikasi bekerja.
  + Saya akan menolak *build,* jika smoke testing gagal dan akan menunggu *stable* sebelum melakukan pengujian mendalam dari fungsionalitas aplikasi.
  + Setelah menerima *build* yang stabil, yang lulus smoke testing akan dilakukan pengujian sedara mendalam menggunakan ***test case*** yang dibuat.
  + Beberapa *test resources* akan menguji aplikasi yang sama dibeberapa lingkungan yang didukung secara bersamaan.
  + Saya kemudian akan melaporkan ***bug*** di alat pelacak *bug* dan megirimkan dev.*management* jika cacat ditemukan pada saat itu
  + Sebagai bagian dari pengujian maka saya akan melakukan jenis pengujian
* *Smoke Testing and Sanity Testing*
* *Regression Testing and Retesting*
* *Usability Testing, Functionality & UI Testing*
  + Saya akan mengulangi siklus uji coba sampai mendapatkan produk yang berkualitas

# Langkah ketiga:

Saya akan mengikuti praktik terbaik berikut ini untuk menjadikan Pengujian yang lebih baik:

* + *Context Driving Testing,* Saya akan melakukan pengujian sesuai dengan konteksnya dari aplikasi yang diberikan
  + *Shift Left Testing,* Saya akan memulai pengujian tersebut dari tahap awal pengembangan itu sendiri alih – alih menunggu *stable build.*
  + *Exploratory Testing,* Saya akan melakukan eksploratory pengujian , selain dari mengeksekusi kasus uji normal
  + *End to end flow testing,* Saya akan menguji scenario end to end yang melibatkan beberapa fungsi untuk mensimulasikan *end users flow.*

# Prosedur Pelaporan Cacat

Selama eksekusi tes

* + Setiap penyimpangan dari perilaku yang diharapkan oleh aplikasi akan dicatat, jika hal tersebut tidak dapat dilaporkan sebagai cacat, maka akan dilaporkan sebagai pengamatan/masalah atau diajukan sebagai pertanyaan
  + Setiap masalah kegunaan juga akan dilaporkan
  + Setelah ditemukan cacat, selanjutnya akan diuji ulang untuk memverifikasi reproduktifitas dari cacat. Tangkapan layar dengan langkah-langkah mereproduksi akan didokumentasikan.
  + Diakhir pelaksanaan tes, cacat yang ditemui akan dikirim bersama dengan observasi.

\*\* Catatan : -- Cacat akan didokumentasikan di excel

: -- Skenario uji dan Kasus uji akan didokumentasikan dalam dokumen excel

## Responsibilities

* Escalations
* Create the Test Plan and get the client signoffs
* Interact with the application, create and execute the test cases
* Report defects
* Coordinate the test execution. Verify validity of the defects being reported.
* Interact with the application
* Create and Execute the Test cases.
* Report defects

# Jadwal Tes

Jadwal tes yang direncanakan adalah sebagai berikut

|  |  |
| --- | --- |
| Task | Time Duration |
| *Creating Test Plan* | *Start Date to End Date* |
| *Test case creation* | *Start Date to End Date* |
| *Test Case Execution* | *Start Date to End Date* |
| *Summary Reports Submission* | *Date* |

# Hasil Uji

Berikut ini harus disampaikan kepada klien:

|  |  |  |
| --- | --- | --- |
| Pengujian | Deskripsi | Tanggal penyelesaian target |
| Test Plan | Details on the scope of the project, test strategy, test schedule, resource requirements, test deliverables and  schedule | Date |

|  |  |  |
| --- | --- | --- |
| Functional test  cases | Test cases created for the scope defined | Date |
| Defect reports | Detailed description of the defects indentified along with screenshots and  steps to reproduce on a daily basis | NA |
| Summary reports | Summary reports— Bugs by bug#,  Bugs by functional area and  Bugs by priority | Date |

# Pricing

NA

# Entry and Exit Criteria

Berikut ii adalah kriteris masuk dan keluar untuk setiap fase software testing life cycle:

# Requirement Analysis

Kriteria masukan : Setelah tim penguji menerima dokumen persyaratan atau detail tentang proyek

Kriteria keluaran : Daftar persyaratan dieksplorasi dan dipahami oleh tim penguji, menghilangkan keraguan

# Test Planning

Kriteria masukan : Persyaratan yang dapat diuji berasal dari dokumen persyaratan yang diberikan atau detail proyek, menghilangkan keraguan

Kriteria keluaran : Dokumen rencana uji (termasuk strategi uji) ditandatangani oleh klien

# Test Designing

Kriteria masukan : Dokumen rencana uji ditandatangani oleh klien

Kriteria keluaran : Dokumen test scenario dan test cases ditandatangani oleh klien

# Test Execution

Kriteria masukan : Dokumen test scenario dan test cases ditandatangani oleh klien, aplikasi siap diuji

Kriteria keluaran : Laporan kasus uji, laporan cacat telah siap

# Test Closure

Kriteria masukan : Test case reports, Laporan cacat telah selesai Kriteria keluaran : Laporan ringkasan tes

# Tools

Berikut adalah daftar alat yang saya gunakan dalam mengerjakan proyek ini:

* + XYZ bug tracking tool
  + Mind map tool
  + Snipping screenshot tool
  + Word and excel documents

# Risiko dan Mitigasi

Berikut adalah daftar risiko yang mungkin terjadi dan cara untuk menguranginya Risk : Tidak tersedianya sumber daya

Mitigasi: Perencanaan sumber daya cadangan Risk : Build URL is not working

Mitigasi: Sumber daya tidak akan bekerja pada task lainnya Risk : Lebih sedikit waktu untuk pengujian

Mitigasi: Meningkatkan sumber daya berdasarkan kebutuhan klien secara dinamis

# Persetujuan

Saya akan mengirimkan berbagai jenis dokumen untuk persetujuan seperti,

* + Test Plan
  + Test Scenarios
  + Test Cases
  + Reports