

## LAMPIRAN

### A. TATA CARA PENGGUNAAN APLIKASI *TESTING*

#### 1. *Onboarding*

Tampilan awal ketika *User* berhasil menginstall aplikasi *mobile Testing*. Selanjutnya akan masuk pada tampilan *onboarding screen* dari aplikasi *testing*, dimana terdapat 3 *page*. Ketika *user* menekan tombol '*Get started*', maka akan masuk pada halaman *login*.

#### 2. Registrasi

Jika *user* belum memiliki akun, maka dapat melakukan Registrasi terlebih dahulu melalui tombol *Register* yang terdapat dibawah *button Login*. Kemudian akan muncul tampilan *form* untuk melakukan registrasi, *user* dimohon untuk mengisi semua *form*.

Dengan keterangan:

<i>Email</i>	inputan wajib <i>email</i> menggunakan <i>email</i> valid
<i>Password</i>	wajib mengisi <i>password</i> minimal 8 karakter
<i>Repeat Password</i>	wajib mengisi <i>confirm password</i> sama dengan data <i>password</i>

Setelah *user* klik *register* maka akan menerima *email* dari server akan mengaktifkan *account* dengan membuka *email* dari aplikasi *testing* dan klik *Verify Email*.

#### 3. *Login*

Setelah melakukan registrasi maka *User* akan diarahkan ke halaman *login* untuk melakukan *login*, data yang diperlukan adalah *email* dan *password*, *User* diharapkan mengisi *email* dan *password* yang sudah didaftarkan sebelumnya untuk autentikasi keamanan.

#### 4. *Home Page*

*Home page* dari aplikasi testing yang terbagi menjadi beberapa fitur yang masing-masing menu pada *home* mempunyai fungsi dan tampilan yang berbeda-beda, yang akan dijelaskan pada poin-poin berikutnya.

- **Nama User**

*User* akan disapa sesuai sama nama *User*

- **Level achievement**

Level *achievement* atau pencapaian *user*, didapatkan sesuai seberapa banyak *user* sudah menambahkan *result* di *Scenario*

- **Search Project**

Fitur *search* ini berguna untuk mencari nama *project*

- **Create**

*Button create project* digunakan untuk menambahkan *project* baru

- **Report**

Digunakan untuk mengetahui seberapa banyak *scenario* yang sudah di *testing*.

- **Notification**

Digunakan untuk mengetahui pesan yang muncul ketika ada *member* baru yang *join* ke *project* kita dan kita sendiri masuk sebagai *role* apa.

- **Account**

Digunakan *user* untuk mengelola informasi dari akun dan terdapat menu *Account Setting, Achievement, about* dan *Log out*.

#### 5. *Create Project*

*User* dapat membuat *project* baru untuk mengelola dan melacak pengujian sesuai dengan data yang dibuat. *User* dapat membuat dengan cara klik *button Create project* pada navigasi bawah, kemudian akan muncul *form* pengisian *project*. Setelah *user* mengisi semua *form* maka *user* klik tombol *create* maka *project* baru berhasil ditambahkan. Setelah berhasil membuat *project* baru, maka *user* akan diarahkan ke *home page* dan *project* yang telah dibuat akan ditampilkan.

Dengan keterangan:

- *Type testing* (Manual dan *Automation*)  
(jika *user* memilih otomatis maka *form platform* akan otomatis *disable*, sedangkan jika memilih manual maka *user* bisa mengisi *platform*)
- *Platform* (*web* dan *mobile*).

## 6. *Edit project*

*User* dapat melakukan *Edit project* setelah mengklik *Edit* sesuai *project* yang akan diubah setelah itu akan muncul *form project* dan *project name* yang sudah terisi. *User* bisa mengubah untuk mengubah (*project name*, *type test*, *platform*), *form* tidak boleh kosong dan klik *button save* maka *project* berhasil diubah.

## 7. *Delete project*

*User* juga bisa menghapus *project* setelah mengklik *Delete* sesuai *project* yang akan dihapus setelah itu akan muncul *pop up* konfirmasi, apabila *user* salah mengklik *project* yang akan dihapus *user* bisa mengklik tombol *cancel* dan apabila *user* yakin akan menghapus *project* tersebut maka *User* bisa mengklik tombol *Delete*.

## 8. **Halaman *version***

Setelah *User* klik salah satu *project* nya maka akan masuk ke halaman *version* dimana akan otomatis ada *version 1* sesuai dengan *project* yang dibuat.

Dengan keterangan:

Untuk *version* hanya bisa dikelola oleh *PO* dan *QA* sedangkan *user* dengan *role DEV* tidak bisa untuk mengelola *version* semua *button* di halaman *version* akan tidak terlihat selain *button* member saja dan di member hanya bisa melihat anggota member saja.

## 9. *Create Version*

Fitur "*Create Version*" digunakan untuk membuat versi baru dari *project* yang ada dalam aplikasi. Versi digunakan untuk mengorganisasi dan melacak perkembangan pada *project* yang sedang dikerjakan.

### 10. *Edit Version*

Fitur "*Edit Version*" dapat digunakan *user* untuk mengubah *version name* pada *project* yang sedang dikerjakan jika terjadi kesalahan penulisan.

### 11. *Duplicate Version*

Jika *user* ingin melakukan *testing* aplikasi di versi terbaru, maka *User* dapat menduplikat *version* karena semua data meliputi *scenario*, *feature* dan *result* akan otomatis ikut ter duplikasi juga.

### 12. *Delete Version*

Fitur "*Delete Version*" dapat digunakan *user* untuk menghapus versi *project* yang ada. Penghapusan versi biasanya dilakukan ketika versi tersebut tidak lagi relevan atau perlu dihapus untuk alasan tertentu. *User* juga bisa menghapus *version* setelah mengklik *Delete* sesuai *version* yang akan dihapus, kemudian akan muncul *pop up* apakah yakin akan dihapus, apabila *user* salah mengklik *version* yang akan dihapus *user* bisa mengklik tombol *cancel* dan apabila *user* yakin akan menghapus *version* tersebut maka *User* bisa mengklik tombol *Delete*.

### 13. *Invite Member*

Fitur ini memungkinkan *user* mengundang untuk menambahkan *member* kedalam *project* yang sudah dibuat. Dengan cara klik *invite*, *User* diminta mengisi *email member* yang sudah terdaftar sebelumnya, lalu memilih *role* (*QA* atau *Programmer*) dan mengklik *button send*. Fitur ini hanya bisa dilakukan oleh *PO*. Setelah *PO* memasukkan *email* dan mengirim maka *member* yang di *invite* akan menerima *email* konfirmasi.

### 14. Perbedaan *Role PO, QA, dan Developer*

<i>PO</i>	<i>PO (Product owner)</i> dalam aplikasi <i>testing</i> berperan sebagai <i>User</i> dan semua <i>User</i> dapat menjadi <i>PO</i> , jika <i>User</i> tersebut membuat <i>project</i> baru dan <i>PO</i> memiliki hak akses semua aksi di manual <i>testing</i> ( <i>Create Project, Edit Project, Delete Project, Create Version, Edit Version, Duplicate Version, Delete Version, Create Scenario, Edit Scenario, delete Scenario, Create Feature, Edit Feature, Add Result, dan Edit Result</i> ) dan di <i>API testing</i> . Selanjutnya <i>PO</i> juga
-----------	---

	memiliki akses untuk manajemen tim melalui fitur <i>invite member</i> dengan menambahkan <i>member</i> dan menghapus <i>member</i> di halaman <i>invite member</i>
<i>QA</i>	<i>QA (Quality Assurance)</i> dalam aplikasi <i>Testing</i> mempunyai peran hampir sama dengan <i>PO</i> . Namun, terdapat sedikit perbedaan pada bagian manajemen tim. <i>PO</i> mempunyai akses untuk <i>invite</i> para <i>member</i> , tetapi untuk <i>QA</i> tidak memiliki akses untuk <i>invite member</i> . Melainkan <i>QA</i> hanya dapat mengakses para <i>member</i> yang tergabung dalam <i>project</i> tersebut.
<i>Developer</i>	<i>Developer</i> dalam aplikasi <i>Testing</i> yang telah diundang ke dalam <i>project</i> tersebut tidak mempunyai hak akses apapun kecuali hanya memiliki akses untuk melihat <i>project</i> . <i>Project</i> tersebut diantaranya yaitu <i>project</i> Manual maupun <i>API Testing</i> yang telah dikerjakan maupun belum dikerjakan oleh pihak <i>QA</i> .

### 15. Detail Member

Fitur ini digunakan *user* untuk melihat dan mengelola informasi *member*, Pada halaman *member* terdapat beberapa informasi dan fitur.

### 16. Delete Member

Fitur ini digunakan *user* dengan *role PO* untuk menghapus *member*.

### 17. Halaman Scenario

Halaman ini digunakan *user* untuk membuat, mengelola, dan melacak *Scenario* dalam *project* yang sudah tersusun sesuai *version*. Setelah *User* Klik salah satu *version* maka akan masuk ke halaman *scenario*, pada halaman *scenario* terdapat beberapa fitur.

keterangan:

Label Warna Biru	label tersebut digunakan untuk memberikan kategori <i>scenario</i> pada <i>test case</i> . Dengan menggunakan label, <i>user</i> dapat mengelompokkan sesuai dengan fitur.
Selain Label Warna Biru	label tersebut digunakan untuk menunjukkan <i>bug</i> pada <i>test case</i> yang sudah diinputkan <i>result</i> nya. terdapat 3 label untuk <i>bug</i> , yaitu: <ul style="list-style-type: none"> <li>• <i>Result state</i> : <i>pass dan fail</i></li> <li>• <i>Priority</i> : <i>low, medium, high, urgent</i></li> <li>• <i>Severity</i> : <i>low, minor, major, critical</i></li> </ul>

Label Warna Hijau	label tersebut menunjukkan status ketika <i>result</i> dengan kondisi <i>pass</i> , sehingga tidak menampilkan label <i>priority</i> dan <i>severity</i> .
-------------------	--

### 18. *Create Scenario*

Fitur ini digunakan *user* untuk membuat *Scenario* baru.

Dengan keterangan:

<i>Dropdown Feature</i>	<i>user</i> harus <i>create feature</i> terlebih dahulu, setelah itu <i>user</i> bisa memilih <i>feature</i> yang sudah ditambahkan sebelumnya
<i>Dropdown Category</i>	<i>user</i> memilih <i>positive</i> atau <i>negative</i> di <i>form category</i>

### 19. *Edit Scenario*

Fitur ini digunakan *user* untuk meng-*edit* dan memperbarui *scenario* yang ada dalam *project*.

### 20. *Delete Scenario*

Selain bisa mengubah data *scenario user* juga bisa menghapus *scenario*.

### 21. *Create Feature*

Fitur ini digunakan *user* untuk membuat *feature* dalam *scenario* pada *project*. *Feature* membantu *user* untuk mencakup berbagai kemungkinan jalur uji yang berbeda. *Feature* ditambahkan di *form scenario*, dan otomatis akan terisi ke dalam pilihan pada *form feature* dan *user* dapat mengubah nama *feature* tersebut.

### 22. Tahap *Edit Feature*

Jika *user* salah atau *typo* memasukkan nama *feature* tidak masalah karena *user* bisa mengubah nama *feature* tersebut dengan masuk ke halaman *feature list*.

### 23. *Filter Scenario*

*Filter Scenario* sendiri dibuat agar memudahkan *user* untuk mengelompokkan data *scenario* berdasarkan *feature* yang bisa dipilih.

## 24. Halaman Detail *Scenario*

Jika *user* belum menambahkan data *result* maka tampilan di Detail *Scenario* dan jika *user* sudah melakukan *testing* dan menambahkan data *result* maka tampilan di detail test case akan bertambah informasi.

## 25. *Create Result*

Fitur ini digunakan *user* untuk membuat dan mencatat hasil dari *scenario* yang telah dijalankan. Dengan fitur ini, *user* dapat dengan mudah melacak dan merekam hasil uji untuk setiap kasus uji yang dieksekusi.

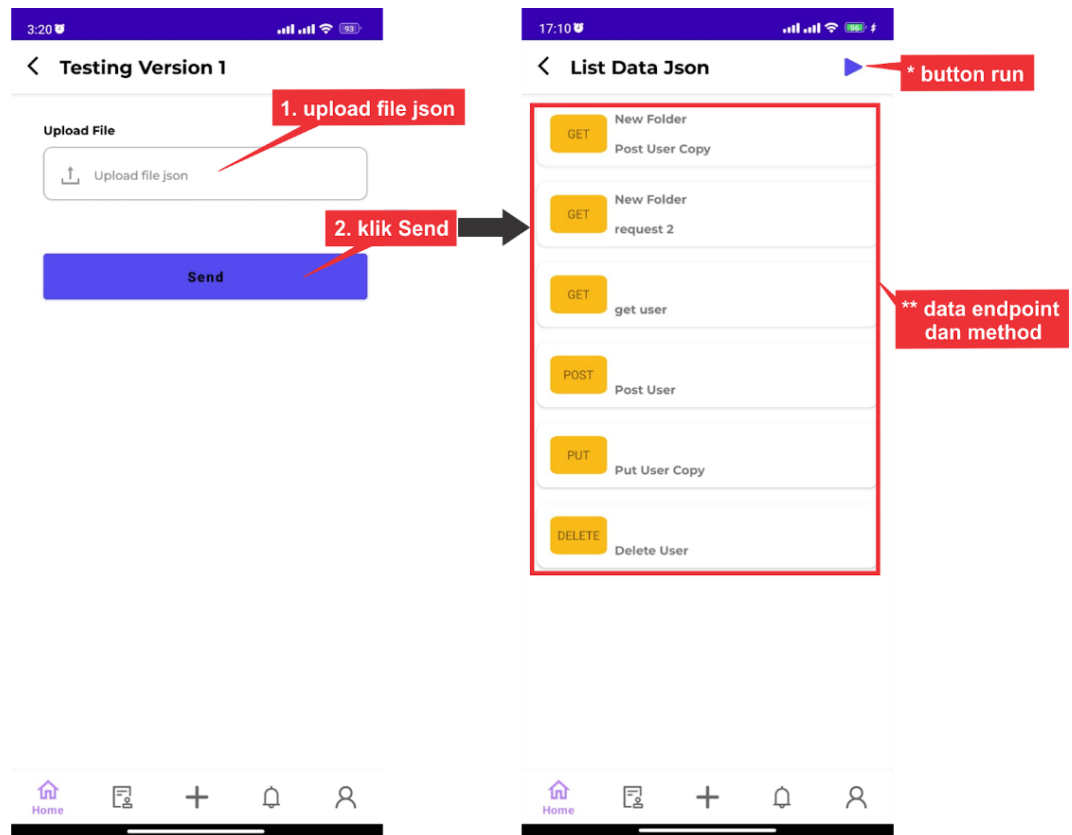
keterangan:

<i>Actually</i>	apa yang terjadi setelah melakukan <i>test step</i>
<i>State</i>	terdiri dari <i>Pass</i> dan <i>Fail</i>
<i>Priority</i>	terdiri dari <i>Urgent, high, medium, low</i>
<i>Severity</i>	terdiri dari <i>Critical, major, minor, low</i>
<i>Note</i>	catatan untuk <i>QA</i> tentang <i>bug</i> yang terjadi
<i>Attachment</i>	untuk menunjukkan bukti <i>bug</i> yang terjadi (file jpg dengan <i>max size 2mb</i> )

## 26. *Edit Result*

Fitur ini dapat digunakan untuk meng-*edit* dan memperbarui *result* yang telah dicatat sebelumnya.

## 27. Automation API



Sebelum ke tahap *Automation API User* harus membuat *project* baru dengan memilih *Testing type Automatic* dan harus mempersiapkan file *json*, cara mendapatkan file tersebut dengan meng-*export collection json* dari aplikasi Postman. Setelah berhasil meng-*upload file collection* selanjutnya klik *button send*, maka aplikasi akan membaca dan memunculkan *endpoint* dan *method* apa saja yang ada di *json collection* yang sudah di *upload* sebelumnya. untuk tampilannya seperti ini.

## 28. Report

Fitur *Report* digunakan untuk mengetahui seberapa banyak *scenario* yang sudah di *testing*. Untuk perhitungan dari *report* ini dengan menghitung seberapa banyak *result* yang sudah ditambahkan di *scenario*. Jika di salah satu *version* semua *scenario* terisi dengan *result* maka *version* tersebut akan 100 persen tercapai.



### **29. Notification**

*Notification* digunakan untuk mengetahui pesan yang muncul ketika ada *member* baru yang *join* ke *project* kita dan kita sendiri masuk sebagai *role* apa.

### **30. Account**

Fitur *Account* pada navigasi digunakan *user* untuk mengelola informasi dari akun dan terdapat menu *Account Setting*, *Achievement*, *about* dan *Log out*. Penjabaran dari masing-masing menu akan dijelaskan pada poin berikutnya.

### **31. Account Setting**

Menu ini dapat digunakan untuk mengatur dan mengelola pengaturan terkait dengan akun pengguna, dimana *user* dapat mengubah nama dan foto profil. Selain itu *user* juga dapat menghapus *account*.

### **32. Achievement**

Menu *Achievement* digunakan untuk menunjukkan seberapa banyak *QA* sudah menambahkan *result* pada tiap *scenario*. Diawal *QA* akan memiliki tingkatan *Beginner* dan jika *QA* menambahkan *result* 200 *scenario* maka tingkatan *QA* akan naik ke *intermediate*.

### **33. About**

Menu *About* menyediakan informasi tim kerja yang terlibat dalam pembuatan dan pengembangan dari aplikasi *testing*.

### **34. Logout**

Menu *Logout* dapat digunakan *user* untuk keluar dari akun Anda dalam aplikasi *Testing*.