



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**IMPLEMENTASI *AUTOMATION TESTING* MENGGUNAKAN
TOOL KATALON STUDIO PADA APLIKASI *MOBILE*
ANDROID: STUDI KASUS PADA APLIKASI DUNIA GAMES**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana
komputer**

**DANANG TRI SAPUTRO
0110220057**

**PROGRAM STUDI TEKNIK INFORMATIKA SEKOLAH
TINGGI TERPADU NURUL FIKRI
DEPOK
AGUSTUS 2024**



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**IMPLEMENTASI *AUTOMATION TESTING* MENGGUNAKAN
TOOL KATALON STUDIO PADA APLIKASI *MOBILE*
ANDROID : STUDI KASUS PADA APLIKASI DUNIA GAMES**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana komputer

DANANG TRI SAPUTRO

0110220057

**PROGRAM STUDI TEKNIK INFORMATIKA SEKOLAH
TINGGI TERPADU NURUL FIKRI
DEPOK
AGUSTUS 2024**

HALAMAN PERNYATAAN ORISINALITAS

Skripsi/Tugas Akhir ini adalah hasil karya peneliti, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Danang Tri Saputro
NIM : 0110220057

Depok, 05 Agustus 2024

Tanda Tangan



Danang Tri Saputro

HALAMAN PENGESAHAN

Skripsi/Tugas akhir ini diajukan oleh :

Nama : Danang Tri Saputro

NIM : 0110220057

Program Studi : Teknik Informatika

Judul Skripsi : Implementasi Automation Testing Menggunakan Tool Katalon Studio Pada Aplikasi Mobile Android : Studi Kasus Pada Aplikasi Dunia Games

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJIAN

Pembimbing



(Salman El Farisi, S.Kom, M.Kom.)

Penguji



(Imam Haromain, S.Si., M.Kom.)

Ditetapkan di : Depok

Pada Tanggal : 27 Juli 2024

KATA PENGANTAR

Puji syukur peneliti panjatkan kehadirat Allah SWT, karena atas berkat rahmat-Nya, peneliti dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan Skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Komputer Program Studi Sistem Informasi pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Peneliti menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi peneliti untuk menyelesaikan Skripsi/tugas Akhir ini. Oleh karena itu, peneliti mengucapkan terima kasih kepada:

1. Allah SWT
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moral maupun materil dalam penyelesaian tugas ini.
3. Bapak Lukman Rosyidi, STT., MM., MT, selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Bapak Henry Saptono, S.Si., M.Kom., selaku Dosen Pembimbing Akademik Teknik Informatika angkatan 2020 kelas Akhir Pekan.
5. Ibu Tifani Nabarian, S.Kom., M.T.I., selaku Ketua Program Studi Teknik Informatika.
6. Bapak Salman El Farisi, S.Kom, M.Kom., selaku Dosen Pembimbing Tugas Akhir yang terus memberikan dorongan dan dukungan dengan sabar dan selalu kebersamai dalam menyelesaikan penulisan ilmiah ini.
7. Bapak Imam Haromain, S.Si., M.Kom., selaku Dosen Penguji dalam penelitian ini.
8. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing peneliti dalam menuntut ilmu yang telah diberikan.
9. Lingkungan kerja Divisi ITQA dari Telkomsigma yang turut membantu dalam penyelesaian penelitian ini.

Dalam penulisan ilmiah ini tent saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang peneliti miliki. Walaupun demikian, peneliti telah

berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati peneliti menerima kritikan dan saran dari pembaca. Akhir kata, peneliti berharap bahwa Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu

Depok, 24 Juni 2024

Peneliti

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Danang Tri Saputro

NIM : 0110220057

Program Studi : Teknik Informatika

Jenis karya : Skripsi/Tugas Akhir

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT- NF **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty - Free Right)** atas karya ilmiah saya yang berjudul :

**IMPLEMENTASI AUTOMATION TESTING MENGGUNAKAN TOOL
KATALON STUDIO PADA APLIKASI MOBILE ANDROID : STUDI
KASUS PADA APLIKASI DUNIA GAMES**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 05 Agustus 2024

Yang Menyatakan



(Danang Tri Saputro)

ABSTRAK

Nama : Danang Tri Saputro
NIM : 0110220057
Program Studi : Teknik Informatika
Judul : Implementasi Automation Testing Menggunakan Tool Katalon Studio Pada Aplikasi Mobile Android : Studi Kasus Pada Aplikasi Dunia Games

Pengujian otomatis atau biasa disebut *Automation testing* merupakan suatu proses yang melibatkan penggunaan script atau alat-alat khusus untuk mengotomatisasi eksekusi kasus uji. Pada penelitian ini *automation testing* dilakukan pada aplikasi Dunia games menggunakan *tool* katalon studio, Dunia Games merupakan platform dengan berbagai fitur di bidang game, Fitur-fitur yang ditawarkan berupa: menjadi *platform* media berita terkait *e-sport*, penyedia jasa pembelian paket dan *topup item game* dan lainnya. Penelitian ini menggunakan metode evaluatif yaitu mengukur dan menentukan hasil dari rencana atau proyek. Pendekatan tersebut melibatkan proses diperolehnya data dari hasil eksperimen implementasi automation testing pada 12 skenario yang telah ditentukan sebelumnya pada aplikasi Dunia Games. *Output* dari penelitian ini adalah data dari 12 skenario tes baik yang dapat dilakukan automasi maupun yang tidak bisa dilakukan, data tersebut menjadi bahan evaluasi tentang implementasi automasi dalam pengujian aplikasi. Hasil implementasi automation testing dari 12 *test case*, sebanyak 11 *test case* berhasil untuk dieksekusi dan 1 *test case* gagal dieksekusi dikarenakan terdapat *issue error* pada *backend*.

Kata kunci : *Automation testing*, *Android*, Katalon studio,

ABSTRACT

Name : Danang Tri Saputro

NIM : 0110220057

Study Program : Informatics Engineering

Title : *Implementation of Automation Testing Using Katalon Studio Tool on Android Mobile Applications: A Case Study on Dunia Games Application.*

automation testing, is a process that involves the use of scripts or specialized tools to automate the execution of test cases. In this study, automation testing was conducted on the Dunia Games application using the Katalon Studio tool. Dunia Games is a platform with various features in the gaming field, offering features such as being a news media platform related to e-sports, providing services for purchasing packages and topping up game items, among others. This research uses an evaluative method, which involves measuring and determining the outcomes of a plan or project. This approach involves obtaining data from the experimental implementation of automation testing on 12 predefined scenarios in the Dunia Games application. The output of this research is data from 12 test scenarios, both those that can be automated and those that cannot, which serves as evaluation material regarding the implementation of automation in application testing. The results of the automation testing implementation on 12 test cases showed that 11 test cases were successfully executed, while 1 test case failed to execute due to a backend error issue.

Keywords: *Automation testing, Android, Katalon Studio*

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR.....	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	vii
ABSTRAK	viii
ABSTRACT	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar belakang	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan dan Manfaat Penelitian	3
1.4 Batasan Masalah	4
1.5 Sistematika Penulisan	5
BAB II KAJIAN LITERATUR	6
2.1 <i>Android</i>	6
2.2 <i>Software Testing Life Cycle (STLC)</i>.....	6
2.2.1 Tahap-tahap pada <i>software testing life cycle</i>	7
2.3 <i>Blackbox Testing</i>	9
2.4 <i>Automation Testing</i>	10
2.5 Katalon Studio.....	11
2.6 Penelitian Terkait.....	12

BAB III METODOLOGI PENELITIAN	14
3.1 Tahapan Penelitian	14
3.1.1 Kajian Literatur	15
3.1.2 Pengumpulan Data	15
3.1.3 Analisa Data	15
3.1.4 Pembuatan <i>Test case</i>	15
3.1.5 Pembuatan <i>Script</i>	15
3.1.6 Eksekusi <i>Script</i>	16
3.1.7 Evaluasi Hasil	16
3.1.8 Analisa <i>Error</i>	16
3.1.9 Laporan	17
3.2 Rancangan Penelitian	17
3.2.1 Jenis Penelitian.....	17
3.2.2 Metode Pengumpulan Data	17
3.2.3 Metode Analisa	18
3.2.4 Metode Pengujian	18
3.3 Lingkungan Penelitian.....	18
3.4 Alat dan Bahan.....	19
BAB VI HASIL PEMBAHASAN	20
4.1 Analisa Perancangan	20
4.1.1 Perancangan Sistem	20
4.1.2 Pembuatan dan Analisa <i>Testcase</i>	22
4.2 Implementasi dan Evaluasi	25
4.2.1 <i>Setup Environment Automation Testing</i>	25
4.2.2 Implementasi <i>Script Automation</i>	26
4.2.3 Ekeekusi <i>Script Automation</i>	62
4.2.4 Hasil Eksekusi <i>Script Automation</i>	66
4.3 Evaluasi Hasil Implementasi.....	75

BAB V.....	77
KESIMPULAN DAN SARAN	77
5.1 Kesimpulan	77
5.2 Saran	77
DAFTAR PUSTAKA	78
LAMPIRAN.....	80

DAFTAR GAMBAR

Gambar 2. 1 <i>Software Testing Life Cycle</i>	7
Gambar 3. 1 Tahapan Penelitian	14
Gambar 4. 1 Perancangan <i>Automation testing</i>	20
Gambar 4. 2 <i>Appium Directory</i>	26
Gambar 4. 3 Konfigurasi <i>Object Spy Mobile</i>	27
Gambar 4. 4 list objek dari halama aplikasi	28
Gambar 4. 5 Objek yang tersimpan di <i>Object Repository</i>	28
Gambar 4. 6 Objek fitur registrasi dan login.....	29
Gambar 4. 7 TC001 - <i>Input email</i> dengan format yang salah.....	29
Gambar 4. 8 TC001_ <i>evidence1</i>	30
Gambar 4. 9 TC001- <i>Input email</i> yang sudah ter-registrasi.....	30
Gambar 4. 10 TC001_ <i>evidence2</i>	31
Gambar 4. 11 TC001- <i>Input password</i> yang formatnya salah	31
Gambar 4. 12 TC001_ <i>evidence3</i>	32
Gambar 4. 13 TC001 - <i>Input</i> kode OTP yang salah.....	32
Gambar 4. 14 TC001_ <i>evidence4</i>	33
Gambar 4. 15 TC002 - <i>Input email</i> dan password yang benar	33
Gambar 4. 16 TC002_ <i>evidence1</i>	34
Gambar 4. 17 TC002 - <i>Input</i> OTP	34
Gambar 4. 18 TC002_ <i>evidence2</i>	35
Gambar 4. 19 TC002 – Menyetujui privasi polsih.....	35
Gambar 4. 20 TC002_ <i>evidence3</i>	36
Gambar 4. 21 TC003 - <i>Input Email</i> dengan format yang salah	36
Gambar 4. 22 TC003_ <i>evidence1</i>	37
Gambar 4. 23 TC003 - <i>Input password</i> yang salah.....	37
Gambar 4. 24 TC003_ <i>evidence2</i>	38
Gambar 4. 25 TC004 - <i>Input Email</i> yang telah terdaftar	38
Gambar 4. 26 TC004_ <i>evidence1</i>	39
Gambar 4. 27 TC004 - <i>Input password</i> yang benar	39

Gambar 4. 28 TC004_ <i>evidence2</i>	40
Gambar 4. 29 - TC005 - Verifikasi <i>section</i>	40
Gambar 4. 30 TC005_ <i>evidence1</i>	41
Gambar 4. 31 TC005 - Menekan tombol <i>like</i> dan <i>bookmark</i>	41
Gambar 4. 32 TC005_ <i>evidence2</i>	42
Gambar 4. 33 TC005 - menulis komentar pada artikel.....	42
Gambar 4. 34 TC005_ <i>evidence3</i>	43
Gambar 4. 35 TC006 - Verifikasi <i>section</i> dan <i>tab</i>	43
Gambar 4. 36 TC006_ <i>evidence1</i>	44
Gambar 4. 37 TC006 - Membuat Postingan	44
Gambar 4. 38 TC006_ <i>evidence2</i>	45
Gambar 4. 39 TC006_ <i>evidence3</i>	45
Gambar 4. 40 menekan tombol <i>like</i> dan menulis komentar.....	45
Gambar 4. 41TC007_ <i>evidence1</i>	46
Gambar 4. 42 TC007_ <i>evidence2</i>	46
Gambar 4. 43 TC008 - <i>Input</i> nama tournament	47
Gambar 4. 44 TC008_ <i>evidence1</i>	47
Gambar 4. 45 TC008 - <i>Setting deadline Tournament</i>	48
Gambar 4. 46 TC008_ <i>evidence2</i>	49
Gambar 4. 47 TC008_ <i>evidence3</i>	49
Gambar 4. 48 TC008 - <i>Input</i> jumlah partisipan	49
Gambar 4. 49 TC008 - <i>Setting deadline</i> pendaftaran <i>Tournament</i>	50
Gambar 4. 50 TC008 - membuat <i>bracket tournament</i>	50
Gambar 4. 51TC009_ <i>evidence4</i>	51
Gambar 4. 52 TC009 - <i>Input</i> nama roomchat	52
Gambar 4. 53 TC009_ <i>evidence1</i>	52
Gambar 4. 54 TC009_ <i>evidence2</i>	52
Gambar 4. 55 TC009 - memilih <i>game</i>	52
Gambar 4. 56 TC009_ <i>evidence3</i>	53
Gambar 4. 57 TC009_ <i>evidence4</i>	53
Gambar 4. 58 TC009 - <i>Input</i> deskripsi dan lokasi	53

Gambar 4. 59 TC009_evidence5.....	54
Gambar 4. 60 TC009_evidence6.....	54
Gambar 4. 61 TC009 - <i>input</i> jumlah <i>member</i>	54
Gambar 4. 62 Tc009_evidence7.....	55
Gambar 4. 63 TC009_evidence8.....	55
Gambar 4. 64 TC010 - <i>Input</i> pesan.....	55
Gambar 4. 65 TC010_evidence1	56
Gambar 4. 66 TC010_evidence2.....	56
Gambar 4. 67 TC011 - tekan tombol <i>buy</i> tanpa <i>input</i> apapun	56
Gambar 4. 68 TC011_evidence1	57
Gambar 4. 69 TC011 - tekan tombol <i>buy</i> tanpa <i>input ID games</i>	57
Gambar 4. 70 TC011_evidence2.....	58
Gambar 4. 71 TC011_input OTP yang salah.....	58
Gambar 4. 72 TC011_evidence3.....	58
Gambar 4. 73 TC012 - <i>Input ID games</i> dan pilih paket <i>diamond</i>	59
Gambar 4. 74 TC012_evidence1	60
Gambar 4. 75 TC012_evidence2.....	60
Gambar 4. 76 TC012_input nomor dengan format yang benar	60
Gambar 4. 77 TC012_evidence2.....	61
Gambar 4. 78 TC012_evidence3.....	61
Gambar 4. 79 TC012 - <i>Input</i> OTP	61
Gambar 4. 80 TC012_evidence5.....	62
Gambar 4. 81 TC012_evidence6.....	62
Gambar 4. 82 <i>Test Suite</i>	63
Gambar 4. 83 <i>Test suite</i> per-fitur	63
Gambar 4. 84 Tombol <i>Add</i> pada <i>Test suite</i>	64
Gambar 4. 85 <i>Test case</i> dipilih.....	64
Gambar 4. 86 <i>Test suite</i> yang siap untuk dieksekusi	64
Gambar 4. 87 Pilih <i>paltform</i> untuk menjalankan <i>script</i>	65
Gambar 4. 88 Pilih <i>device</i> yang tesambung.....	65
Gambar 4. 89 <i>Test case</i> yang tereksekusi	65

Gambar 4. 90 <i>Test Ops</i>	66
Gambar 4. 91 <i>Dashboard</i> Hasil pengujian.....	66
Gambar 4. 92 <i>Summary result testing</i>	66
Gambar 4. 93 <i>Test case</i> yang telah tereksekusi.....	67
Gambar 4. 94 TC001 – Status <i>Passed</i>	68
Gambar 4. 95 TC002-Status <i>Passed</i>	68
Gambar 4. 96 TC003-Status <i>Passed</i>	69
Gambar 4. 97 TC004 - Status <i>Passed</i>	69
Gambar 4. 98 TC005 - Status <i>Passed</i>	70
Gambar 4. 99 TC006 - Status <i>Passed</i>	70
Gambar 4. 100 TC007 – Status <i>Passed</i>	71
Gambar 4. 101 TC008 - Status <i>Failed</i>	71
Gambar 4. 102 TC008 - <i>Detail Error</i>	72
Gambar 4. 103 TC009 - Status <i>Passed</i>	72
Gambar 4. 104 TC010 - Status <i>Passed</i>	73
Gambar 4. 105 TC011 - Status <i>Passed</i>	73
Gambar 4. 106 TC012 - Status <i>Passed</i>	74

DAFTAR TABEL

Tabel 2. 1 Penelitian terkait	12
Tabel 4. 1 Test case	22
Tabel 4. 2 Test case berdasarkan status	75

BAB I

PENDAHULUAN

1.1 Latar belakang

Dalam era digital yang terus berkembang, perangkat lunak menjadi komponen vital dalam berbagai aspek kehidupan kita, mulai dari aplikasi perbankan hingga kendaraan otonom. Perangkat lunak yang beredar di masyarakat cukup beragam yang paling umum yaitu aplikasi *android*. Aplikasi *android* tumbuh cukup pesat di Indonesia hal ini berbanding lurus dengan jumlah kepemilikan *smartphone* di Indonesia. Menurut lembaga riset *Digital Marketing Emarketer* [1] memperkirakan jumlah pengguna aktif *smartphone* di Indonesia pada tahun 2018 akan menembus lebih dari 100 juta orang. Dalam konteks ini peningkatan keandalan, kinerja, dan keamanan perangkat lunak menjadi imperatif bagi organisasi pengembang. Untuk memenuhi tantangan ini pengujian kualitas dari perangkat lunak telah muncul sebagai strategi yang menjanjikan untuk meningkatkan efisiensi dan kualitas pengujian perangkat lunak.

PT Sigma Cipta Caraka atau bisa disebut Telkomsigma merupakan perusahaan solusi *IT end-to-end* yang salah satu bidangnya berfokus pada penjaminan kualitas perangkat lunak atau *software quality assurance*. Telkomsigma telah menjadi vendor proyek besar dari beberapa perusahaan salah satunya adalah PT Telekomunikasi Seluler (Telkomsel) dan aplikasi Dunia Games. Dunia Games merupakan *platform* dengan berbagai fitur di bidang *game*. Fitur-fitur yang ditawarkan berupa: menjadi *platform* media berita terkait *e-sport*, penyedia jasa pembelian paket dan *topup item game*, *platform* penyelenggaraan turnamen dan *event e-sport*, serta menyediakan ruang komunitas *game* dan saat ini telah memiliki 1 juta pengunduh di *playstore*.

Telkomsigma selaku vendor *quality assurance* dari aplikasi Dunia Games bertanggungjawab memastikan apakah aplikasi tersebut sudah layak digunakan oleh pengguna maka dilakukan pengujian. Pengujian pun perlu

dilakukan dari segala sisi perangkat lunak mulai dari *User Interface*, *Functional*, *NonFunctional*, *API & Performance*. Aspek-aspek tersebut dikategorikan ke metode *black box testing* yaitu melakukan evaluasi berdasarkan detail aplikasi seperti penampilan antarmuka, fungsionalitas yang tersedia, dan kesesuaian alur kerja dengan kebutuhan sistem yang diinginkan oleh pembuatnya [2]. Pada proyek aplikasi dunia games ini peneliti sebagai pekerja terlibat dalam *functional testing* yang dilakukan secara manual dalam artian setiap skenario yang diuji dilakukan langsung oleh peneliti dengan *device* fisik. Metode manual ini memang cukup umum di dunia *quality assurance* tetapi memiliki kekurangan yaitu kurang efisien jika pengujian dilakukan secara berulang karena dalam pengujian aplikasi dunia games fitur-fitur yang sudah ada lebih dulu diuji ulang untuk memastikan semua fitur berfungsi dengan normal setelah penambahan fitur baru atau biasa disebut Regresi / *Regression*. Untuk pengujian regresi dibutuhkan sejumlah orang untuk melakukannya padahal fiturnya sudah diuji beberapa kali. Selain itu pengujian manual sering terjadi *human error* dalam penyajian hasil pengujian, karena dari itu uji automasi atau *automation testing* dibutuhkan karena dapat menutupi kekurangan tersebut.

Ada beberapa alasan mengapa uji automasi berkembang pesat, antara lain efisiensi pengujian yang lebih tinggi, tingkat akurasi dan keandalan yang ditingkatkan, kemampuan untuk menggunakan dan mengulang skrip uji, cakupan pengujian yang lebih luas, simulasi lingkungan pengguna, pengembalian investasi yang lebih tinggi dengan penghematan waktu dan biaya, kemampuan untuk menangani *volume* dan simultanitas yang lebih besar, serta deteksi awal *bug* [3]. Beberapa *tools automation* yang *free* digunakan dan setiap tool memiliki fitur dan cara penggunaan yang hampir sama jadi pengujian automasi sudah sangat mungkin untuk dilakukan. Katalon studio merupakan salah satu tools yang *user friendly* karena menawarkan fitur *record and playback* yang dapat digunakan oleh orang awam. Katalon Studio menggunakan Apache Groovy sebagai bahasa pemrogramannya dan mendukung mode non-GUI untuk integrasi CI/CD.

Namun, Katalon Studio tidak mendukung pengujian terdistribusi dan hanya terfokus pada pengujian otomatisasi untuk Web, Seluler, dan API. Katalon menyediakan Katalon Studio dan *Recorder* sebagai alat untuk uji otomatisasi [3].

Penelitian ini berjudul **Implementasi Automation Testing Menggunakan Tool Katalon Studio Pada Aplikasi Mobile Android: Studi Kasus Pada Aplikasi Dunia Games** yang bertujuan untuk mengetahui cara implementasi uji otomasi menggunakan *tool* katalon studio dan melihat seberapa efektif dan efisien penerapannya pada projek aplikasi dunia games.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, maka uraian permasalahan dalam draft tugas akhir ini adalah sebagai berikut :

“Bagaimana melakukan implementasi uji automasi pada aplikasi dunia games”

Untuk menjawab rumusan masalah tersebut peneliti akan mengkaji beberapa hal sebagai berikut :

- a. Bagaimana pembuatan *script* uji automasi menggunakan *tool* katalon studio untuk pengujian aplikasi Dunia Games?
- b. Bagaimana cara menjalankan *script* uji automasi menggunakan *tool* katalon studio untuk pengujian aplikasi Dunia Games?
- c. Bagaimana hasil dari implementasi uji automasi menggunakan *tool* katalon studio untuk pengujian aplikasi Dunia Games?

1.3 Tujuan dan Manfaat Penelitian

- a. Mengetahui cara pembuatan *script* uji automasi menggunakan *tool* katalon studio pada aplikasi Dunia Games.
- b. Mengetahui cara menjalankan *script* uji automasi menggunakan *tool* katalon studio pada aplikasi Dunia Games.
- c. Mengetahui hasil implementasi uji automasi menggunakan *tool* katalon studio untuk pengujina. Aplikasi Dunia Games.

Adapun manfaat penulisan tugas akhir ini sebagai berikut :

- a. Adanya pemahaman teori tentang pengujian automasi kepada aplikasi Dunia Games.
- b. Adanya hasil evaluasi yang dapat dijadikan sebagai pertimbangan untuk memperbaiki aplikasi Dunia Games.
- c. Perkaya literatur teknik informatika terkait bidang *Quality Assurance*.

1.4 Batasan Masalah

Adanya batasan masalah saat melakukan penelitian ini adalah sebagai berikut:

- a. Pengujian dilakukan hanya pada beberapa fitur seperti fitur *login*, fitur *roomchat*, *gamefeed*, artikel dan topup.
- b. Automasi diimplementasikan pada 12 skenario pengujian, antara lain :
 1. Gagal Melakukan registrasi (email/password yang belum sesuai)
 2. Sukses melakukan registrasi
 3. Login dengan *invalid credential*
 4. Login dengan *valid credential*
 5. Menyukai dan menyimpan artikel
 6. Membuat postingan di *gamefeed*
 7. Menyukai dan mengomentari postingan user lain
 8. Membuat sebuah tournament
 9. Membuat sebuah roomchat
 10. Mengirim pesan ke *room chat*
 11. Gagal melakukan topup
 12. Sukses Melakukan topup *diamond*
- c. Fitur katalon studio yang digunakan pada penelitian ini yaitu *Spy Object*, *testcase*, *testsuite* dan *testops*.

1.5 Sistematika Penulisan

Sistematika penulisan Tugas Akhir ini adalah sebagai berikut :

- a. BAB 1: PENDAHULUAN menyajikan gambaran umum tentang pencapaian tujuan akhir penelitian ini yaitu **implementasi *automation testing* menggunakan tool katalon studio pada aplikasi *mobile android: studi kasus pada aplikasi dunia games*** . Ini mencakup latar belakang sejarah tentang *automation testing* dan aplikasi dunia games , deskripsi masalah, tujuan dan manfaat penelitian, batasan masalah, serta struktur penulisan.
- b. BAB II LANDASAN TEORI, di bab ini akan memberikan analisis mendalam terhadap teori-teori dan literatur yang menjadi pondasi utama dari penelitian ini. Topik yang dibahas berupa pengertian dari STLC (*Software Test Life Cycle*), *Android*, *Automation testing*, dan Katalon Studio. Melalui pembahasan yang mendetail, bagian ini akan menguraikan konsep-konsep kunci yang relevan dengan topik penelitian, serta menjelaskan bagaimana teori-teori ini berperan dalam konteks penelitian yang sedang dilakukan.
- c. BAB III METODOLOGI PENELITIAN, Bagian ini akan memberikan gambaran terperinci tentang langkah-langkah yang diambil dalam melakukan penelitian ini, mencakup tahap-tahap pengimplementasikan automasi pada dunia games, *tool* yang digunakan, Analisa data, hingga metode pengujian yang digunakan.
- d. BAB IV IMPLEMENTASI DAN EVALUASI, Bagian ini akan menguraikan secara komprehensif proses pelaksanaan, hasil yang telah dicapai, dan evaluasi dari jalannya penelitian implementasi *automation testing* terhadap aplikasi dunia games menggunakan *tool* katalon studio.
- e. BAB V KESIMPULAN DAN SARAN, Bagian ini berfungsi sebagai kesimpulan dari penelitian, menyajikan jawaban terhadap rumusan masalah yang telah ditentukan terkait implementasi *automation testing* terhadap aplikasi dunia games menggunakan katalon studio.

BAB II

KAJIAN LITERATUR

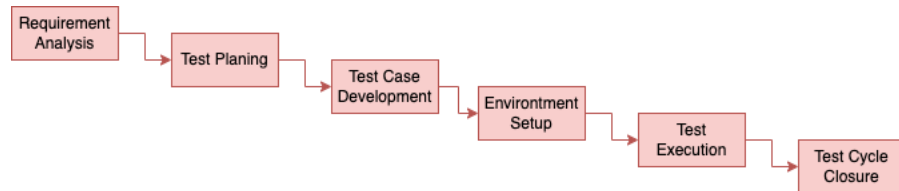
2.1 Android

Android adalah sebuah sistem operasi yang dirancang untuk perangkat *mobile* [4]. Dalam ekosistem *Android*, tidak ada perbedaan antara aplikasi inti, yang merupakan aplikasi bawaan sistem, dengan aplikasi pihak ketiga, yang dikembangkan oleh pengembang lain. Semua aplikasi ini memiliki akses yang sama melalui *Application Programming Interface (API)* yang disediakan oleh *Android*. *API* ini memberikan kemampuan bagi aplikasi untuk mengakses berbagai aspek perangkat keras, seperti kamera dan sensor, serta data ponsel, termasuk kontak dan lokasi pengguna. Selain itu, *API* ini juga memungkinkan aplikasi untuk mengakses data sistem, memungkinkan integrasi yang lebih dalam dan fungsionalitas yang lebih kaya bagi pengguna.

2.2 Software Testing Life Cycle (STLC)

Dalam *software testing* mengenal istilah *software testing life cycle (STLC)* yaitu merujuk pada serangkaian proses terstruktur yang digunakan dalam pengujian perangkat lunak [5]. Ini sering dianggap sebagai bagian integral dari siklus hidup pengembangan perangkat lunak. Proses ini terdiri dari beberapa tahapan yang disusun secara berurutan, dari perencanaan hingga penyelesaian pengujian. Setiap tahap memiliki kriteria dan hasil yang telah ditetapkan sebelumnya untuk menjamin kelancaran dan keberhasilan pengujian. Tahap-tahapnya meliputi perencanaan pengujian, analisis kebutuhan pengujian, perancangan skenario pengujian, pembuatan kasus uji, eksekusi uji, pelaporan hasil, dan evaluasi kinerja pengujian.

2.2.1 Tahap-tahap pada *software testing life cycle*



Gambar 2. 1 *Software Testing Life Cycle*

Berikut tahapan dari *software testing life cycle* [6] :

a. *Requirement Analysis*

Dalam *Software Testing Life Cycle* (STLC), tahapan pertama adalah menganalisis kebutuhan. Pada tahap ini, para pengujian (*tester*) mempelajari persyaratan perangkat lunak untuk mengidentifikasi aspek yang dapat diuji dari sudut pandang pengujian. Mereka berinteraksi dengan berbagai pihak, seperti klien, analis bisnis, kepala teknis, arsitek sistem, dan lain-lain, dengan tujuan untuk memahami persyaratan secara mendalam dan rinci. Hal ini membantu dalam merencanakan pengujian yang tepat dan memastikan bahwa semua aspek penting dari perangkat lunak akan diuji sesuai dengan kebutuhan pengguna dan bisnis.

b. *Test Planning*

Test planning adalah tahap di mana setiap detail pengujian direncanakan dalam siklus STLC. Pada tahap ini, para pengujian mengumpulkan semua persyaratan yang diperlukan, termasuk analisis *file brief* untuk mengidentifikasi kebutuhan sistem dan biaya yang mungkin diperlukan, serta menyiapkan upaya pengujian. Hasil dari tahap ini adalah rencana pengujian (*test plan*) yang akan digunakan sebagai dasar untuk mengembangkan kasus uji dan perkiraan waktu yang diperlukan untuk pengujian. Setelah test plan disusun, para pengujian dapat melanjutkan dengan mengembangkan detail kasus uji untuk melaksanakan pengujian.

c. *Test Case Development*

Pada tahap ini, yang merupakan tahap pembuatan dalam siklus STLC, dilakukan pembuatan, verifikasi, dan penyempurnaan butir uji serta skrip pengujian yang telah dipersiapkan pada tahap perencanaan pengujian sebelumnya. Proses dimulai dengan identifikasi data uji yang diperlukan, yang kemudian ditinjau secara menyeluruh untuk memastikan kesesuaian dengan persyaratan yang telah ditetapkan. Setelah itu, data uji tersebut dikerjakan ulang atau diperbaharui sesuai dengan perubahan atau pembaruan yang terjadi dalam persyaratan pengujian. Tahap ini merupakan langkah kunci dalam menyiapkan butir uji dan skrip pengujian yang akan digunakan dalam tahap eksekusi uji berikutnya.

d. *Test environment Setup*

Pada tahap environment setup, fokusnya adalah memastikan bahwa lingkungan pengujian, termasuk baik perangkat keras maupun perangkat lunaknya, berjalan sesuai dengan rencana yang telah ditetapkan. Tahap ini dapat berjalan bersamaan dengan tahap pengembangan kasus uji. Hal yang perlu diperhatikan adalah bahwa tahap ini berdiri sendiri dan tidak tergantung pada tahap-tahap lain dalam siklus pengujian. Jika tim pengembang telah menyiapkan lingkungan pengujian, maka tim pengujian tidak perlu terlibat dalam proses tersebut.

e. *Test Execute*

Tahap ini adalah saat eksekusi pengujian berlangsung berdasarkan *test plan* dan kasus uji yang telah disiapkan sebelumnya. Para pengujian akan menjalankan setiap kasus uji sesuai dengan rencana yang telah ditetapkan. Kasus uji yang berhasil dieksekusi sesuai dengan harapan akan diberi status "*pass*", sementara yang tidak sesuai akan diberi status "*fail*". Setiap temuan bug akan dilaporkan kepada tim pengembang untuk diperbaiki. Pengujian ulang akan

dilakukan setelah perbaikan *bug* dilakukan, untuk memastikan bahwa perbaikan tersebut efektif dan tidak menimbulkan masalah baru.

f. Test Cycle Closure

Test cycle closer merupakan tahap penutupan yang penting dalam STLC. Pada tahap ini, dilakukan serangkaian aktivitas yang bertujuan untuk menyelesaikan proses pengujian dengan baik dan menyajikan hasilnya secara tepat kepada pemangku kepentingan terkait.

Secara keseluruhan, *test cycle closer* adalah tahap penting yang menandai penyelesaian dari seluruh siklus pengujian perangkat lunak. Dengan melakukan aktivitas-aktivitas yang terkait dengan tahap ini secara *komprehensif*, tim pengujian dapat memastikan bahwa hasil pengujian disampaikan dengan akurat dan bahwa pelajaran berharga dapat dipetik untuk meningkatkan proses pengujian di masa mendatang.

2.3 Blackbox Testing

Metode *Black Box* merupakan salah satu pendekatan yang digunakan dalam pengujian aplikasi *mobile*, yang secara khusus berfokus pada evaluasi berbagai aspek seperti tampilan aplikasi, fungsi-fungsi yang tersedia[7], serta kesesuaian alur fungsi dengan proses bisnis yang terkait. Dalam konteks ini, pendekatan *Black Box* memperhatikan bagaimana pengguna akan berinteraksi dengan aplikasi tanpa perlu mengetahui atau memeriksa secara langsung kode sumber program yang ada di baliknya.

Pengujian menggunakan metode *Black Box* tidak melibatkan pemeriksaan detail terhadap *source code* program. Sebaliknya, fokus utamanya adalah pada spesifikasi fungsional dari perangkat lunak itu sendiri. Dengan demikian, para pengujian akan lebih condong untuk mengevaluasi fungsi-fungsi aplikasi, antarmuka pengguna, serta korelasi antara alur kerja aplikasi dengan proses bisnis yang diharapkan.

Studi sebelumnya telah menunjukkan bahwa metode *Black Box testing* memiliki kecenderungan untuk menemukan sejumlah masalah yang umumnya ditemui dalam pengembangan aplikasi. Diantaranya termasuk, tetapi tidak terbatas pada, kesalahan dalam fungsi-fungsi yang diimplementasikan, ketidaksesuaian antarmuka pengguna, kesalahan dalam struktur data dan akses basis data, masalah kinerja, serta kesalahan dalam proses inisialisasi dan terminasi aplikasi.

Dengan demikian, metode *Black Box* merupakan alat yang penting dalam pengembangan aplikasi untuk memastikan kualitas aplikasi *mobile* yang dihasilkan, dengan fokus pada pengalaman pengguna dan kecocokan dengan tujuan bisnis yang diinginkan.

2.4 Automation Testing

Pengujian otomatis atau biasa disebut *Automation testing* merupakan suatu proses yang melibatkan penggunaan skrip atau alat-alat khusus untuk mengotomatisasi eksekusi kasus uji [8]. Berbeda dengan pendekatan manual di mana seorang penguji harus secara langsung menjalankan setiap kasus uji secara berurutan, pengujian otomatis memungkinkan untuk memberikan instruksi langsung kepada aplikasi yang Diuji (AUT) melalui *script* atau alat-alat tersebut, sehingga membebaskan waktu dan sumber daya yang dapat digunakan oleh penguji untuk fokus pada kegiatan yang lebih strategis dan bernilai tambah.

Namun, penting untuk diingat bahwa pengujian otomatis tidak hanya terbatas pada fase eksekusi kasus uji. Meskipun sering kali dianggap sebagai pelaksanaan kasus uji secara otomatis, konsep pengujian otomatis sebenarnya jauh lebih luas daripada itu. Secara keseluruhan, pengujian otomatis dapat dipandang sebagai istilah yang mencakup segala aktivitas pengujian yang dapat diotomatisasi di seluruh siklus pengujian perangkat lunak, mulai dari perencanaan, pembuatan skenario uji, eksekusi, hingga analisis hasil dan pelaporan. Dengan adanya pengujian otomatis yang efektif, organisasi dapat meningkatkan efisiensi, konsistensi, dan akurasi

pengujian, serta mempercepat siklus pengembangan perangkat lunak secara keseluruhan.

2.5 Katalon Studio

Katalon Studio, sebuah aplikasi sumber terbuka yang dikembangkan oleh Katalon LLC, telah menjadi salah satu solusi utama dalam dunia pengujian perangkat lunak secara otomatis [9]. Dirancang untuk memenuhi kebutuhan pengujian lintas *platform*, Katalon Studio memberikan dukungan lengkap untuk tiga jenis pengujian utama: pengujian web, pengujian API, dan pengujian *mobile*. Melalui antarmuka yang intuitif dan fitur yang kaya, pengguna dapat dengan mudah membuat, menjalankan, dan mengelola skenario pengujian yang kompleks.

Salah satu keunggulan utama dari Katalon Studio adalah integrasinya yang luas dengan berbagai teknologi dan *platform external*. Misalnya, pengguna dapat dengan mudah menyinkronkan proyek pengujian mereka dengan alat manajemen uji coba seperti Qtest dan JIRA untuk pelacakan yang efisien dan kolaborasi tim yang lebih baik. Selain itu, integrasi dengan layanan uji coba *cloud* seperti Kobiton memungkinkan pengujian *mobile* yang lebih luas dan terdistribusi secara geografis.

Selain integrasi yang luas, Katalon Studio juga menawarkan fleksibilitas dalam hal penyesuaian dan pengelolaan kode pengujian. Dengan dukungan penuh untuk bahasa *script* seperti Groovy dan kemampuan untuk mengintegrasikan kode dari repositori git seperti GitHub, pengguna memiliki kontrol penuh atas skenario pengujian mereka dan dapat dengan mudah mengotomatiskan alur kerja pengujian mereka sesuai dengan kebutuhan proyek.

Dengan fitur-fitur canggihnya dan dukungan komprehensif untuk berbagai jenis pengujian, Katalon Studio telah menjadi pilihan utama bagi para profesional pengujian perangkat lunak di seluruh dunia. Dengan terus berupaya untuk meningkatkan dan mengembangkan *platform*-nya, Katalon

LLC terus memperkuat posisinya sebagai pemimpin dalam industri pengujian perangkat lunak otomatis.

2.6 Penelitian Terkait

Penulisan tugas akhir ini tidak lepas dari berbagai inspirasi lain dari penelitian- penelitian sebelumnya yang memiliki latar belakang yang sama. Adapun penelitian terkait yaitu pada tabel berikut

Tabel 2. 1 Penelitian terkait

No	Nama dan Tahun	Judul	Topik	Persamaan/Perbedaan
1	Arief Zulianto, Ayi Purbasari, Neni Suryani, Ari Indra Susanti, Fedri R. Rinawan, Wanda G. Purnama. 2021	JEPIN (Jurnal Edukasi dan Penelitian Informatika) Pemanfaatan Katalon Studio untuk Otomatisasi Pengujian <i>Black-Box</i> pada Aplikasi iPosyandu	<i>Software Quality Assurance</i>	Persamaan: menggunakan tools katalon studio dan metode blackbox testing Perbedaan: menggunakan framework groovy saat pembuatan script automasi
2	Fityan Ardi, Hanson Prihantoro Putro. 2021	Pengujian <i>Black Box</i> Aplikasi <i>Mobile</i> Menggunakan Katalon Studio (Studi Kasus: ACC Partner PT. Astra Sedaya Finance)	<i>Software Quality Assurance</i>	Persamaan : menggunakan tools katalon studio dan metode blackbox testing Perbedaan : menggunakan framework groovy saat pembuatan script automasi
3	Herlinda, Dona Katarina, Erlin Windia Ambarsari 2019	Automation Testing Tool Dalam Pengujian Aplikasi Belajar Tajwid Pada Platform Android	<i>Automation testing</i>	Persamaan : menggunakan tools katalon studio dan metode blackbox testing Perbedaan : menggunakan framework groovy saat pembuatan script automasi
4	Annisa Indrayanti, Bheta Agus Wardijono dan Nur Rizky Aulia 2021	Analisis Pengujian Graphical User Interface E-Commerce Dengan Menggunakan Katalon Studio	<i>Automation testing</i>	Persamaan:menggunakan tools katalon studio Perbedaan:melakukan pengujian pada aplikasi android, menggunakan meted blackbox testing
5	Wahyu Nur Cholifah, Yulianingsih, Sri Melati Sagita. 2018	Pengujian Black Box Testing Pada Aplikasi Action & Strategy Berbasis Android Dengan Teknologi Phonegap	<i>Software Quality Assurance</i>	Persamaan : menggunakan metode blackbox testing Perbedaan : mengimplementasikan uji automasi menggunakan tools katalon studio

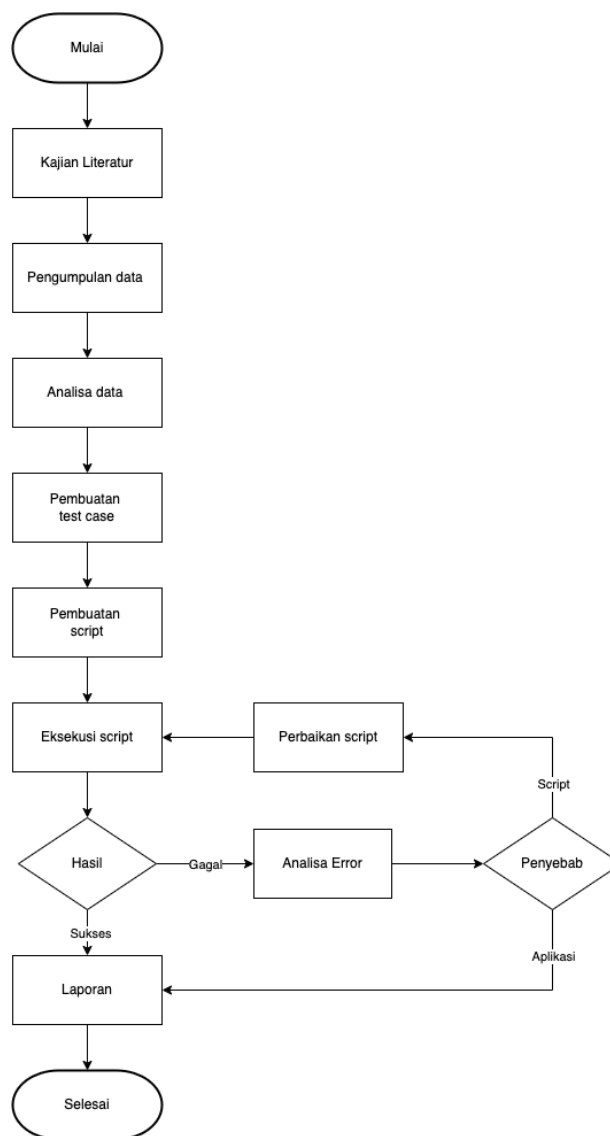
Pada table 2.1 berisi beberapa penelitian yang memiliki topik sama yaitu *software testing*. Pada penelitian nomor 1 Penerapan alat Katalon Studio untuk mengotomatisasi proses pengujian aplikasi iPosyandu bertujuan untuk meminimalisir kekurangan yang tidak dapat dihindari oleh pengujian manual serta mengurangi risiko kesalahan manusia. *Test case* yang digunakan dalam pengujian menggunakan metode *record dan playback* dengan Katalon Studio[3], Penelitian nomor 2 [7] mengenai pengujian automasi kepada aplikasi ACC *Partner* menggunakan *tool* katalon studio, pengujian yang dilakukan merupakan pengujian *black box* dimana yang diuji hanya pada rana fungsionalitas aplikasi tidak sampai ke *coding*, dan penelitian nomor 3 [10] mengenai pengujian automasi kepada aplikasi belajar tajwid menggunakan *tool* katalon studio, dan di penelitian ini juga menggunakan metode *blackbox testing*. Penelitian-penelitian memiliki persamaan menggunakan katalon sebagai *tool* pengujian dan *blackbox testing* sebagai metode pengujian hal tersebut serupa dengan penelitian yang disusun peneliti perbedaannya pada penelitian ini peneliti menggunakan *framework groovy* untuk pembuatan *script automation*

Pada penelitian nomor 4 [11] mengulas analisis waktu respon antarmuka pengguna (GUI) dari dua *e-commerce* yang berbeda. Pengujian dilakukan dari akses beranda hingga tahap pemesanan produk. Meski sama-sama menggunakan metode pengujian *blackbox*, perbedaannya adalah *platform* yang diuji: *website e-commerce* untuk satu *platform* dan aplikasi *mobile* Dunia Games untuk yang lainnya. Penelitian terakhir [12] Pengujian dilakukan pada aplikasi *action & strategi* dalam dua tahap, pertama uji fungsi aplikasi *android* oleh sejumlah pengguna dan dilakukan secara bersamaan. pengujian kedua uji fungsi web admin oleh seorang pengguna dan pada kedua hasil pengujian tidak ditemukan adanya kesalahan pada sistem. sama sama menggunakan *black box testing* dan melakukan pengujian pada aplikasi *mobile* perbedaannya penelitian ini dilakukan secara manual/manual testing sedangkan pada penelitian yang disusun peneliti menerapkan uji automasi/*automation testing* menggunakan *tool* katalon studio.

BAB III METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian ini menjelaskan mengenai proses dilaksanakannya penelitian mulai dari proses pengumpulan data, analisa, pembuatan *test case*, pembuatan *script automation* dan *reporting* hasil pengujian yang dapat dilihat pada gambar dibawah ini :



Gambar 3. 1 Tahapan Penelitian

Berikut Penjabaran dari gambar 3.1 tahapan dalam melakukan penelitian ini

3.1.1 Kajian Literatur

Pada tahapan ini peneliti melakukan pengkajian teori-teori terkait penelitian tersebut melalui beberapa sumber seperti jurnal penelitian, artikel *website* dan media lain.

3.1.2 Pengumpulan Data

Pada tahap ini peneliti melakukan observasi pada dokumen *test case* dari penelitian lain untuk mendapatkan referensi pembuatan *test case*, selain itu peneliti juga mengobservasi langsung pada aplikasi dunia games untuk memperjelas alur kerja aplikasi.

3.1.3 Analisa Data

Analisa dilakukan pada data yang diperoleh dari proses *review* dokumen *test case* dan observasi pada aplikasi Dunia Games, data yang diperoleh berupa alur dari fitur yang ada pada Dunia Games.

3.1.4 Pembuatan *Test case*

Skenario yang telah ditentukan sebelumnya akan dibuat menjadi *test case*, *test case* merupakan pengembangan dari skenario yang isinya akan mempermudah pengujian biasanya *test case* berisi nama fitur, deskripsi, *type*, prakondisi, *step*, *expected result*, *actual result*, dan status.

3.1.5 Pembuatan *Script*

Pembuatan *script* dilakukan dengan acuan *test case* yang telah dibuat sebelumnya, pembuatan *script* dilakukan menggunakan katalon studio dan Bahasa pemrograman java, dalam pembuatan *script automation* memiliki beberapa tahap antara lain :

1) Menangkap objek aplikasi

Dalam implementasi automation memerlukan objek sebagai *selector* pada *script*, objek berupa elemen yang ada pada aplikasi seperti *button*, *form* dan lain sebagainya, pada setiap elemen tersebut perlu dipastikan

memiliki identitas masing-masing agar *tool spy object* dapat menangkapnya, umumnya identitasnya berupa *testid / Xpath*.

2) Coding script

Setelah memiliki objek yang dibutuhkan, langkah selanjutnya adalah *coding script* berdasarkan *test case* yang telah dibuat sebelumnya, pada tahap ini *script* harus dipastikan mengikuti *step* dari setiap *test case*.

3) Create Test suite

Setelah script terbuat langkah terakhir adalah membuat *test suite*, hal ini dilakukan untuk persiapan eksekusi *script*.

3.1.6 Eksekusi Script

Tahap yang merupakan inti dari penelitian ini, eksekusi *script* dilakukan setelah pembuatan *script* selesai, pada tahap ini semua *script* yang telah dibuat akan dijalankan untuk memperoleh hasil pengujian.

3.1.7 Evaluasi Hasil

Hasil yang didapatkan dari eksekusi script perlu dilakukan evaluasi untuk mengetahui apakah terjadi *error* dan penyebabnya.

3.1.8 Analisa Error

Tahap ini dilakukan ketika terjadi *error* saat melakukan eksekusi *script*, dalam Analisa *error* dibagi menjadi 2 kategori :

1) *Error script*

Jika ditemukan *error* terjadi karena kesalahan penulisan script maka langkah selanjutnya adalah memperbaiki *script* dan eksekusi ulang.

2) *Error aplikasi*

Error aplikasi merupakan faktor umum yang dapat menghentikan eksekusi, biasanya berupa data tidak termuat, aplikasi *force closed* dan lain sebagainya, jika ditemukan demikian Langkah selanjutnya adalah pelaporan ke yang bertanggungjawab dalam hal ini ada *developer* aplikasi.

3.1.9 Laporan

Setelah mendapatkan hasil dari pengujian, langkah selanjutnya adalah membuat laporan hasil pengujian, laporan ini biasanya berisi jumlah *test case* yang teruji, status dari setiap *test case* yang diuji, *evidence* / bukti pengujian, dan *error* (jika terjadi), dalam *automation testing* laporan akan di-generate secara otomatis lengkap dengan *evidence* dan waktu pengujian.

3.2 Rancangan Penelitian

3.2.1 Jenis Penelitian

Penelitian ini merupakan penelitian evaluatif yaitu mengukur dan menentukan hasil dari rencana atau proyek. Pendekatan tersebut melibatkan proses diperolehnya data dari hasil observasi dari hasil eksekusi script automation pada 12 skenario yang telah ditentukan sebelumnya pada aplikasi Dunia Games. *Output* dari penelitian ini adalah hasil dari eksekusi script baik yang berhasil dieksekusi maupun yang gagal dieksekusi, data tersebut menjadi bahan evaluasi tentang implementasi automasi dalam pengujian aplikasi.

3.2.2 Metode Pengumpulan Data

Adapun pengumpulan data dilakukan dengan tahap-tahap sebagai berikut :

1. Studi Literatur

Pada tahap ini peneliti mencari referensi jurnal dan penelitian yang masih memiliki topik yang berkaitan dengan penelitian yang bertujuan untuk memperdalam pengetahuan peneliti terkait permasalahan yang diteliti data yang didapat berupa persamaan dan perbedaan antara penelitian ini dan penelitian lain yang terkait.

2. Observasi

Peneliti melakukan observasi mendalam pada hasil eksekusi script automation menggunakan Katalon Studio yang diterapkan pada aplikasi Dunia Games. Observasi ini bertujuan untuk mengumpulkan data yang komprehensif mengenai performa dan efektivitas script automation yang telah dibuat. Dalam proses observasi tersebut, peneliti menganalisis

jumlah test case yang berhasil dieksekusi serta jumlah test case yang gagal dieksekusi. Selain itu, peneliti juga mencatat secara rinci penyebab kegagalan eksekusi pada setiap test case yang gagal.

3.2.3 Metode Analisa

Metode yang digunakan dalam analisa data pada penelitian ini adalah kuantitatif, Metode kuantitatif adalah jenis metode yang mengandalkan data numerik untuk menganalisis fenomena dan menjawab pertanyaan penelitian[13]. Pada penelitian ini Analisa kuantitatif dilakukan pada hasil dari eksekusi *script automation* menggunakan tool katalon studio, hasil tersebut akan dijumlahkan serta dihitung dalam bentuk *pesentase* untuk mengukur besar keberhasilan *script automation* menggunakan tool katalo studio dijalankan dalam penelitian ini.

3.2.4 Metode Pengujian

Pada penelitian ini, peneliti melakukan pengujian pada *script* dengan cara mengeksekusi langsung script yang telah dibuat menggunakan alat Katalon Studio. Langkah-langkah pengujian ini dirancang untuk memastikan bahwa skrip bekerja sesuai dengan spesifikasi yang telah ditentukan sebelumnya dan memberikan hasil yang akurat serta dapat diandalkan. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam peningkatan metodologi pengujian perangkat lunak, khususnya dalam penggunaan alat otomatisasi seperti Katalon Studio.

3.3 Lingkungan Penelitian

Peneliti melakukan penelitian ini secara langsung pada lingkungan kerja di PT.Sigma Cipta Caraka (Telkomsigma) bergerak dibidang komunikasi dan Inovasi di dunia IT yang beralamat di Telkom *Landmark Tower*, Jl. Gatot Subroto No.Kav. 52, RT.6/RW.1, *West Kuningan*, Mampang Prapatan, *South Jakarta City*, Jakarta 12710.

3.4 Alat dan Bahan

Adapun alat dan bahan yang digunakan peneliti dalam menyusun penelitian ini sebagai berikut :

A. Perangkat Lunak

Perangkat lunak yang digunakan untuk membuat implementasi Automation testing pada penelitian antara lain:

- 1) Aplikasi Dunia Games yang digunakan sebagai objek penelitian implementasi *automation testing*.
- 2) Katalon Studio sebagai alat pembuatan *script automation*.
- 3) Appium Server digunakan untuk mengkonfigurasi *device android* ke Katalon studio.
- 4) Spreadsheet digunakan untuk membuat *Test case* yang akan diimplementasikan ke *automation testing*.
- 5) Google chrome digunakan untuk mengakses *spreadsheet, google form, dan report* hasil *automation testing*.
- 6) Sistem Operasi yang digunakan untuk melakukan penelitian ini adalah macOS Big Sur versi 11.7.1.

B. Perangkat Keras

Perangkat yang digunakan untuk melakukan penelitian ini antara lain:

- 1) Laptop:

Model	: Macbook Pro 2015
Processor	: Intel core i7
RAM	: 16 GB
SSD	: 256 GB

- 2) Handphone Android :

Merk	: Infinix
OS Versi	: Android 11
RAM	: 6 GB
Memori	: 128 GB

BAB VI

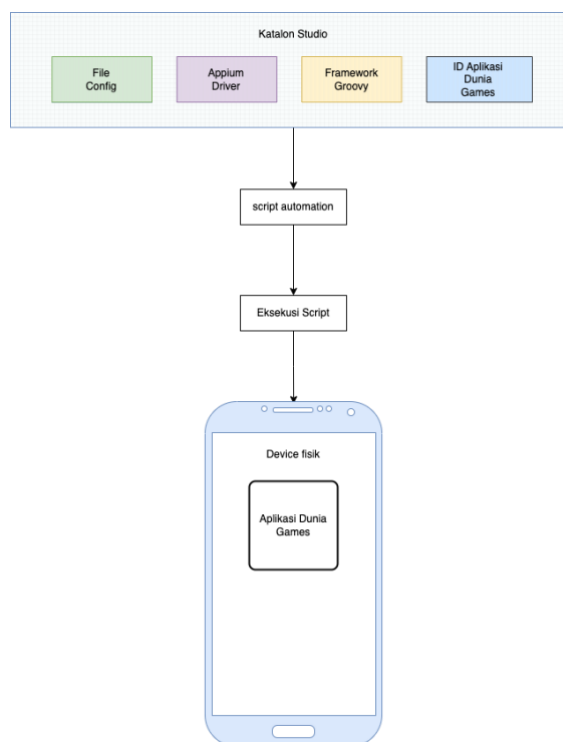
HASIL PEMBAHASAN

4.1 Analisa Perancangan

Pada bab sebelumnya, telah dijelaskan mengenai metodologi penelitian dan studi literatur dari penelitian yang terkait untuk mendukung penelitian ini. Kini, penelitian telah memasuki tahap yang krusial, yaitu tahap pelaksanaan. Dalam pelaksanaan penelitian ini terdiri dari beberapa tahap sebagai berikut :

4.1.1 Perancangan Sistem

Pada tahap ini peneliti melakukan perancangan dari sistem yang akan dibangun yang dalam penelitian ini berupa *automation testing*, berikut diagram dari alur *automation testing* yang telah di rancang:



Gambar 4. 1 Perancangan *Automation testing*

Berikut penjelasan dari setiap komponen yang ada pada gambar 4.1 beserta dengan fungsinya :

1) Katalon Studio

a) *File Config*

Berupa *file* yang dibutuhkan untuk menjalankan *script automation testing* termasuk *global variable*, *testcase*, *object library* dan lain-lain.

b) *Appium driver*

Library Package yang dibutuhkan untuk menghubungkan *script automation* dengan *device mobile*, Appium driver akan otomatis terinstall pada *device mobile* ketika eksekusi *script automation* berlangsung

c) *Framework groovy*

Merupakan kerangka kerja *automation testing* yang umum digunakan pada *tool* katalon. Berbahasa pemrograman Java, *framework* ini cukup mudah untuk dipahami hanya perlu pengetahuan dasar mengenai bahasa pemrograman java.

d) ID aplikasi Dunia Games

Merupakan identitas dari aplikasi yang akan diuji yaitu Dunia Games, ID aplikasi diperlukan agar *script* dapat mengidentifikasi aplikasi yang akan dijalankan oleh *script*-nya.

2) *Script Automation*

Script automation dibuat dalam Bahasa java dengan kerangka kerja *groovy*, *script automation* dibuat dari *test case* yang merupakan rincian dari skenario *test* yang telah ditentukan pada Batasan masalah pada penelitian ini.

3) Eksekusi *Script Automtaion*

Eksekusi *script* dilakukan langsung pada katalon studio, sebelum itu *script test case* dimasukan dan dieksekusi di folder *test suite*, sehingga hasil dan *report testing* terbuat secara otomatis.

4) *Device* fisik

a) Aplikasi Dunia Games

Pelaksanaan penelitian ini menggunakan *device* fisik berupa *handphone android* dengan spesifikasi OS versi 11, memori 6 /128 GB. Pada *device* sudah harus ter-*install* aplikasi Dunia Games yang akan diujiakan.

4.1.2 Pembuatan dan Analisa Testcase

Pada bab 1 pembatasan sebelumnya sudah disebutkan skenario yang akan dilakukan implementasi *automation testing*, yaitu ada 12 skenario yang mewakili fitur utama dari aplikasi Dunia Games, selain itu 12 skenario tersebut dipilih dengan karakteristik dan step yang beragam untuk menambah eksplorasi *automation testing* pada penelitian ini.

Langkah selanjutnya adalah mengembangkan skenario tersebut menjadi lebih rinci mulai dari prakondisi, tes step dan ekspektasi hasil yang bisa disebut sebagai *Test case*. *Test Case* adalah rangkaian skenario yang dibuat oleh QA untuk memastikan bahwa sistem yang diuji memenuhi persyaratan dan standar tertentu serta berfungsi dengan baik[14]. gunanya untuk mempermudah melakukan pengujian atau dalam kasus penelitian ini mempermudah implementasi ke *automation testing script*. Berikut hasil dari pengembangan skenario menjadi *test case*:

Tabel 4. 1 *Test case*

<i>Test case</i>					
<i>ID</i>	TC001	<i>Feature</i>	Registration	<i>Type</i>	Negative
<i>Name</i>	Fail to registration				
<i>Desc</i>	As user fail to regist new account (email or password invalid, wrong otp)				
<i>Precondition</i>	Non login, had a internet connection				
<i>Step</i>	1. Open Dunia Games application 2. Click profile menu on bar menu 3. Click icon mail 4. Click tab Register 5. At email form input incorrect format email 6. Input incorrect format password (not contains capital alphabet, number and character) 7. Input incorrect OTP				
<i>Expected result</i>	*can view several error like: - wrong format email -password not match -wrong otp *button register disable				

ID	TC002	Feature	Registration	Type	Positive
Name	Successfully registration new account				
Desc	As user success to registration new account				
Precondition	Non login, had a internet connection				
Step	<ol style="list-style-type: none"> 1. Open Dunia Games application 2. Click profile menu on bar menu 3. Click icon mail 4. Click tab register 5. At email form input correct format email 6. Click register button 7. Input correct format password (contains capital alphabet, 				
Expected result	*success to register and direct to homepage				
ID	TC003	Feature	Login	Type	Negative
Name	Success to login				
Desc	As user fail to regist new account (email or password invalid, wrong otp)				
Precondition	Non login, had a internet connection				
Step	<ol style="list-style-type: none"> 1. Open Dunia Games appilcation 2. Click profile menu on menu bar 3. Click icon mail 4. Input email 5. Click button login 6.input incorrect password 7. Click continue button 				
Expected result	*can see toast error wrong email or password •failed to login				
ID	TC004	Feature	Login	Type	Positive
Name	Fail to login				
Desc	As user can input correct credential				
Precondition	Non login, had a internet connection				
Step	<ol style="list-style-type: none"> 1. Open Dunia Games appilcation 2. Click profile menu on menu bar 3. Click icon mail 4. Input correct email 5. Click button login 6.input correct password 7. Click continue button 				
Expected result	*successfull login *direct to homepage				
ID	TC005	Feature	Article	Type	Positive
Name	Like and bookmark the article				
Desc	User can like and bookmark the article				
Precondition	Had a internet connection				
Step	<ol style="list-style-type: none"> 1. Open Dunia Games application and do login 2. Tap article submenu 3. Tap bar one of the article from list 4. Tap button like 5. Tap button bookmark 				
Expected result	*button like will active *button bookmark will active				
ID	TC006	Feature	Gamefeed	Type	Positive
Name	Create post in gamefeed				
Desc	User can create post				
Precondition	Had a internet connection				
Step	<ol style="list-style-type: none"> 1. Open Dunia Games application and do login 2. Tap Menu game 3. Tap one of game 				

	4. Tap post tab 5. Tap create post 6. Write post in post field 7. Tap submit				
Expected result	*user succesfully create the post				
ID	TC007	Feature	Gamefeed	Type	Positive
Name	Like and comment one of the post				
Desc	User can like and comment the post				
Precondition	Had a internet connection				
Step	1. Open Dunia Games application and do login 2. Tap Menu game 3. Tap one of game 4. Tap post tab 5. Tap like button from one of the post 6. Write comment in comment section 7. Tap send button				
Expected result	*user succesfully like and comment the post				
ID	TC008	Feature	Tournament	Type	Positive
Name	Create tournament				
Desc	User can create Tournament				
Precondition	Had a internet connection				
Step	1. Open Dunia Games application and do login 2. Tap tournamet submenu 3. Tap create tournament button 4. Input all field 5. Input all field setting tournament 6. Tap submit 7. View detail tournamet page				
Expected result	*Make sure there are several tab like : - Highlight - Participant - Watch * there button join tournament *there registration date				
ID	TC009	Feature	Room chat	Type	Positive
Name	Create roomchat				
Desc	User can create roomchat				
Precondition	Had a internet connection				
Step	1. Open Dunia Games application and do login 2. Tap roomchat menu 3. Tap create roomchat 4. Input name room chat 5. Choose games 6. Set member qouta 7. Set room as privat/general 8. Tap submit button				
Expected result	User succesfully create room chat				
ID	TC010	Feature	Room chat	Type	Positive
Name	Send message to one of room chat				
Desc	User can send message to one of room chat				
Precondition	Had a internet connection				
Step	1. Open Dunia Games application and do login 2. Tap roomchat menu 3. Tap one of room chat 4. Input message inf chat field 5. Tap send				

Expected result	*user succesfully send message to roomchat				
ID	TC011	Feature	Topup	Type	Negative
Name	Fail to topup diamond				
Desc	User fail to topup diamond game				
Precondition	Had a internet connection				
Step	1. Open Dunia Games application and do login 2. Tap Menu Topup 3. Tap one of game voucher 4. Input wrong id games 5. Input wrong phone number 6. Input wrong OTP				
Expected result	*button buy disable *can view several error message like: -wrong id games -wrong phone number				
ID	TC012	Feature	Topup	Type	Positive
Name	Topup diamond one of games				
Desc	User can topup diamond				
Precondition	Had a internet connection				
Step	1. Open Dunia Games application and do login 2. Tap Menu Topup 3. Tap one of game voucher 4. Input id games 5. Input phone number 6. Input Email (optional) 7. Tap buy button 8. Tap confirmation button				
Expected result	User direct to success payment page				

4.2 Implementasi dan Evaluasi

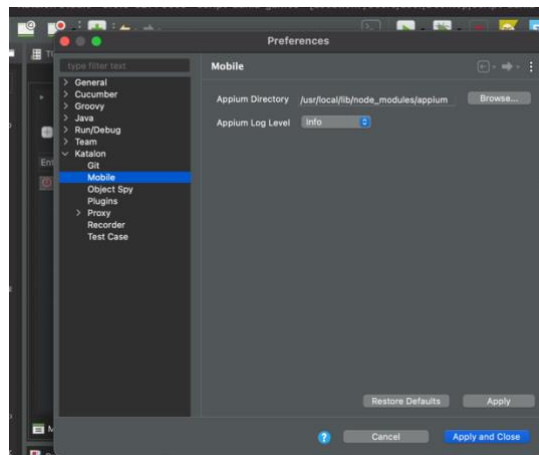
Pada subab ini akan menjelaskan secara rinci implementasi script automation pada setiap *Test case* dan evaluasi dari hasil pengujian *Automation testing* yang dijalankan.

4.2.1 Setup Environment Automation Testing

Setup environment yang dimaksud adalah konfigurasi yang dilakukan pada *device* yang digunakan untuk membuat automation testing, Berikut adalah hal-hal yang perlu disiapkan untuk membuat *script automation*:

- 1) Program Katalon Studio, dapat diunduh secara gratis di website official katalon pada link berikut <https://katalon.com/download>, peneliti menyarankan untuk menggunakan katalon studio versi 8.6.9 karena sudah stabil, setelah terinstall lakukan login pada katalon studio menggunakan akun yang telah dibuat sebelumnya, jika belum punya dapat membuat akun pada website katalon.

- 2) Nodejs, diperlukan untuk menginstall *library package* yang diperlukan untuk kebutuhan *testing*
- 3) Appium *package*, dapat diinstall menggunakan *manager package* dari node js yaitu NPM, versi yang direkomendasikan oleh peneliti yaitu Appium versi 1.22.0 karena cocok dengan versi katalon yang telah diberitahukan sebelumnya, untuk penginstallan dapat menggunakan perintah “`npm i -g appium@1.22.0`”, setelah ter-*install* perlu dilakukan konfigurasi, pergi ke menu katalon studio => *preferences* => katalon => *mobile* tampilannya akan seperti pada gambar dibawah ini :

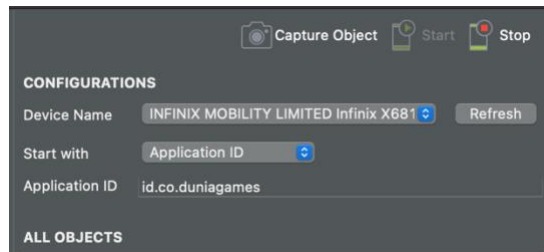


Gambar 4. 2 Appium Directory

Setelah itu sesuaikan direktor dengan direktori Appium package yang telah terinstall, setelah itu setup *environment* telah selesai.

4.2.2 Implementasi Script Automation

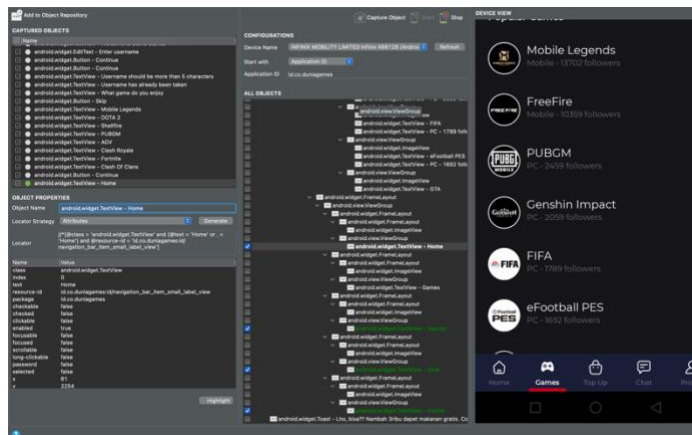
Langkah pertama dalam implementasi yaitu menangkap seluruh objek yang dibutuhkan *script automation* dari fitur aplikasi Dunia games, peneliti menggunakan tool dari katalon studio yaitu *Object Spy Mobile*. Untuk menggunakannya diperlukan beberapa dua hal yaitu *device* yang terhubung dan id aplikasi dari Dunia Games, id aplikasi digunakan untuk mengenali aplikasi yang akan di tangkap objeknya, untuk konfigurasinya dapat dilihat pada gambar berikut



Gambar 4. 3 Konfigurasi *Object Spy Mobile*

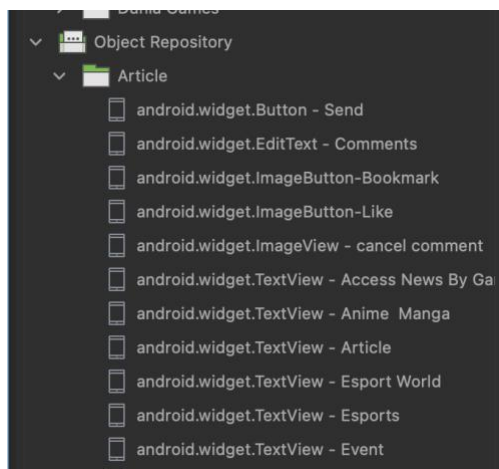
Dapat dilihat pada gambar ada beberapa tombol yang memiliki fungsi yang berbeda-beda, ada tombol *start* untuk memulau tangkap objek, tombol *stop* untuk menghentikan tangkap objek, tombol *capture object* untuk menangkap objek pada halaman yang baru diakses pada aplikasi dan yang terakhir tombol *refresh* untuk mengembalikan konfigurasi ke kondisi semula.

Langkah selanjutnya klik tombol *start* untuk memulai, akan muncul halaman aplikasi pada bagian *device view*, umumnya yang akan muncul adalah halaman yang terakhir diakses pada aplikasi jika aplikasi belum terbuka maka akan muncul secara *default* halaman *home*, selain *device view* ada bagian *all object* yang akan menampilkan semua objek di halaman yang sedang muncul pada *device view*, untuk menangkap objek-objek tersebut klik *check box* yang ada pada ujung list objek, jika sudah ke checklist maka object akan secara otomatis tertangkap dan muncul di bagian *captured object*, jika ingin menangkap objek di halaman lain, arahkan aplikasi pada device ke halaman yang ingin ditangkap setelah itu klik tombol *capture object* untuk menangkap ulang objek maka, list objek pada bagian *all object* akan berubah sesuai dengan halaman yang baru diakses.



Gambar 4. 4 list objek dari halaman aplikasi

Setelah objek yang diperlukan tertangkap, langkah selanjutnya menyimpan objek ke *repository object*, disarankan untuk diletakan difolder sesuai dengan nama fitur agar mudah dalam penggunaannya ke script.



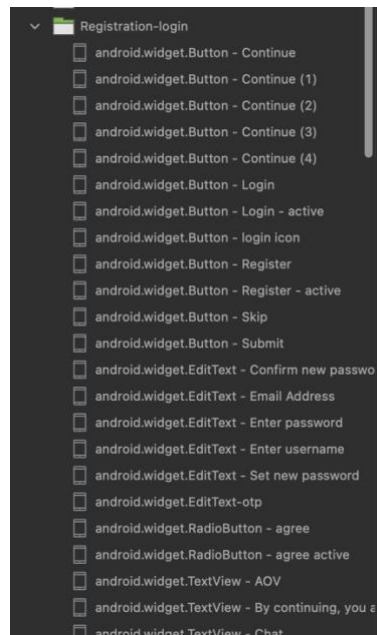
Gambar 4. 5 Objek yang tersimpan di *Object Repository*

Setelah Objek-objek yang diperlukan telah ditangkap, maka proses pembuatan *script automation* bisa dilakukan, berdasarkan jumlah *Test case* yang telah dibuat berikut rincian implementasi dari 15 Test case tersebut:

1) *Fail to registration*

Pada *test case* ini pengguna menjalankan skenario gagal melakukan registrasi tujuannya untuk mengetahui *error message*

yang akan didapat pada setiap halaman, sebelumnya objek yang dibutuhkan untuk menjalankan skenario ini sudah ditangkap, dapat di lihat pada gambar 4.6 berikut.



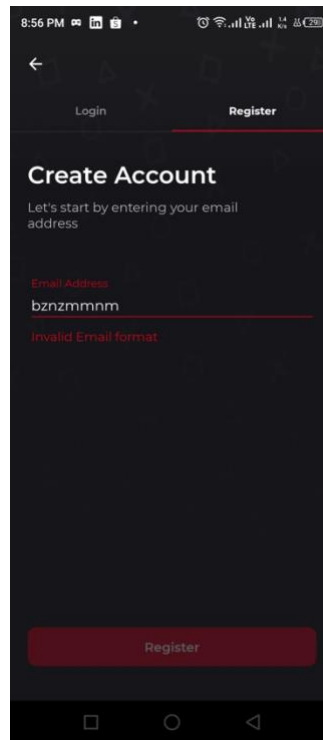
Gambar 4. 6 Objek fitur registrasi dan login

```
64
65 def TC001(String tcid, def device_height, def device_width) {
66     Mobile.comment(tcid + ' - Start')
67
68     Mobile.comment(tcid + ' - Expected Result - user can see wrong format email error message ')
69
70     String TC = 'TC001'
71
72     try {
73         int atc = 1
74
75         Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
76
77         atc = atc + 1
78
79         Mobile.tap(findTestObject('Object Repository/Navbar/android.widget.ImageView - Profile'), 0)
80
81         Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - login icon'), 0)
82
83         Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.TextView - Register'), 0)
84
85         Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Email Address'), 'cgshbbjhsj', 0)
86
87         Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
88
89         atc = atc + 1
90
91         Mobile.pressBack()
92     }
```

Gambar 4. 7 TC001 - *Input email* dengan format yang salah

Berdasarkan potongan kode yang terlihat pada Gambar 4.7 di atas, peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input* email dengan format yang salah. *method tap()* pada

line 59, 61 dan 63 mensimulasikan proses pengarahaan ke halaman registrasi yang kemudian dilanjutkan dengan *method setText()* yang akan meng-*input* nilai yang ada pada parameter yang berupa kata random, hal ini akan men-*trigger message error* dapat dilihat pada gambar dibawah ini:



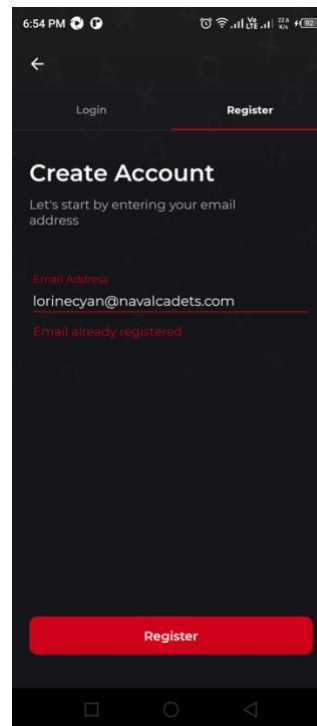
Gambar 4. 8 TC001_evidence1

```
72 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - login icon'), 0, FailureHandling.STOP_ON_FAILURE)
73
74 Mobile.click(findTestObject('Object Repository/Registration-login/android.widget.TextView - Register'), 0, FailureHandling.STOP_ON_FAILURE)
75
76 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Email Address'), 'dangngn1@gmail.com', FailureHandling.STOP_ON_FAILURE)
77
78 Mobile.longPress(findTestObject('Object Repository/Registration-login/android.widget.Button - Register'), 5, FailureHandling.STOP_ON_FAILURE)
79
80 Mobile.waitForElementPresent(findTestObject('Object Repository/Registration-login/android.widget.TextView - Email already registered'), 10, FailureHandling.STOP_ON_FAILURE)
81
82 Mobile.takeScreenshot(RuntimeFunction.getReportFolder() + '\\'+TC+'-setec'+img', FailureHandling.STOP_ON_FAILURE)
83
84 etc = etc + 1
85
```

Gambar 4. 9 TC001- *Input email* yang sudah ter-registrasi

Selanjutnya peneliti akan mensimulasikan langkah-langkah meng-*input email* yang telah ter-registrasi, terlihat potongan kode pada Gambar 4.9 di atas *method tap()* pada line 72 dan 73 mensimulasikan proses pengarahaan ke halaman registrasi yang kemudian dilanjutkan dengan *method setText()* yang akan meng-*input* nilai pada parameter yang berupa email yang telah terdaftar

kemudian dilanjutkan *method longPress* yang akan menekan tombol registrasi, hal ini akan *trigger message error* dapat dilihat pada gambar dibawah ini:



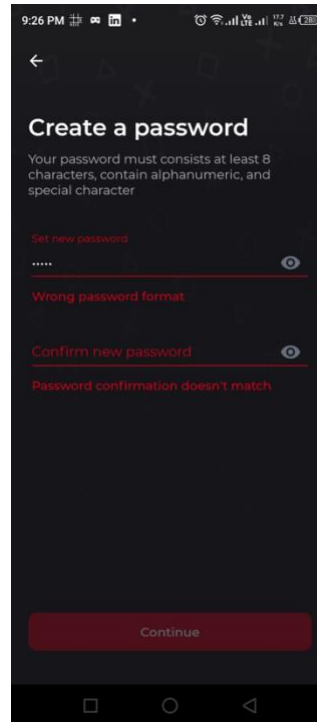
Gambar 4. 10 TC001_evidence2

```
98
99
100
101
102
103
104
```

```
Mobile.setText(findTestObject("Object Repository/Registration-Login/android.widget.EditText - Set new password"), "12344", false, false, true, STOP_ON_FAILURE)
Mobile.verifyElementExist(findTestObject("Object Repository/Registration-Login/android.widget.TextView - Wrong password format"), 0, false, false, true, STOP_ON_FAILURE)
Mobile.verifyElementExist(findTestObject("Object Repository/Registration-Login/android.widget.TextView - Password confirmation doesn't match"), 0, false, false, true, STOP_ON_FAILURE)
```

Gambar 4. 11 TC001- *Input password* yang formatnya salah

Setelah itu dilanjutkan langkah-langkah meng-input password yang tidak sesuai dengan kriteria, dapat dilihat pada potongan kode Gambar 4.11 di atas. *method setText()* pada *line 99* akan meng-input nilai ada pada parameter yang berupa kata random, kemudian *method verifyElementExist()* pada *line 101* dan *102* akan memverifikasi *message error* yang muncul pada kedua *field*, dapat dilihat pada gambar dibawah ini:



Gambar 4. 12 TC001_evidence3

```

116 Mobile.longPress(findTestObject('Object Repository/Registration-login/android.widget.Button - Continue'), 5, FailureHandling.STOP_ON_FAILURE)
117
118 Mobile.verifyElementExist(findTestObject('Object Repository/Registration-login/android.widget.TextView - Enter Verify Code'), 0, FailureHandling.STOP_ON_FAILURE)
119
120 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText-otp'), '8123', 0, FailureHandling.STOP_ON_FAILURE)
121
122 Mobile.longPress(findTestObject('Object Repository/Registration-login/android.widget.Button - Continue'), 5, FailureHandling.STOP_ON_FAILURE)
123
124 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
125
126 atc = atc + 1
127

```

Gambar 4. 13 TC001 - Input kode OTP yang salah

Langkah terakhir adalah mensimulasikan meng-*input* OTP yang tidak valid, dapat dilihat pada potongan script gambar 4.13 diatas, *method longPress* akan mejalankan step menekan tombol *continue* yang akan mengarahkan ke halaman verifikasi OTP lalu akan diverifikasi oleh *method verifyElementExist()* untuk memastikan aplikasi sudah berada di halaman tersebut, selanjutnya *method setText()* pada line 121 akan meng-*input* nilai dari parameter yang berupa nomor random, hal ini akan men-trigger *message error* yang dapat dilihat pada gambar dibawah ini:



Gambar 4. 14 TC001_evidence4

2) *Successfully registration new account*

Pada test case ini user menjalankan skenario berhasil melakukan registrasi tujuannya untuk mengetahui fitur registrasi berjalan dengan benar dan setelah registrasi akan diarahkan kehalaman home.

```

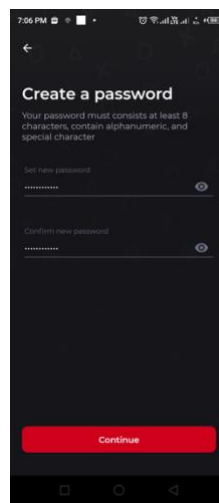
59 Mobile.tap(findTestObject('Object Repository/Navbar/android.widget.ImageView - Profile'), 0, FailureHandling.STOP_ON_FAILURE)
60 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - Login Icon'), 0, FailureHandling.STOP_ON_FAILURE)
61 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.TextView - Register'), 0, FailureHandling.STOP_ON_FAILURE)
62 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Email Address'), 'wrongavi@navalcadets.com', 0, FailureHandling.STOP_ON_FAILURE)
63 Mobile.longPress(findTestObject('Object Repository/Registration-login/android.widget.Button - Register'), 5, FailureHandling.STOP_ON_FAILURE)
64 Mobile.verifyElementExist(findTestObject('Object Repository/Registration-login/android.widget.TextView - Create a password'), 0, FailureHandling.STOP_ON_FAILURE)
65 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Set new password'), 'Tidakapa123', 0, FailureHandling.STOP_ON_FAILURE)
66 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Confirm new password'), 'Tidakapa123', 0, FailureHandling.STOP_ON_FAILURE)
67 Mobile.takeScreenshot(OutConfDuration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
68 atc = atc + 1
69

```

Gambar 4. 15 TC002 - *Input email dan password yang benar*

Berdasarkan potongan kode yang terlihat pada Gambar 4.15 di atas, peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input* email dan password yang benar, pada *line* 59-63 merupakan *method tap()* yang akan mengarahkan aplikasi ke

halaman registrasi kemudian line selanjutnya ada *method setText()* meng-input nilai yang telah dimasukan dalam parameter yang berupa email yang belum pernah melakukan registrasi setelah itu method tap akan menekan tombol *continue* yang akan mengarahkan ke halama *create password*, method *verifyElementExist()* pada line 69 akan memverifikasi objek yang ada pada halaman create password untuk memastikan bahwa aplikasi benar diarahkan ke halaman *create password* dapat dilihat pada gambar dibawah



ini:

Gambar 4. 16 TC002_evidence1

```

79 Mobile.longPress(findTestObject('Object Repository/Registration-Login/android.widget.Button - Continue'), 0, FailureHandling.STOP_ON_FAILURE)
80
81 Mobile.verifyElementExist(findTestObject('Object Repository/Registration-Login/android.widget.TextView - Enter Verify Code'), 0, FailureHandling.STOP_ON_FAILURE)
82
83 Mobile.delay(10)
84
85 Mobile.takeScreenshot(Configurations.getReportFolder()+'\\'+TC+'_'+otc+'.png', FailureHandling.STOP_ON_FAILURE)
86
87 otc = otc + 1

```

Gambar 4. 17 TC002 - Input OTP

Lalu dilanjutkan langkah-langkah meng-input OTP yang *valid* namun setelah dianalisa test case ini dikategorikan *non feasible* atau tidak bisa di otomatisasi karena memerlukan kode OTP yang valid dari pihak ke tiga, Berdasarkan potongan kode yang terlihat pada Gambar 4.17 di atas. *method longPress* akan menjalankan step menekan tombol *continue* yang akan mengarahkan ke halaman verifikasi OTP lalu akan diverifikasi oleh method *verifyElementExist* untuk memastikan aplikasi sudah

berada di halaman tersebut lalu dilanjutkan *method delay()* pada line 83 yang akan menunda *script* berjalan selama 30 detik hingga kode OTP diterima dan peneliti memasukan secara manual pada *field* OTP, dapat terlihat pada bukti gambar dibawah ini:

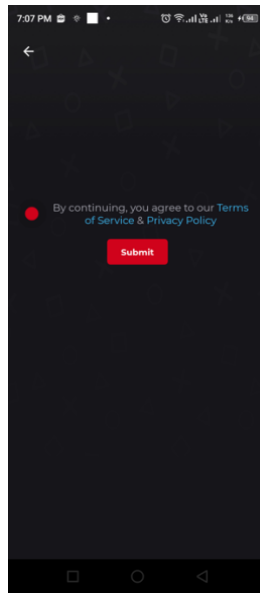


Gambar 4. 18 TC002_evidence2

```
88 Mobile.longPress(FindTestObject(Object Repository/Registration-login/android.widget.Button - Continue'), 5, FailureHandling.STOP_ON_FAILURE)
89
90 Mobile.verifyElementExist(FindTestObject(Object Repository/Registration-login/android.widget.TextView - By continuing, you agree to our Terms of Service - Privacy Policy'), 6,
91
92 Mobile.tap(FindTestObject(Object Repository/Registration-login/android.widget.RadioButton - agree'), 6, FailureHandling.STOP_ON_FAILURE)
93
94 Mobile.takeScreenshot(Configuration.getReportFolder()+'\\'+TC+'-'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
95
96 atc = atc + 1
97
98
```

Gambar 4. 19 TC002 – Menyetujui privasi polsih

Lalu langkah terakhir adalah mencentang privasi polsih dari aplikasi Duina games, dapat dilihat pada *line* 89 *method longPress* pada gambar 4.19 yang akan mejalankan step menekan tombol *continue* yang akan mengarahkan ke halaman privasi polsih lalu akan diverifikasi oleh *method verifyElementExist* untuk memastikan aplikasi sudah berada di halaman tersebut lalu dilanjutkan *method tap()* pada *line* 93 yang akan mencentang *checkbox* dapat dilihat pada bukti gambar dibawah ini :



Gambar 4. 20 TC002_evidence3

3) *Fail to login*

Pada *test case* ini user menjalankan skenario gagal melakukan login dengan beberapa step yang bertujuan untuk mengetahui *error message* yang akan muncul pada fitur login.

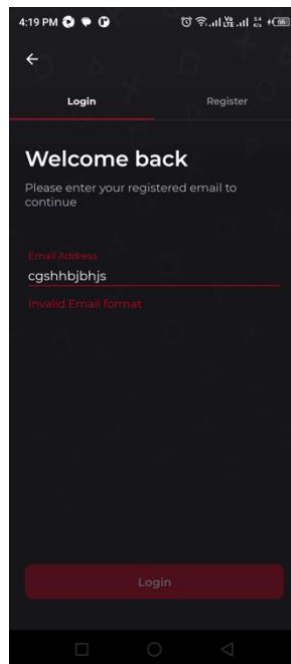
```

58 Mobile.tap(findTestObject('Object Repository/Navbar/android.widget.ImageView - Profile'), 0)
59 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - login icon'), 0)
60
61 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - login icon'), 0)
62
63 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Email Address'), 'cgsHbJbhjs', 0)
64
65 Mobile.takeScreenshot(QuiConfiguration.getReportFolder()+ '\\'+TC+' '+atc+'.png', FailureHandling.STOP_ON_FAILURE)
66
67 atc = atc + 1

```

Gambar 4. 21 TC003 - *Input Email* dengan format yang salah

Berdasarkan potongan kode yang terlihat pada Gambar 4.21 di atas, peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input* email dengan format yang salah . *method tap()* pada *line* 59, dan 61 mensimulasikan proses pengarahaan ke halaman login yang kemudian dilanjutkan dengan *method setText()* yang akan meng-*input* nilai yang telah dimasukan dalam parameter yang berupa kata random, hal ini akan men-*trigger message error* dapat dilihat pada gambar dibawah ini

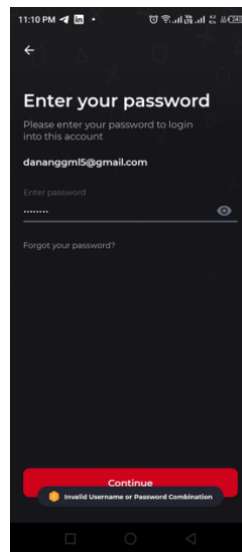


Gambar 4. 22 TC003_evidence1

```
76  
79 atc = atc + 1  
80  
81 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Enter password'), '123443434', 0)  
82  
83 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - Continue'), 0)  
84  
85 Mobile.takeScreenshot(runConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)  
86
```

Gambar 4. 23 TC003 - *Input password* yang salah

Kemudian berdasarkan potongan kode yang terlihat pada Gambar 4.21 di atas, peneliti mencoba menjalankan langkah-langkah berikutnya yaitu meng-*input* email dengan format yang salah. *method setText()* pada *line* 81 yang akan meng-*input* nilai yang telah dimasukkan dalam parameter yang berupa kata random kemudian akan dilanjutkan *method tap()* yang akan menekan *line* 81 tombol *continue* hal ini akan men-*trigger message error* dapat dilihat pada gambar dibawah ini :



Gambar 4. 24 TC003_evidence2

4) *Success to Login*

Pada *test case* ini pengguna akan menjalankan skenario berhasil melakukan login pada aplikasi dunia games, tujuannya untuk mengetahui apakah fitur login berjalan dengan baik.

```

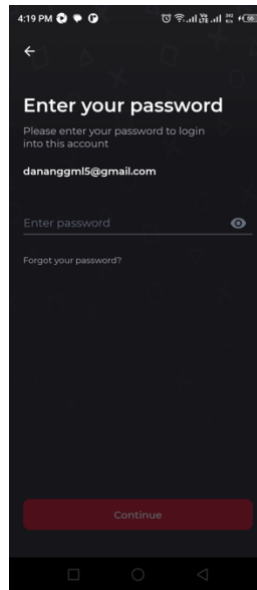
58 Mobile.tap(findTestObject('Object Repository/Navbar/android.widget.ImageView - Profile'), 0)
59
60 Mobile.tap(findTestObject('Object Repository/Registration-login/android.widget.Button - Login icon'), 0)
61
62 Mobile.setText(findTestObject('Object Repository/Registration-login/android.widget.EditText - Email Address'), 'dananggm15@gmail.com', 0)
63
64 Mobile.longPress(findTestObject('Object Repository/Registration-login/android.widget.Button - Login - active'), 0)
65
66 Mobile.takeScreenshot(Configuration.getReportFolder() + '\\*TC*' + '@tca'.png', FailureHandling.STOP_ON_FAILURE)
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Gambar 4. 25 TC004 - *Input Email* yang telah terdaftar

Berdasarkan potongan kode yang terlihat pada Gambar 4.21 di atas, peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input* email dengan format yang benar dan sudah terdaftar. *method tap()* pada *line* 59, dan 61 mensimulasikan proses pengarahaan ke halaman login yang kemudian dilanjutkan dengan *method setText()* yang akan meng-*input* nilai yang telah dimasukkan dalam parameter yang berupa kata email yang benar, lalu *method longpress* akan menekan tombol login dan aplikasi akan diarahkan ke halaman password, dapat dilihat pada bukti gambar dibawah ini

:



Gambar 4. 26 TC004_evidence1

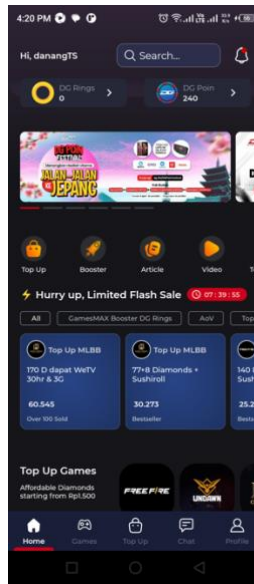
```

70 Mobile.setText(findTestObject("Object Repository/Registration-login/android.widget.EditText - Enter password"), "Tidakapa#123", 0, FailureHandling.STOP_ON_FAILURE)
71 Mobile.tap(findTestObject("Object Repository/Registration-login/android.widget.Button - Continue"), 0, FailureHandling.STOP_ON_FAILURE)
72 Mobile.delay(5)
73 Mobile.takeScreenshot(Configuration.getReportFolder() + "\\*TC*"+etc+".png", FailureHandling.STOP_ON_FAILURE)
74 etc = etc + 1

```

Gambar 4. 27 TC004 - *Input password* yang benar

Selanjutnya peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input* password yang sesuai dengan akun tersebut dapat terlihat pada Gambar 4.21 di atas, *method setText()* pada *line 71* yang akan meng-*input* nilai yang telah dimasukan dalam parameter yang berupa password yang benar kemudian akan dilanjutkan *method tap()* yang akan menekan tombol *continue*, *method delay()* pada *line line 75* digunakan untuk memberi jeda agar sebelum *method* lain dilanjutkan, ekspektasi pada test case ini ialah diarahkan kehalaman home dapat terlihat pada bukti gambar dibawah ini :



Gambar 4. 28 TC004_evidence2

5) Like and bookmark the article

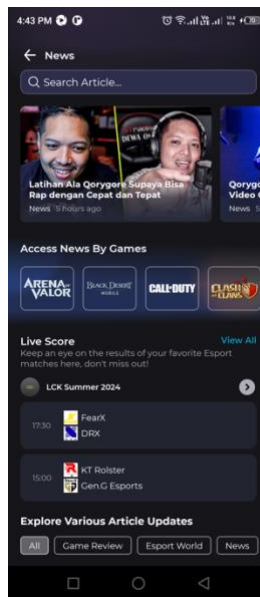
Pada *test case* ini pengguna akan menjalankan skenario menyukai artikel, menyimpan artikel ke *bookmark* dan mengomentari artikel, tujuannya untuk mengetahui tombol *like*, *bookmark* dan kolom komentar berfungsi dengan baik.

```

53  int atc = 1;
54
55  Mobile.tap(FindTextObject<Object Repository/Article/android.widget.TextView - Article>(), @)
56
57  Mobile.verifyElementExist(FindTextObject<Object Repository/Article/android.widget.TextView - Access News By Games>(), @)
58
59  Mobile.verifyElementExist(FindTextObject<Object Repository/Article/android.widget.TextView - Access News By Games>(), @)
60
61  Mobile.takeScreenshot(Configuration.getReportFolder()+"\\TC"+atc+".png", FailureHandling.STOP_ON_FAILURE)
62
63  atc = atc + 1;
64
65  Mobile.scrollToText("Explore Various Article Updates")
66
67  Mobile.verifyElementExist(FindTextObject<Object Repository/Article/android.widget.TextView - Explore Various Article Updates>(), @)
68
69  Mobile.takeScreenshot(Configuration.getReportFolder()+"\\TC"+atc+".png", FailureHandling.STOP_ON_FAILURE)
70
71  atc = atc + 1;
  
```

Gambar 4. 29 - TC005 - Verifikasi *section*

Pada *snippet code* diatas akan menjalankan langkah-langkah validasi dari setiap komponen pada halaman artikel, *method tap()* pada *line 55* akan melakukan klik pada submenu artikel lalu *method verifyElementExist()* pada *script* diatas akan verifikasi setiap *section* pada halaman artikel, dilanjutkan *method scrollToText()* yang berfungsi mengulirkan layar hingga *text* yang dimaksud ditemukan, untuk memperjelas dapat dilihat pada bukti gambar dibawah ini :



Gambar 4. 30 TC005_evidence1

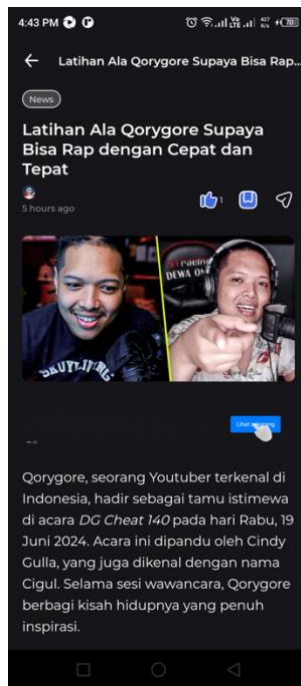
```

70
71   atc = atc + 1
72   Mobile.tap(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (1)'), 0)
73   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
74   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
75   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
76   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
77   Mobile.takeScreenshot(QunConfiguration.getReportFolder()+"\\TC+ "+atc+".png", FailureHandler.STOP_ON_FAILURE)
78   atc = atc + 1
79
80
81   Mobile.tap(findTestObject('Object_Repository/Article/android.widget.Imagebutton-Like'), 0)
82   Mobile.tap(findTestObject('Object_Repository/Article/android.widget.Imagebutton-Bookmark'), 0)
83   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
84   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
85   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
86   Mobile.verifyElementExist(findTestObject('Object_Repository/Article/android.widget.TextView - Siapan Kusta, Ukuran Call of Duty Black Ops 6 di Xbox Capai 300GB (2)'), 0)
87   atc = atc + 1

```

Gambar 4. 31 TC005 - Menekan tombol like dan bookmark

Kemudian peneliti akan menjalankan langkah-langkah menyukai artikel dan menyimpan artikel ke *bookmark* terlihat pada gambar 4.31 di atas, *method tap()* pada *line 73* akan melakukan klik pada salah satu artikel yang akan diarahkan pada *detail* artikel, *method verifyElementExist()* akan memverifikasi judul artikel untuk memastikan benar diarahkan ke halaman *detail* artikel, setelah itu *method tap()* pada *line 81* dan *83* akan melakukan klik pada tombol *like* dan *bookmark* sehingga tombol tersebut menjadi aktif, dapat dilihat pada bukti gambar dibawah ini :



Gambar 4. 32 TC005_evidence2

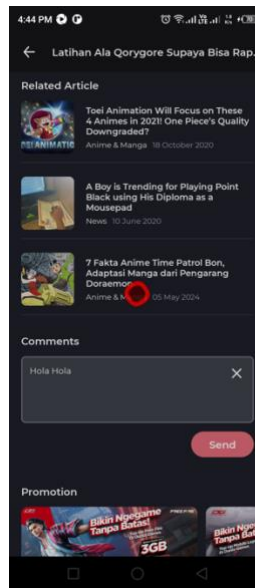
```

99      atc = atc + 1
100     Mobile.scrollToText('Comments')
101
102     Mobile.verifyElementExist(findTestObject('Object Repository/Article/android.widget.TextView - Related Article'), 0)
103
104     Mobile.takeScreenshot(BunConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
105     atc = atc + 1
106
107     Mobile.setText(findTestObject('Object Repository/Article/android.widget.EditText - Comments'), 'Halo Halo',0)
108
109     Mobile.takeScreenshot(BunConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
110     atc = atc + 1
111
112     Mobile.tap(findTestObject('Object Repository/Article/android.widget.Button - Send'), 0)
113
114     Mobile.takeScreenshot(BunConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
115     atc = atc + 1

```

Gambar 4. 33 TC005 - menulis komentar pada artikel

Selanjutnya peneliti akan mensimulasikan langkah-langkah menulis komentar pada artikel dapat terlihat pada gambar 4.33 di atas. *Method scrollToText* akan mengulirkan layar hingga text “*Comment*” ditemukan setelah itu akan diverifikasi dengan *method verifyElementExist()* pada *line 103*, *method setText()* pada *line 109* yang akan meng-*input* nilai yang telah dimasukan dalam parameter yang berupa komentar setelah itu dilanjutkan *method tap()* pada *line 115* yang akan menekan tombol *send*, untuk memperjelas dapat dilihat pada bukti gambar berikut :



Gambar 4. 34 TC005_evidence3

6) *create post in gamefeed*

Test case ini akan menjalankan skenario membuat postingan pada *gamefeed*, tujuannya untuk mengetahui *form post* berfungsi dengan baik sehingga pengguna dapat membuat postingan tanpa mengalami masalah apapun.

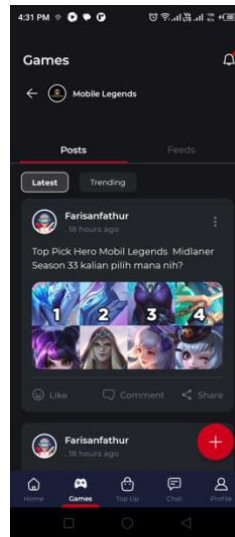
```

63 Mobile.tap(FindTestObject('Object Repository/Games Feed/android.widget.TextView - Mobile Legends'), 0)
64 Mobile.verifyElementExist(FindTestObject('Object Repository/Games Feed/android.widget.TextView - Mobile Legends (1)'), 0)
65 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.Button - Follow'), 0)
66 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Latest'), 0)
67 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Trending'), 0)
68 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Latest'), 0)
69 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Trending'), 0)
70 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Latest'), 0)
71 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Trending'), 0)
72 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Latest'), 0)
73 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Trending'), 0)
74 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Latest'), 0)
75 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Trending'), 0)
76 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Latest'), 0)
77 Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.TextView - Trending'), 0)

```

Gambar 4. 35 TC006 - Verifikasi *section* dan *tab*

Snippet code pada gambar 4.35 diatas akan menjalankan langkah-langkah validasi dari setiap komponen pada halaman *gamefeed*, method *tap()* pada *line* 64 akan melakukan klik pada list game lalu *method verifyElementExist()* script diatas akan verifikasi setiap *section* pada halaman *gamefeed*, untuk memperjelas dapat dilihat pada bukti gambar dibawah ini :



Gambar 4. 36 TC006_evidence1

```

77 Mobile.tap(findTestObject('Object Repository/Games Feed/android.widget.ImageButton - create post'), @)
78
79 Mobile.verifyElementExist(findTestObject('Object Repository/Games feed/android.widget.TextView - Create Post'), @)
80
81 Mobile.verifyElementExist(findTestObject('Object Repository/Games feed/android.widget.Button - Post'), @)
82
83 Mobile.takeScreenshot(runConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
84
85 atc = atc + 1
86
87
88 Mobile.setText(findTestObject('Object Repository/Games feed/android.widget.EditText - What do you want to share'), 'Halo World ',@)
89
90 Mobile.takeScreenshot(runConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
91
92 atc = atc + 1
93
94 Mobile.tap(findTestObject('Object Repository/Games feed/android.widget.Button - Post'), @)
95
96 Mobile.takeScreenshot(runConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
97
98 atc = atc + 1
99

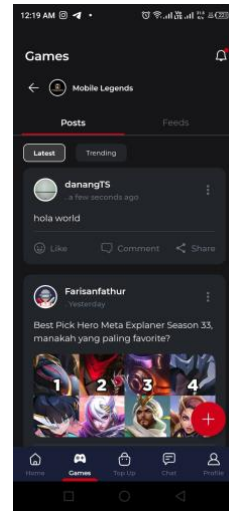
```

Gambar 4. 37 TC006 - Membuat Postingan

Selanjutnya *snippet code* diatas akan menjalankan langkah-langkah pembuatan postingan pada gamefeed, method *tap()* pada *line 78* akan melakukan klik pada tombol *create post* lalu *method verifyElementExist()* *script* diatas akan verifikasi setiap *section* pada halaman create post, *method setText()* pada *line 88* yang akan meng-*input* nilai yang ada pada parameter yang berupa kata-kata ke *field post*, lalu *method tap()* *line 94* akan melakukan klik pada tombol *post*, untuk memperjelas dapat dilihat pada bukti gambar dibawah ini :



Gambar 4. 38 TC006_evidence2



Gambar 4. 39 TC006_evidence3

7) *like one of the post*

pada *test case* ini pengguna akan menjalankan skenario menyukai salah satu postingan dan menulis komentar kepada postingan tersebut. Tujuannya untuk mengetahui apakah tombol *like* dan kolom komentar pada postingan berfungsi dengan baik.

```

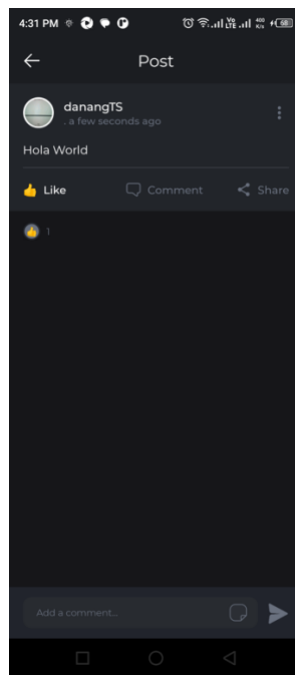
75
76     atc = atc + 1
77     Mobile.tap(FindTestObject('Object Repository/Games feed/android.widget.ImageView - Like'), 0)
78
79
80     Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
81
82     atc = atc + 1
83
84     Mobile.tap(FindTestObject('Object Repository/Games feed/android.widget.ImageView - comment'), 0)
85
86     Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.Button - Post'), 0)
87
88     Mobile.verifyElementExist(FindTestObject('Object Repository/Games feed/android.widget.EditText - Add a comment'), 0)
89
90     Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
91
92     atc = atc + 1
93
94     Mobile.setText(FindTestObject('Object Repository/Games feed/android.widget.EditText - Add a comment'), 'Hola World ', 0)
95
96     Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
97
98     atc = atc + 1
99
100    Mobile.tap(FindTestObject('Object Repository/Games feed/android.widget.ImageView - enter'), 5)
101
102    Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
103
104    atc = atc + 1

```

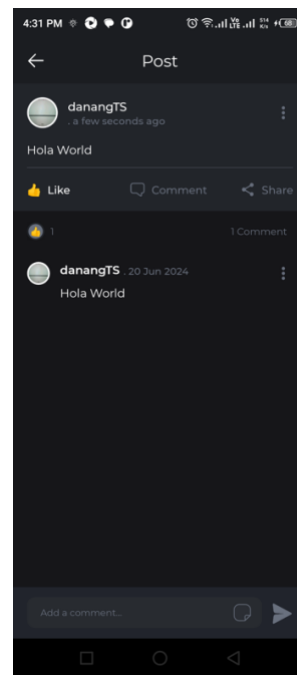
Gambar 4. 40 menekan tombol like dan menulis komentar

Pada *snippet code* diatas akan menjalankan langkah-langkah menyukai dan memberikan komentar pada salah satu postingan di gamefeed, method *tap()* pada *line* 78 akan melakukan klik pada tombol *like* lalu *method tap()* line 84 akan melakukan klik pada

tombol comment yang akan mengarahkan aplikasi pada detail postingan,s setelah itu method `verifyElementExist` akan memverifikasi objek yang ada pada halaman detail postingan untu memastikan bahwa benar telah diarahkan ke halaman tersebut, *method* `setText()` pada *line* 94 yang akan meng-*input* nilai yang ada pada parameter yang berupa kata-kata ke filed komentar, lalu *method* `tap()` *line* 100 akan melakukan klik pada tombol *enter*, untuk memperjelas dapat dilihat pada bukti gambar dibawah ini :



Gambar 4. 41TC007_evidence1



Gambar 4. 42 TC007_evidence2

8) *Create tournament*

Test case terpanjang dalam penelitian ini, skenario yang dijalan yaitu membuat *tournament* game, mulai dari *set* tanggal pendaftaran, jumlah peserta hingga *bracket* pertandingan, tujuanya untuk mengetahui setiap *field* yang ada berfungsi dengan baik.

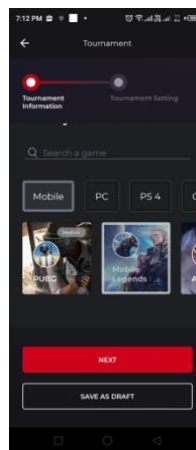
```

75     atc = atc + 1
76
77
78     Mobile.setText(FindTestObject('Object Repository/Tournament/android.widget.EditText - Tournament Name'), 'Tarkan1345', 0)
79
80     Mobile.tap(FindTestObject('Object Repository/Tournament/android.widget.ImageView - Visible'), 5)
81
82     Mobile.delay(2)
83
84     Mobile.tap(FindTestObject('Object Repository/Tournament/android.widget.TextView - Mobile'), 5)
85
86     Mobile.tap(FindTestObject('Object Repository/Tournament/android.widget.TextView - Mobile Legends'), 5)
87
88     Mobile.takeScreenshot(LunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
89
90     atc = atc + 1
91
92     Mobile.delay(2)

```

Gambar 4. 43 TC008 - *Input nama tournament*

Berdasarkan potongan kode yang terlihat pada Gambar 4.43 di atas, peneliti mencoba untuk mensimulasikan langkah-langkah meng-input nama tournament, mengatur visibilitas, memilih *platform tournament* dan *game* yang akan ditandingkan. *method setText()* pada line 78 yang akan meng-input nilai yang telah dimasukan dalam parameter yang berupa nama *tournament* dilanjutkan *method tap()* yang akan melakukan klik *radio button* visibilitas *tournament*, pada *case* ini *tournament* diatur menjadi *tournament* umum, kemudian *method tap()* pada line 84 dan 86 akan memilih *platform tournament* dan *game* yang akan ditandingkan, untuk memperjelas dappat dilihat pada gambar dibawah ini:



Gambar 4. 44 TC008_ *evidence1*

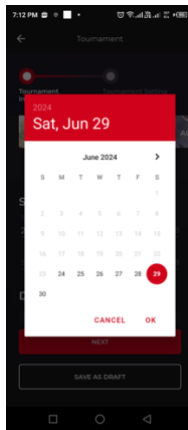
```

93
94     Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.EditText - Select your start date'), 5)
95
96     Mobile.tap(findTestObject('Object Repository/Tournament/android.view.View - 24'), 5)
97
98     Mobile.takeScreenshot(runConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
99
100     atc = atc + 1
101
102     Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - OK'), 5)
103
104     Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.EditText - Select your end date'), 5)
105
106     Mobile.tap(findTestObject('Object Repository/Tournament/android.view.View - 29'), 5)
107
108     Mobile.takeScreenshot(runConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
109
110     atc = atc + 1
111
112     Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - OK'), 5)
113
114     Mobile.delay(2)
115
116     Mobile.setText(findTestObject('Object Repository/Tournament/android.widget.EditText - desc'), 'Tournament antar kampung', 0)
117
118     Mobile.takeScreenshot(runConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
119
120     atc = atc + 1
121
122     Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - NEXT'), 5)
123

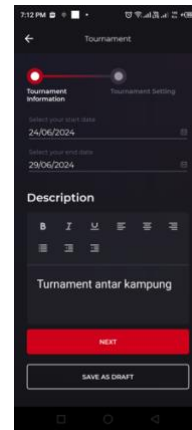
```

Gambar 4. 45 TC008 - *Setting deadline Tournament*

Selanjutnya peneliti akan mensimulasikan langkah-langkah meng-input *deadline tournament* dilaksanakan dan meng-input dekripsi *tournament* dapat dilihat pada Gambar 4.45 di atas. *method tap()* pada line 94 dan 96 mensimulasikan klik pada *field start date* dan pemilihan tanggal mulai *tournament* kemudian *method tap()* pada line 102 akan melakukan klik pada tombol ok untuk menyimpan data tersebut begitu juga *method tap()* pada line 104 dan 106 mensimulasikan klik pada *field end date* dan pemilihan tanggal berakhirnya *tournament* kemudian *method tap()* pada line 112 akan melakukan klik pada tombol ok untuk menyimpan data tersebut, Kemudian dilanjutkan dengan *method setText()* yang akan meng-input nilai yang telah dimasukan dalam parameter yang deskripsi *tournament* untuk memperjelas dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 46 TC008_evidence2



Gambar 4. 47 TC008_evidence3

```

124 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.TextView - Participants'), 0)
125
126 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.Button - NEXT'), 0)
127
128 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.Button - SAVE AS DRAFT'), 0)
129
130 Mobile.setText(findTestObject('Object Repository/Tournament/android.widget.EditText - 4'), '10', 0)
131
132 Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
133
134 atc = atc + 1
135

```

Gambar 4. 48 TC008 - Input jumlah partisipan

Kemudian dilanjutkan potongan kode yang terlihat pada Gambar 4.48 di atas, peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input* jumlah partisipan yang dapat mengikuti *tournament*, *method verifyElementExist* pada line 126 dan 128 akan verifikasi tombol next dan tombol save as draft kemudian *method setText()* pada line 130 akan menginput nilai yang ada pada parameter yang berupa jumlah partisipan yang dapat mengikuti *tournament*.

```

135 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - SETTINGS'), 5)
136
137 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.TextView - Registration Settings'), 0)
138
139 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.TextView - Registration Period'), 0)
140
141 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.Button - SAVE'), 0)
142
143 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc'+'.png', FailureHandling.STOP_ON_FAILURE)
144
145 atc = atc + 1
146
147
148 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.EditText - Registration Opening'), 5)
149
150 Mobile.tap(findTestObject('Object Repository/Tournament/android.view.View - 16'), 5)
151
152 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc'+'.png', FailureHandling.STOP_ON_FAILURE)
153
154 atc = atc + 1
155
156 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - OK'), 5)
157
158 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.EditText - Registration Closing'), 5)
159
160 Mobile.tap(findTestObject('Object Repository/Tournament/android.view.View - 23'), 5)
161
162 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc'+'.png', FailureHandling.STOP_ON_FAILURE)
163
164 atc = atc + 1
165
166 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - OK'), 5)
167
168 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - SAVE'), 5)
169
170 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc'+'.png', FailureHandling.STOP_ON_FAILURE)
171
172 atc = atc + 1

```

Gambar 4. 49 TC008 - Setting deadline pendaftaran Tournament

Selanjutnya peneliti mencoba untuk mensimulasikan langkah-langkah meng-*input deadline* registrasi tournament terlihat pada potongan kode Gambar 4.49 di atas, *method tap()* pada line 148 dan 150 mensimulasikan klik pada *field start registration date* dan pemilihan tanggal mulai registrasi tournament kemudian *method tap()* pada line 156 akan melakukan klik pada tombol ok untuk menyimpan data tersebut begitu juga *method tap()* pada line 158 dan 160 mensimulasikan klik pada *field end registration date* dan pemilihan tanggal berakhirnya registrasi *tournament* kemudian *method tap()* pada line 166 akan melakukan klik pada tombol ok untuk menyimpan data tersebut.

```

173
174 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.RadioButton - Using Bracket'), 5)
175
176 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.TextView - Tournament Stage'), 0)
177
178 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.ImageView - agree'), 5)
179
180 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc'+'.png', FailureHandling.STOP_ON_FAILURE)
181
182 atc = atc + 1
183
184 Mobile.tap(findTestObject('Object Repository/Tournament/android.widget.Button - NEXT'), 5)
185
186 Mobile.delay(5)
187
188 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.TextView - Congratulations'), 0)
189
190 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.Button - GO TO TOURNAMENT'), 0)
191
192 Mobile.verifyElementExist(findTestObject('Object Repository/Tournament/android.widget.Button - DONE'), 0)
193
194 Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\'+TC'+'+atc'+'.png', FailureHandling.STOP_ON_FAILURE)
195

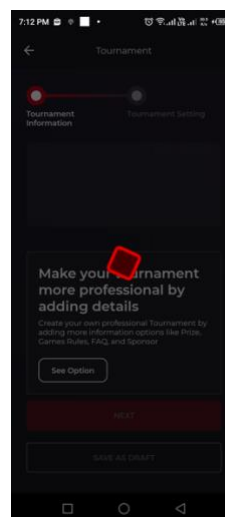
```

Gambar 4. 50 TC008 - membuat *bracket tournament*

Terakhir peneliti mencoba untuk mensimulasikan langkah-langkah memilih *bracket* pertandingan dan mencentang

persetujuan tournament. terlihat pada potongan kode Gambar 4.43 di atas, *method tap()* pada line 174 mensimulasikan klik pada *radio button bracket*, dalam case ini *tournament* akan menggunakan *bracket*, kemudian dilanjutkan *method tap()* yang akan klik *checkbox* persetujuan pembuatan tournament lalu dilanjut *method tap()* pada line 184 yang akan klik tombol *next* dan pembuatan tournament akan diproses, jika pembuatan berhasil maka akan di arahkan ke halaman sukses pembuatan tournament method *verifyElementExist* pada line 188 sampai 192 akan memverifikasi *element* yang ada pada halaman tersebut untuk memastikan halaman yang dimaksud sudah sesuai.

Sayangnya untuk *test case* ini terjadi masalah pada aplikasi, terdapat issue yang ditemukan oleh peneliti ketika menjalankan script yaitu terjadi *loading* terus menerus dan gagal dalam penyimpanan data *tournament*, dapat dilihat pada bukti gambar berikut :



Gambar 4. 51TC009_evidence4

9) *Create roomchat*

Pada *test case* ini pengguna akan menjalankan skenario pembuatan *roomchat*, tujuannya untuk mengetahui apakah *field-field roomchat* berfungsi dengan baik.

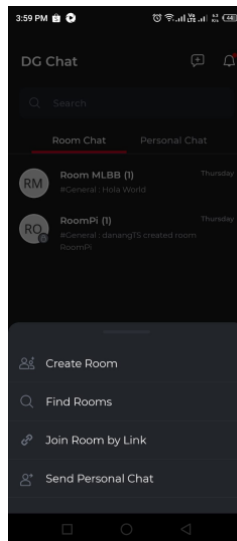

```

66
67 Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.Button - menu lain'), 0)
68
69 Mobile.verifyElementExist(FindTestObject('Object Repository/Chat/android.widget.TextView - Create Room'), 0)
70
71 Mobile.takeScreenshot('..\\ScreenShots\\View.png', FailureHandling.STOP_ON_FAILURE)
72
73 Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.TextView - Create Room'), 0)
74
75 Mobile.setText(FindTestObject('Object Repository/Chat/android.widget.EditText - Room Name'), 'RoomPi', 0)
76

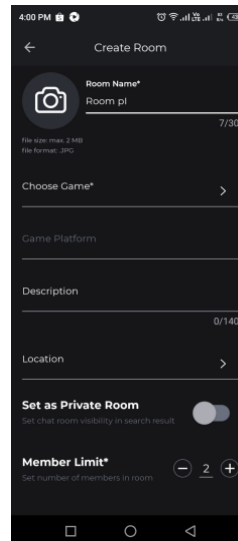
```

Gambar 4. 52 TC009 - Input nama roomchat

Pada *snippet code* diatas akan menjalankan langkah-langkah pembuatan *room chat*, method *tap()* pada *line 67* akan melakukan klik pada tombol menu lain lalu *method verifyElementExist()* pada *line 69* akan memverifikasi menu *create room* lalu akan dilanjutkan method *tap()* pada *line 73* akan melakukan klik pada menu create room, *method setText()* pada *line 75* yang akan menginput nilai yang ada pada parameter yang berupa nama *room* ke *field room name*, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 53 TC009_evidence1



Gambar 4. 54 TC009_evidence2

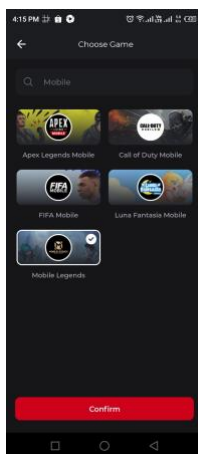
```

77 Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.TextView - Choose Game'), 5)
78
79 Mobile.delay(3)
80
81 Mobile.setText(FindTestObject('Object Repository/Chat/android.widget.EditText - Search (1)'), 'Mobile Legend', 0)
82
83 Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.TextView - Mobile Legends'), 5)
84
85 Mobile.takeScreenshot('..\\ScreenShots\\View.png', FailureHandling.STOP_ON_FAILURE)
86
87 Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.Button - Confirm (1)'), 5)
88

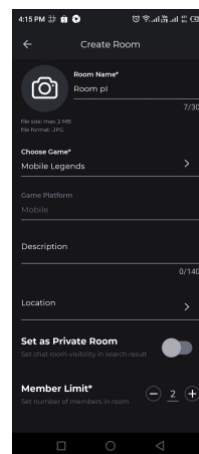
```

Gambar 4. 55 TC009 - memilih game

Setelah itu peneliti akan menjalankan langkah-langkah memilih *game* untuk pembuatan *roomchat*. Terlihat pada potongan kode gambar 4.55 diatas, *method tap()* pada *line 77* akan melakukan klik pada filed choose game yang akan diarahkan ke halaman list game, setelah *method setText()* pada *line 81* yang akan meng-*input* nilai yang ada pada parameter yang berupa nama game ke *search bar*, setelah *method tap()* pada *line 83* akan melakukan klik pada game yang dimaksud, setelah games terpilih maka *method tap()* pada *line 87* akan melakukan klik pada tombol *confirm* untuk melanjutkan proses pembuatan room, untuk memperjelas prose dapat melihat bukti gambar dibawah ini:



Gambar 4. 56 TC009_evidence3



Gambar 4. 57 TC009_evidence4

```

88
89 Mobile.setText(findTestObject('Object Repository/Chat/android.widget.EditText - Description'), 'Mobile Legend',0)
90
91 Mobile.tap(findTestObject('Object Repository/Chat/android.widget.EditText - Location'), 5)
92
93 Mobile.setText(findTestObject('Object Repository/Chat/android.widget.EditText - Search (2)'), 'Jakarta selatan',0)
94
95 Mobile.tap(findTestObject('Object Repository/Chat/android.widget.TextView - JAKARTA SELATAN'), 5)
96
97 Mobile.takeScreenshot('..\\ScreenShots\\View.png', FailureHandling.STOP_ON_FAILURE)
98
99 Mobile.tap(findTestObject('Object Repository/Chat/android.widget.Button - Confirm (1)'), 5)
100

```

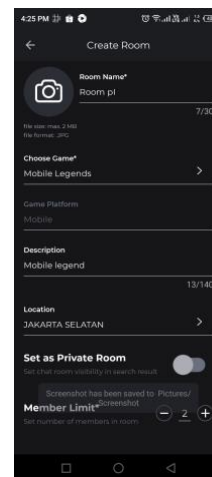
Gambar 4. 58 TC009 - Input deskripsi dan lokasi

Selanjutnya *snippet code* Gamabr 4.58 diatas akan menjalankan langkah-langkah menginput dekripsi dan memilih lokasi dari *room chat* ,*method setText()* pada *line 89* yang akan meng-*input* nilai yang ada pada parameter yang berupa dekripsi ke *field description*, lalu dilanjutkan *method tap()* pada *line 91* akan

melakukan klik pada *field location* yang akan diarahkan ke halaman list kota yang ada di Indonesia , setelah *method setText()* pada *line 93* yang akan meng-*input* nilai yang ada pada parameter yang berupa nama kota ke *search bar*, setelah *method tap()* pada *line 95* akan melakukan klik pada kota yang dimaksud, setelah kota terpilih maka *method tap()* pada *line 99* akan melakukan klik pada tombol *confirm* untuk melanjutkan proses pembuatan *room chat*, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 59 TC009_evidence5



Gambar 4. 60 TC009_evidence6

```

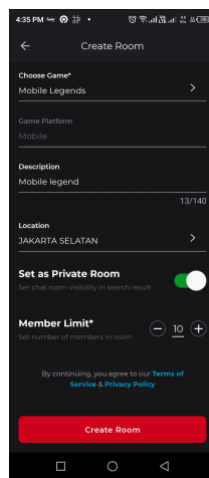
101     Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.Switch private room chat'), 5)
102
103     Mobile.setText(FindTestObject('Object Repository/Chat/android.widget.EditText - 2'), '10', 0)
104
105     Mobile.takeScreenshot('\\\\ScreenShots\\\\View.png', FailureHandling.STOP_ON_FAILURE)
106
107     Mobile.tap(FindTestObject('Object Repository/Chat/android.widget.Button - Create Room'), 5)
108
109     Mobile.takeScreenshot('\\\\ScreenShots\\\\View.png', FailureHandling.STOP_ON_FAILURE)
110

```

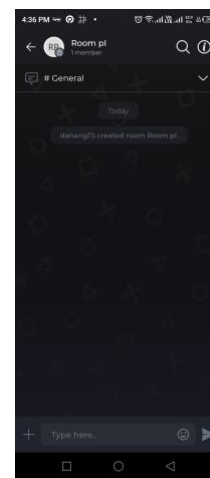
Gambar 4. 61 TC009 - input jumlah member

Terakhir, *snippet code* pada gambar 4.61 diatas akan menjalankan langkah-langkah mengatur visibilitas room chat, menginput jumlah member dan klik tombol *create room*, *method tap()* pada *line 101* akan melakukan klik pada tombol *switch* visibilitas *room*, pada *case* ini *room chat* akan diatur menjadi *room chat* untuk umum, setelah itu *method setText()* pada

line 103 yang akan meng-input nilai yang ada pada parameter yang berupa jumlah member yang boleh ikut kedalam *room chat* ke *field member*, lalu dilanjutkan *method tap()* pada line 107 akan melakukan klik pada tombol *confirm* untuk melanjutkan proses pembuatan *room chat*, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 62 Tc009_evidence7



Gambar 4. 63 TC009_evidence8

10) Send message to one of room chat

Pada *test case* ini pengguna akan menjalankan skenario mengirim pesan pada *roomvchat* yang telah diikuti oleh pengguna, tujuannya untuk mengetahui apakah kolom *chat* berfungsi dengan benar dan pengguna dapat mengirim pesan ke *roomchat*.

```

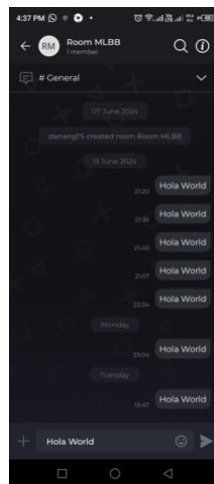
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83

```

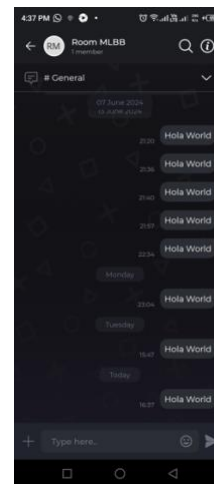
Gambar 4. 64 TC010 - Input pesan

Pada *snippet code* diatas akan menjalankan langkah-langkah menulis dan mengirimkan pesan pada *roomchat* yang diikuti

pengguna, *method tap()* pada *line 66* akan melakukan klik pada salah satu *roomchat*, setelah itu *method verifyElementExist()* akan memverifikasi objek yang ada pada *roomchat* untuk memsadikan benar diarahkan ke *roomchat*, *method setText()* pada *line 74* yang akan meng-*input* nilai yang ada pada parameter yang berupa kata kata ke *field chat*, lalu dilanjutkan *method tap()* pada *line 80* akan melakukan klik pada tombol *enter* untuk mengirim pesan yang telah ditulis sebelumnya, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 65 TC010_evidence1



Gambar 4. 66 TC010_evidence2

11) *Fail to topup diamond*

Pada test case ini pengguna akan menjalankan skenario gagal melakukan *topup diamond* salah satu *game*, tujuannya untuk mengetahui *message error* yang akan muncul pada setiap *field* dan proses dalam melakukan *topup*.

```

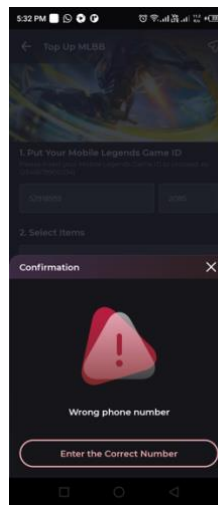
65 Mobile.tap(FindTestObject('Object Repository/Topup/android.widget.TextView - Top Up MLBB'), 0)
66 Mobile.tap(FindTestObject('Object Repository/Topup/android.widget.Button - Buy'), 0)
67
68
69
70 Mobile.verifyElementExist(FindTestObject('Object Repository/Topup/android.widget.TextView - Wrong phone number'), 0)
71
72 Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
73
74 atc = atc + 1
75

```

Gambar 4. 67 TC011 - tekan tombol *buy* tanpa *input* apapun

Pada *snippet code* diatas akan menjalankan langkah-langkah klik tombol *buy* tanpa mengisi nomor dan *id games*, *method tap()*

pada *line* 66 akan melakukan klik pada salah satu produk game setelah itu method *tap()* *line* 68 akan melakukan klik pada tombol buy tujuannya untuk *men-trigger message error*, method *verifyElementExist()* akan memverifikasi objek *message error* yang muncul, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini.

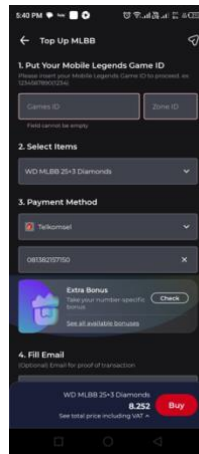


Gambar 4. 68 TC011_evidence1

```
75 Mobile.tap(findTestObject('Object Repository/Topup/android.widget.Button - Enter the Correct Number'), 0)
76
77 Mobile.setText(findTestObject('Object Repository/Topup/android.widget.EditText - Fill Telkomsel Number'), '081382157150', 0)
78
79
80 Mobile.tap(findTestObject('Object Repository/Topup/android.widget.Button - Buy'), 0)
81
82 Mobile.verifyElementExist(findTestObject('Object Repository/Topup/android.widget.TextView - Field cannot be empty'), 0)
83
```

Gambar 4. 69 TC011 - tekan tombol buy tanpa input ID games

Selanjutnya *snippet code* pada gambar 4.69 diatas akan menjalankan langkah-langkah klik tombol buy tanpa mengisi *id game*, method *tap()* pada *line* 75 akan melakukan klik tombol *enter the correct number*, kemudian method *setText()* pada *line* 78 yang akan meng-*input* nilai yang ada pada parameter yang berupa nomor yang valid ke *field number*, dilanjutkan method *tap()* pada *line* 80 akan melakukan klik pada tombol *buy* untuk *men-triggre message error*, method *verifyElementExist()* akan memverifikasi objek *message error* yang muncul, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 70 TC011_evidence2

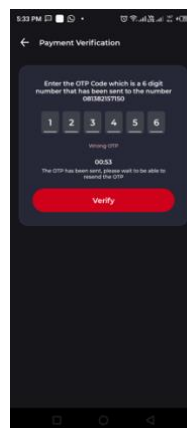
```

125
126 Mobile.setText(findTestObject('Object_Repository/Topup/android.widget.EditText - OTP'), '123456', 0)
127
128 Mobile.tap(findTestObject('Object_Repository/Topup/android.widget.Button - Verify'), 0)
129
130 Mobile.verifyElementExist(findTestObject('Object_Repository/Topup/android.widget.TextView - Wrong OTP'), 0)
131
132 Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
133
134 atc = atc + 1
135

```

Gambar 4. 71 TC011_input OTP yang salah

Terakhir, *snippet code* pada gambar 4.17 diatas akan menjalankan langkah-langkah meng-input kode OTP yang salah , *method setText()* pada *line* 126 yang akan meng-input nilai yang ada pada parameter yang berupa angka random ke *field OTP*, lalu dilanjutkan *method tap()* pada *line* 80 akan melakukan klik pada tombol *verify* untuk men-trigger *message error*, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini.



Gambar 4. 72 TC011_evidence3

12) *Topup diamond one of games*

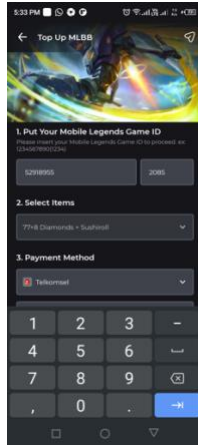
Pada *test case* ini pengguna akan menjalankan skenario berhasil melakukan *topup diamond* yang bertujuan untuk mengetahui apakah transaksi *topup diamond game* berjalan dengan baik.

```
63
64     atc = atc + 1
65
66     Mobile.tap(findTestObject('Object Repository/Topup/android.widget.TextView - Top Up MLBB'), @)
67
68     Mobile.setText(findTestObject('Object Repository/Topup/android.widget.EditText - Games ID'), '52918955', @)
69
70     Mobile.setText(findTestObject('Object Repository/Topup/android.widget.EditText - Zone ID'), '2885', @)
71
72     Mobile.tap(findTestObject('Object Repository/Topup/android.widget.ImageView - expand paket'), @)
73
74     Mobile.takeScreenshot(runConfiguration.getReportFolder()+"\\*TC*_"+atc+".png", FailureHandling.STOP_ON_FAILURE)
75
76     atc = atc + 1
77
78     Mobile.delay(2)
79
80     Mobile.tap(findTestObject('Object Repository/Topup/android.widget.TextView - 3 Diamonds'), @)
81
82     Mobile.tap(findTestObject('Object Repository/Topup/android.widget.Button - Select'), @)
83
```

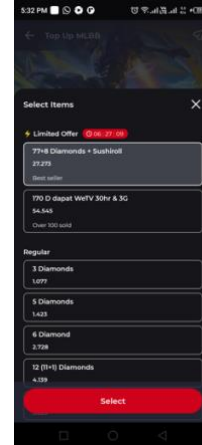
Gambar 4. 73 TC012 - *Input ID games* dan pilih paket *diamond*

Pada *snippet code* diatas akan menjalankan langkah-langkah meng-input *id games* dan memilih paket *diamond*, *method tap()* pada line 66 akan melakukan klik pada salah satu produk *game*, setelah itu dilanjutkan *method setText()* pada line 68 & 70 yang akan meng-input nilai yang ada pada parameter yang berupa *id games* dan *id zone* pada *field*, *method tap()* pada line 72 akan melakukan klik pada field paket untuk memunculkan list paket yang tersedia dilanjut *method tap()* pada line 80 yang akan melakukan klik pada salah satu paket yang tersedia, setelah paket terpilih dilanjutkan *method tap()* pada line 82 akan melakukan klik

pada tombol *select* untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 74 TC012_evidence1



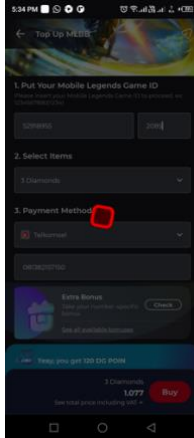
Gambar 4. 75 TC012_evidence2

```
84 Mobile.setText(findTestObject('Object Repository/Topup/android.widget.EditText - Fill Telkonsel Number'), '081382157150',0)
85
86 Mobile.tap(findTestObject('Object Repository/Topup/android.widget.Button - Buy'), 0)
87
88 Mobile.takeScreenshot(RunConfiguration.getReportFolder()+ '\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
89
90 atc = atc + 1
91
```

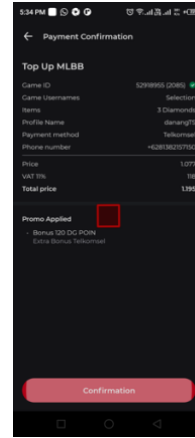
Gambar 4. 76 TC012_input nomor dengan format yang benar

Setelah itu *snippet code* pada gambar 4.76 diatas akan menjalankan langkah-langkah meng-input nomor valid dan klik buy, *setText()* pada *line* 84 yang akan meng-input nilai yang ada pada parameter yang berupa nomor valid ke *field number phone*, lalu dilanjutkan *method tap()* pada *line* 86 akan melakukan klik pada tombol *buy* dan jika *id games* dan nomor telponnya valid

maka akan diarahkan ke halaman konfirmasi, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 77 TC012_evidence2



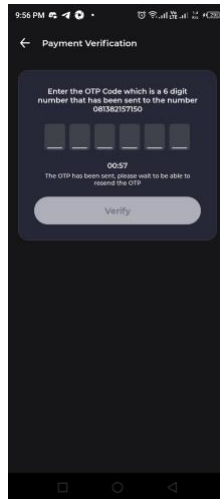
Gambar 4. 78 TC012_evidence3

```
101         atc = atc + 1
102
103
104         Mobile.tap(FindTestObject('Object Repository/Topup/android.widget.Button - Confirmation'), 5)
105         Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
106
107         atc = atc + 1
108
109         Mobile.delay(15)
110
111
112         Mobile.tap(FindTestObject('Object Repository/Topup/android.widget.Button - Verify'), 5)
113         Mobile.verifyElementExist(FindTestObject('Object Repository/Topup/android.widget.TextView - Payment Successful'), 0)
114         Mobile.takeScreenshot(RunConfiguration.getReportFolder()+'\\'+TC+'_'+atc+'.png', FailureHandling.STOP_ON_FAILURE)
115
116
117         atc = atc + 1
118
```

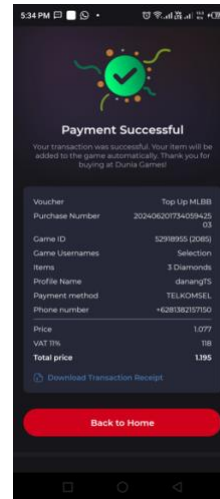
Gambar 4. 79 TC012 - Input OTP

Selanjutnya *snippet code* pada gambar 4.79 diatas akan menjalankan langkah-langkah meng-input kode OTP yang valid , *method tap()* pada *line* 104 akan melakukan klik tombol confirmation akan diarahkan ke halaman verifikasi kode OTP pada proses ini tidak bisa dilakukan secara automation karena perlu kode *valid* dari partner ke tiga maka dari itu peneliti menggunakan *method delay* pada *line* 110 yang akan menjeda script selama 30 detik untuk menunggu kode OTP diterima setelah itu peneliti memasukan kode tersebut secara manual ke *field* OTP, setelah jeda waktu habis maka akan *script* akan menjalankan method selanjutnya yaitu *method tap()* pada *line* 112 yang akan melakukan klik pada tombol *verify* setelah itu jika pembelian berhasil

dilakukan maka akan diarahkan ke halaman pembelian sukses, *method verifyElementExist* akan verifikasi objek pada halaman tersebut untuk memastikan pembayaran berhasil, untuk memperjelas prosesnya dapat dilihat pada bukti gambar dibawah ini:



Gambar 4. 80 TC012_evidence5

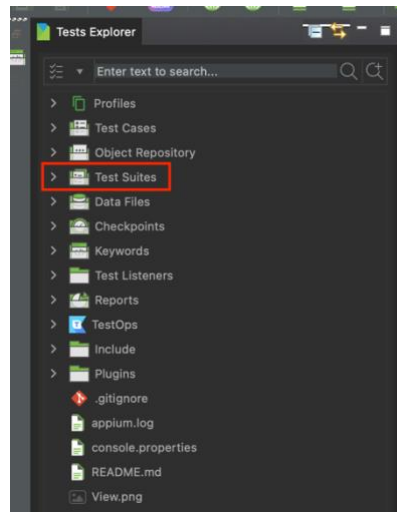


Gambar 4. 81 TC012_evidence6

4.2.3 Ekekusi Script Automation

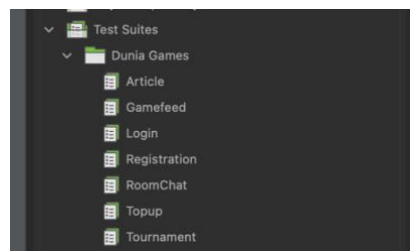
Tahap selanjutnya setelah *script* selesai dibuat adalah pengekseskuan sehingga pengujian dapat berjalan secara otomatis, pada tahap ini ada beberapa hal yang perlu dilakukan untuk membuat *report* hasil pengujian tercetak secara otomatis.

Test case yang siap di eksekusi perlu dibuatkan *Test suite*-nya terlebih dahulu, *Test suite* merupakan salah satu fitur dari katalon yang berfungsi menampung *test case* sehingga *test case* dapat dijalankan secara bersamaan, jika *test case* dieksekusi tapi tidak pada *test suite* maka *report* hasil eksekusi tidak ada tercetak, *test suite* terletak sejajar dengan *test case* yaitu pada *sidebar test explore* dari katalon studio dapat dilihat pada gambar berikut:



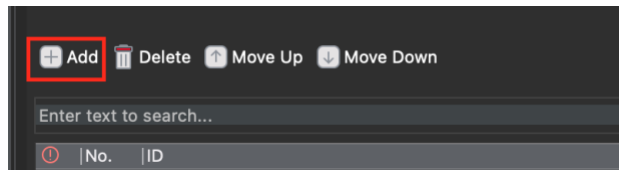
Gambar 4. 82 *Test Suite*

Pada penelitian ini peneliti membuat *test suite* sesuai dengan fitur yang akan diuji agar lebih mudah mengeksekusi dan menyiapkan pra kondisi dari setiap *test case* yang dapat dilihat pada gambar berikut :

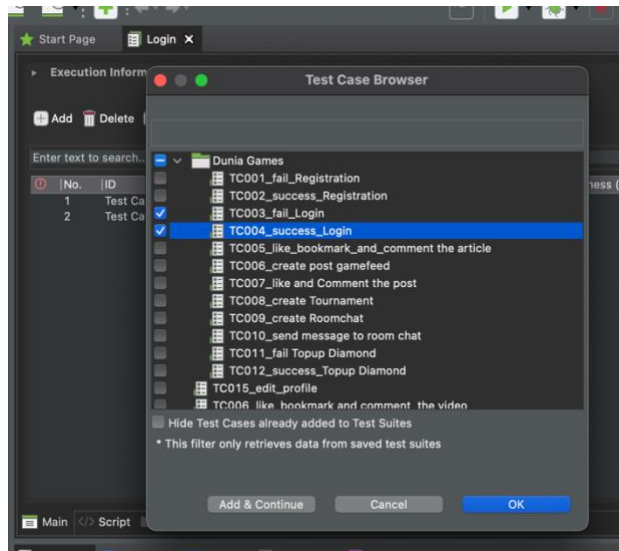


Gambar 4. 83 *Test suite per-fitur*

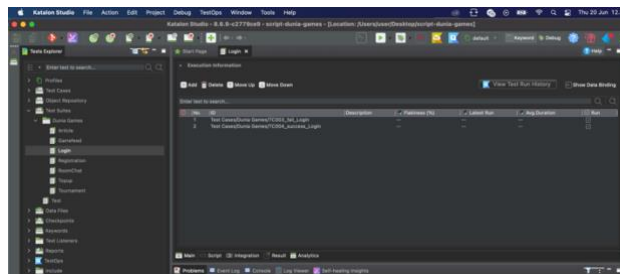
Setelah *test suite* dibuat langkah selanjutnya adalah menambahkan *test case* ke *test suite* dengan cara klik tombol add pada bagian atas setelah itu akan muncul *popup* yang berisi *list testcase*-nya, klik *checkbox* pada *test case* yang ditambahkan lalu klik oke, setelah itu *test suite* siap untuk dieksekusi, untuk memperjelas dapat dilihat pada gambar-gambar dibawah ini:



Gambar 4. 84 Tombol *Add* pada *Test suite*

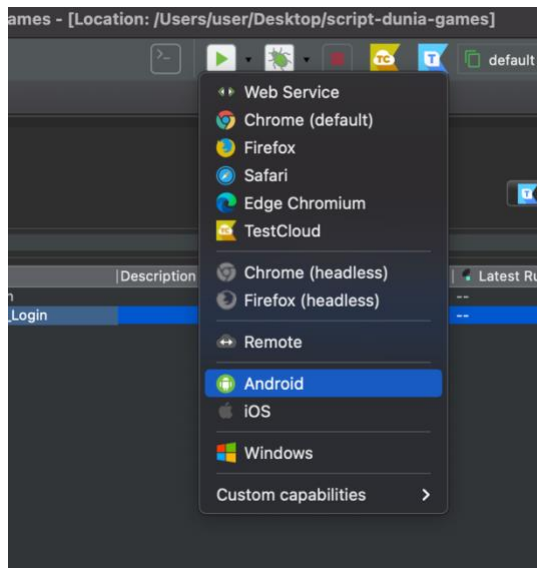


Gambar 4. 85 *Test case* dipilih

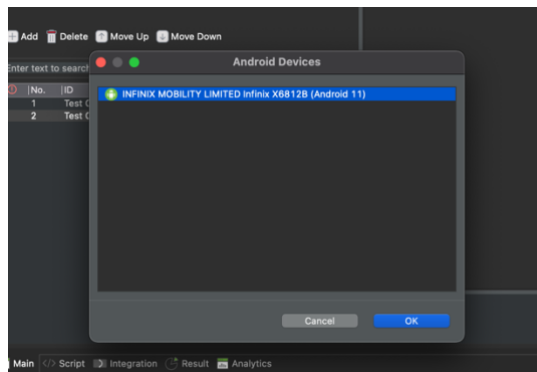


Gambar 4. 86 *Test suite* yang siap untuk dieksekusi

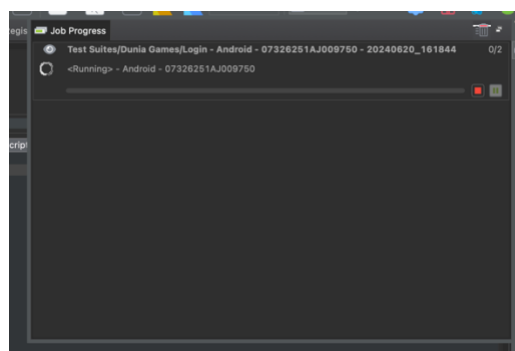
Sebelum eksekusi dilakukan perlu dipastikan *device android* sudah tersambung dengan komputer lalu klik opsi *run button* yang ada pada atas *test suite* lalu pilih *device android* setelah itu akan muncul *device* yang tersambung lalu pilih *device* tersebut dan klik oke maka *script* pun akan dijalankan, untuk memperjelas prosesnya dapat dilihat pada gambar-gambar berikut :



Gambar 4. 87 Pilih *platform* untuk menjalankan *script*



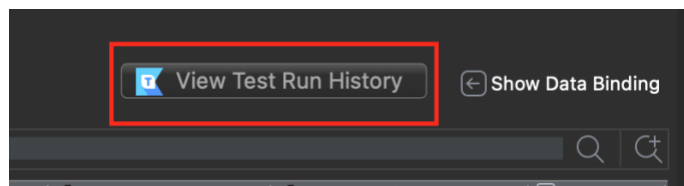
Gambar 4. 88 Pilih *device* yang tersambung



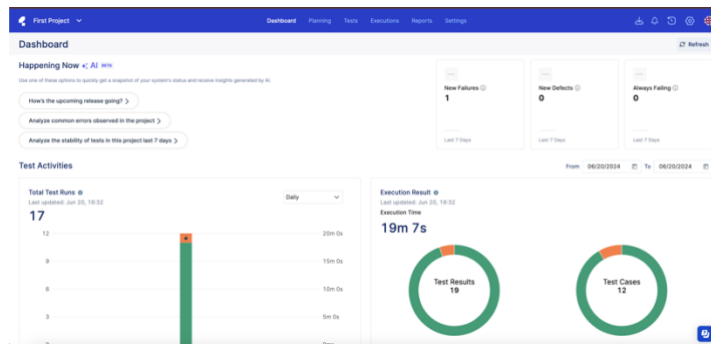
Gambar 4. 89 *Test case* yang tereksekusi

4.2.4 Hasil Eksekusi Script Automation

Setelah *Test suite* tereksekusi maka report hasil akan secara otomatis terbuat, *report* hasil pengujian dapat diakses melalui *Test Ops* yaitu fitur dari katalon yang akan menyajikan ringkasan dari hasil *testing* yang telah dijalankan, mengaksesnya dengan klik tombol *test ops* pada bagian atas *test suite* maka akan diarahkan secara otomatis ke *browser* dan *dashboard* proyek, untuk memperjelas pada dilihat pada gambar dibawah ini :

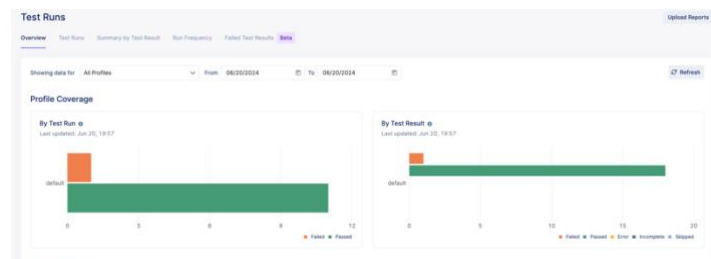


Gambar 4. 90 *Test Ops*



Gambar 4. 91 *Dashboard* Hasil pengujian

Berikut ini adalah hasil pengujian dari implementasi *automation testing* pada aplikasi Dunia Games pada penelitian ini:



Gambar 4. 92 *Summary result testing*

Pada gambar diatas menunjukan jumlah *test case* yang telah di eksekusi beserta statusnya, dapai dilihat pada bagan sebelah kiri total keseluruhan

Test case ada 12 dengan 11 *test case* dinyatakan *passed* atau lulus dan 1 *failed* atau gagal, sedangkan pada grafik di kanan merupakan jumlah pengujian dijalankan, perhari ini ada 18 kali pengujian dengan 17 berstatus *passed* atau lulus dan 1 *testcase* gagal.

Name	Path	Maintainer	Last executed	Last updated	No. of executions	Average Duration	Flakiness (%)	Requirements
TC008_create Tournament	Uploaded Data.../Dunia Games		27 mins ago	7 mins ago	5	1m 8s	60.00	
TC002_success_Registration	Uploaded Data.../Dunia Games		32 mins ago	29 mins ago	5	1m 11s	0.00	
TC001_fail_Registration	Uploaded Data.../Dunia Games		39 mins ago	38 mins ago	4	1m 2s	0.00	
TC011_fail_Topup Diamond	Uploaded Data.../Dunia Games		2 hours ago	2 hours ago	3	1m 17s	0.00	
TC012_success_Topup Diamond	Uploaded Data.../Dunia Games		2 hours ago	2 hours ago	3	1m 19s	0.00	
TC005_like_bv...the article	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	4	52s	25.00	
TC009_create Roomchat	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	3	44s	0.00	
TC010_send message to room chat	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	3	27s	0.00	
TC008_create post gamefeed	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	4	31s	0.00	
TC007_like and Comment the post	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	4	46s	0.00	
TC003_fail_Login	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	3	30s	0.00	
TC004_success_Login	Uploaded Data.../Dunia Games		3 hours ago	3 hours ago	3	30s	0.00	

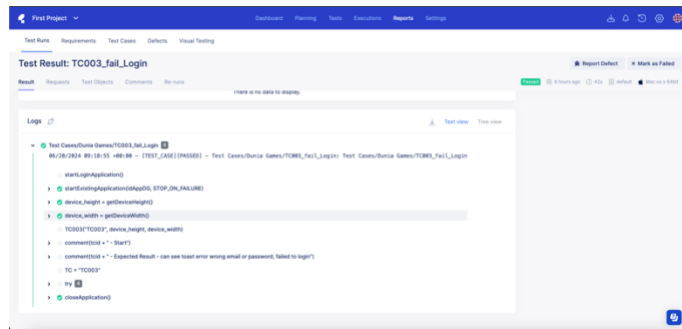
Gambar 4. 93 *Test case* yang telah tereksekusi

Pada gambar diatas dapat dilihat list *test case* yang telah selesai tereksekusi, disetiap *testcase* terdapat kolom waktu, durasi dan kerentanan yang ditemukan, berdasarkan gambar tersebut durasi terlama eksekusi ada pada angka 1 menit 8 detik yang merupakan *test case* pembuatan *tournament* dengan status *failed* atau gagal, pada *test case* tersebut juga ditemukan kerentanan / *flakiness* paling besar dari keseluruhan *test case* yaitu sebesar 60%, angka *flakiness* ini yang menjadi salah satu faktor status *test case* tersebut *failed*.

Berikut penjelasan lebih mendetail terkait hasil pengujian automation setiap *test case* :

dijalankan sehingga status pada *test case* tersebut *passed* tetapi tetap tidak bisa dilakukan automation sepenuhnya.

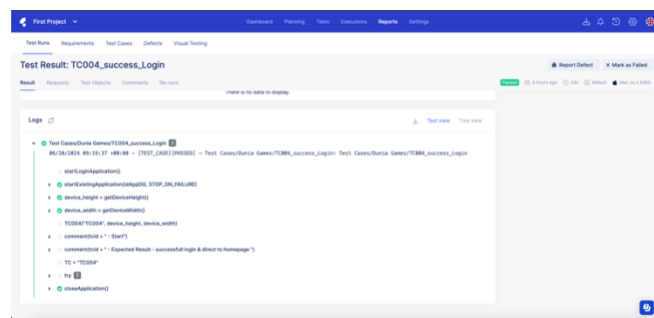
3) *Fail to login*



Gambar 4. 96 TC003-Status *Passed*

Test case ini dikategorikan *feasible* untuk *automation testing*, karena step pada skenario dijalankan automation selain itu hasil status pengujian juga berupa *passed* yang artinya sudah memenuhi ekspektasi yang telah ditentukan.

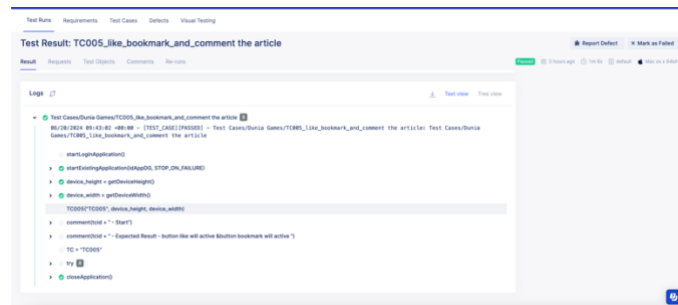
4) *Succes to login*



Gambar 4. 97 TC004 - Status *Passed*

Test case ini juga dikategorikan *feasible* dan dapat dilihat pada gambar *test case* mendapat status *Passed* dalam artian sudah memenuhi ekspektasi.

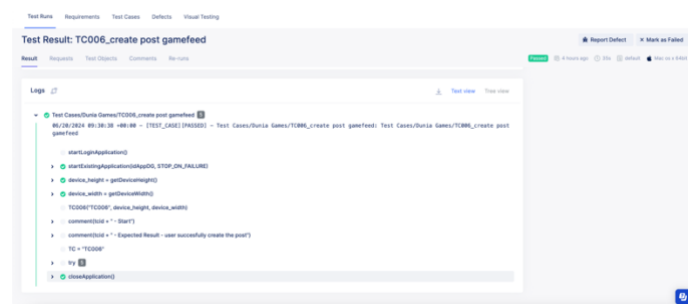
5) Like, bookmark and comment article



Gambar 4. 98 TC005 - Status Passed

Pada gambar diatas merupakan *detail* dari skenario menyukai, menyimpan ke *bookmark* dan menulis komentar ke salah satu artikel, dapat dilihat mendapatkan status *passed* yang berarti berhasil dijalankan dan sesuai ekspektasi, selain itu *test case* ini dikategorikan *test case feasible* karena dapat sepenuhnya dijalankan secara *automation*

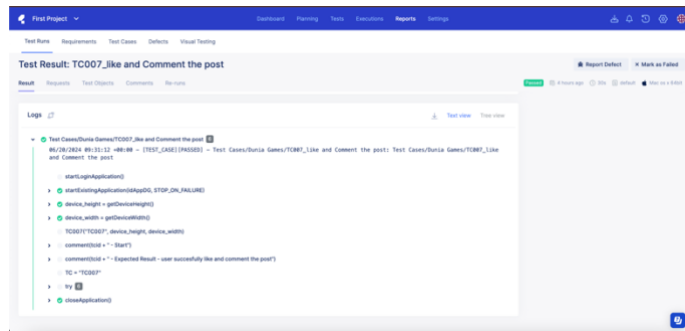
6) Create post in gamefeed



Gambar 4. 99 TC006 - Status Passed

Pada gambar diatas merupakan *detail* dari *test case* membuat postingan di *gamefeed*, dapat dilihat mendapatkan status *passed* dalam artian berhasil dijalankan dan sesuai ekspektasi yang telah ditentukan, *test case* tersebut juga dikategorikan *feasible* karena sepenuhnya dijalankan secara *automation*.

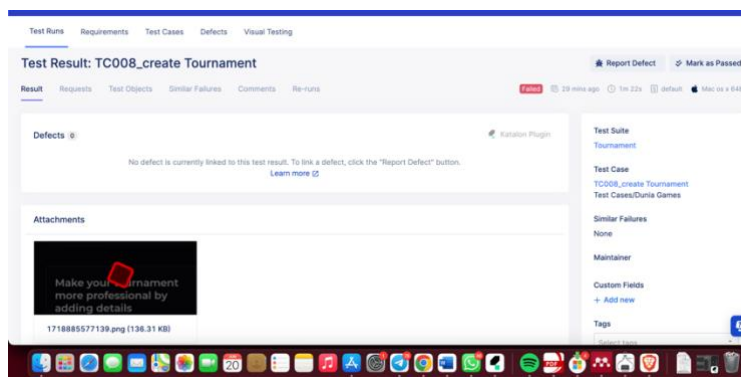
7) Like and comment one of the post



Gambar 4. 100 TC007 – Status *Passed*

Pada gambar diatas merupakan *detail* dari *test case* menyukai dan menulis komentar di salah satu postingan, dapat dilihat *test case* tersebut mendapatkan status *passed* yang berarti berhasil dijalankan dan sesuai dengan ekspektasi selain itu *test case* tersebut dikategorikan *feasible*.

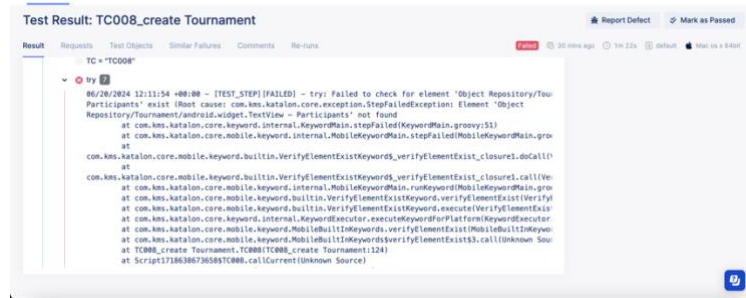
8) Create tournament



Gambar 4. 101 TC008 - Status *Failed*

Pada gambar diatas merupakan *detail* dari *test case* membuat *tournament games*, satu-satunya test case yang berstatus *failed* pada penelitian ini, penyebabnya karena terjadi *loading* yang terlalu lama lalu muncul *message error* “terjadi kesalahan pada server” ketika

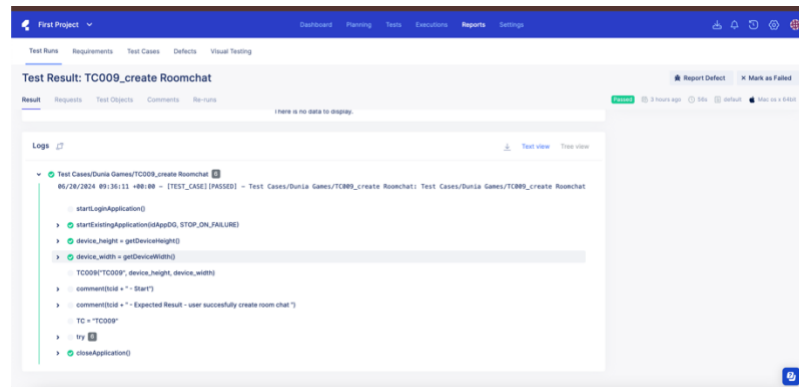
mencoba menyimpan data *tournament*, hal ini membuat aplikasi *stuck* pada halaman *form tournament* sehingga *script* tidak dapat menjalankan step selanjutnya.



Gambar 4. 102 TC008 - Detail Error

Dapat dilihat pada *detail test case* diatas, terjadi kegagalan verifikasi objek partisipan dikarenakan tidak bisa menemukan objek tersebut, hal ini yang membuat *test case* menjadi *failed*, selain itu *test case* ini dikategorikan sebagai *test case feasible* karena memungkinkan untuk dijalankan sepenuhnya oleh *script automation*.

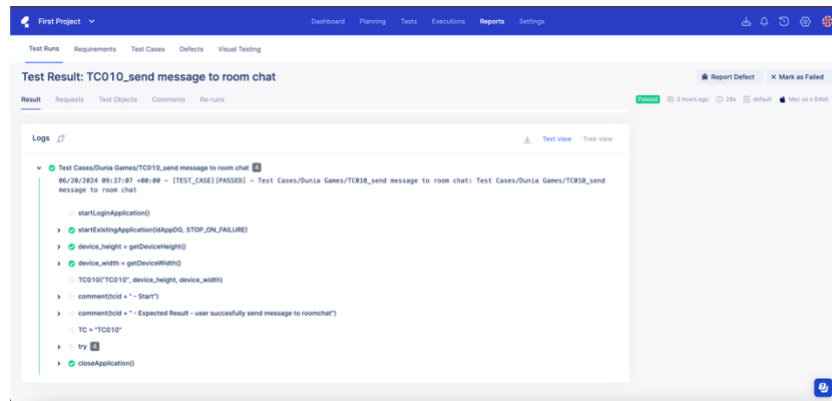
9) Create room chat



Gambar 4. 103 TC009 - Status Passed

Pada gambar diatas merupakan *detail* dari *test case* membuat *room chat*, dapat dilihat bahwa *test case* tersebut mendapatkan status *passed* yang dalam artian berhasil untuk dijalankan dan sesuai dengan ekspektasi, selain itu *test case* ini dikategorikan sebagai *feasible* karena bisa sepenuhnya di jalankan oleh *script automation*.

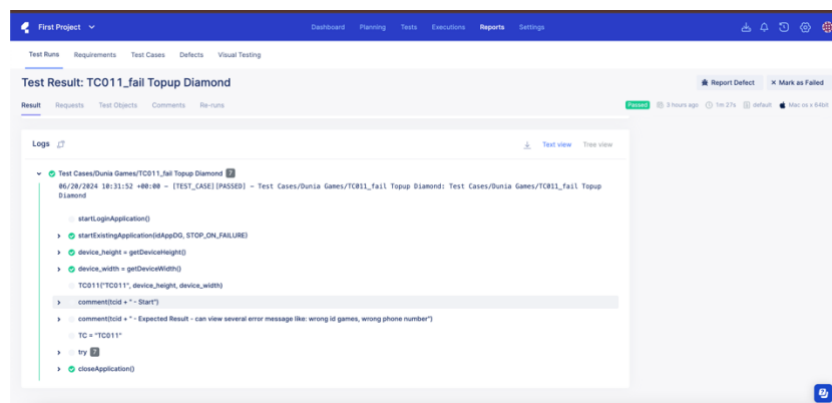
10) Send message to room chat



Gambar 4. 104 TC010 - Status *Passed*

Pada gambar diatas merupakan *detail* dari *test case* mengirim pesan ke salah satu *room chat* yang diikuti pengguna, dapat dilihat pada gambar bahwa *test case* tersebut berhasil dijalankan sehingga mendoatka status *passed*, selain itu *test case* ini juga dikategorikan sebagai *feasible* karena dapat sepenuhnya dijalankan oleh *script automation*.

11) Fail to topup diamond one of game

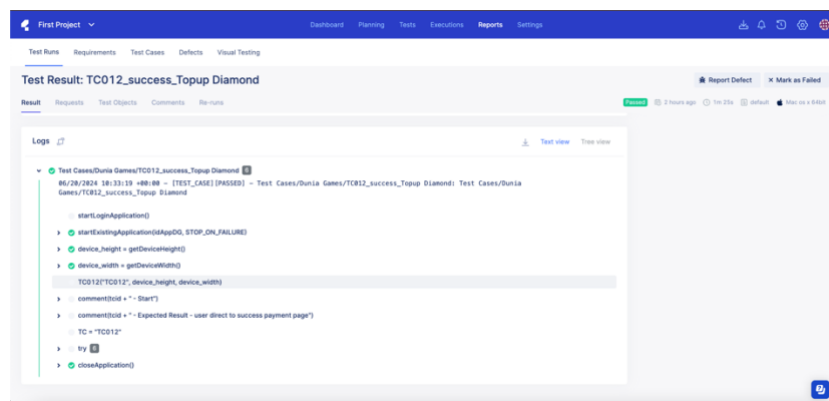


Gambar 4. 105 TC011 - Status *Passed*

Pada gambar diatas merupakan *detail* dari *test case* gagal melakukan *topup diamond* salah satu game, dapat dilihat bahwa *test case* tersebut berhasil dijalankan karena mendapatkan status *passed*, selain itu *test*

case ini terbilang *feasible* karena belum membutuhkan OTP yang benar dari *third party*

12) Success to topup diamond one of game



Gambar 4. 106 TC012 - Status *Passed*

Pada gambar diatas merupakan *detail* dari *test case* berhasil melakukan *topup diamond* salah satu *games*, dapat dilihat pada detail tersebut bahwa *test case* tersebut berhasil dijalankan sehingga mendapatkan status *passed*, namun *test case* ini dikategorikan sebagai *test case* yang *non feasible* karena pada proses melakukan *topup* memerlukan kode OTP dari *third party*, peneliti memodifikasi script dengan menambahkan *delay* hingga OTP didapatkan, setelah peneliti memasukan OTP secara manual sehingga step selanjutnya dapat berjalan dan *test case* tetap dijalankan dengan *script automation* walaupun tidak sepenuhnya.

4.3 Evaluasi Hasil Implementasi

Hasil yang didapat setelah *script* pada penelitian ini dieksekusi adalah sebagai berikut

Tabel 4. 2 *Test case* berdasarkan status

TC	Skenario	Status
TC001	<i>Fail to registration</i>	Berhasil dieksekusi
TC002	<i>Successfully registration new account</i>	Berhasil dieksekusi
TC003	<i>Fail to login</i>	Berhasil dieksekusi
TC004	<i>Success to login</i>	Berhasil dieksekusi
TC005	<i>Like and bookmark the article</i>	Berhasil dieksekusi
TC006	<i>Create post in gamefeed</i>	Berhasil dieksekusi
TC007	<i>Like one of the post</i>	Berhasil dieksekusi
TC008	<i>Create tournament</i>	Tidak berhasil dieksekusi
TC009	<i>Create roomchat</i>	Berhasil dieksekusi
TC010	<i>Send message to one of room chat</i>	Berhasil dieksekusi
TC011	<i>Fail to topup diamond</i>	Berhasil dieksekusi
TC012	<i>Topup diamond one of games</i>	Berhasil dieksekusi

Dari 12 *test case* yang diujikan 11 diantaranya dinyatakan berhasil dieksekusi dan sudah memenuhi ekspektasi yang telah ditetapkan, sedangkan 1 *test case* tidak berhasil dieksekusi dikarenakan *issue* pada aplikasi lebih tepatnya terdapat *error* pada *server* Tindakan yang perlu diambil oleh seorang QA adalah melaporan temuan *defect* kepada *developer* agar segera diperbaiki.

Dengan kata lain tingkat keberhasilan pengujian sebesar pada aplikasi Dunia games dipenelitian ini sebesar 91.6 % dengan rincian hitung berikut:

$$\frac{11}{12} \times 100 = 91.6\%$$

Implementasi *automation testing* pada penelitian ini sudah cukup baik tetapi belum sepenuhnya sempurna, kedepannya akan lebih baik jika melakukan integrasi pada tools deployment dan tools dokumentasi seperti CI/CD dan Jira. Hal ini memungkinkan agar *automation testing* dapat lebih mudah dalam penerapan STLC (Software Testing Life Cycle) yang lebih kompleks seperti dokumentasi hasil pengujian, dokumentasi *issue*, integrasi kode baru, eksekusi menggunakan pipeline scheduler dan lain-lain.

Implementasi *script* pada penelitian ini menggunakan Bahasa Groovy yang cukup rumit untuk dimengerti oleh orang awam. Sebaiknya, penelitian kedepannya bisa menerapkan BDD (*Behavior Driven Development*) yang menggunakan Bahasa Gherkin, yang lebih mudah dibaca. Tujuannya adalah agar kolaborasi dengan *developer* dan klien saat pengembangan perangkat lunak menjadi lebih mudah.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian ini, implementasi *automation testing* pada aplikasi dunia games yang telah dibahas, dapat disimpulkan bahwa:

1. implementasi *automation testing* pada aplikasi dunia games menggunakan katalon studio dilakukan dalam beberapa tahap yaitu :
 - menentukan skenario pengujian
 - mengembangkan skenario menjadi *test case*
 - menangkap objek pada aplikasi dunia games,
 - *test case* di implementasikan menjadi script automation berbahasa pemrograman java dengan *framework groovy*.
2. Menjalankan script automation dilakukan dengan membuat test suite lalu memasukan test case script ke test suite yang telah dibuat, hal ini dilakukan agar *report* hasil pengujian dapat ter-*generate* secara otomatis, setelah itu hasil pengujian dapat diakses melalui test ops.
3. Hasil implementasi *automation testing* dari 12 *test case*, sebanyak 11 *test case* berhasil untuk dieksekusi dan 1 *test case* gagal dieksekusi dikarenakan terdapat *issue error* pada *backend*.

5.2 Saran

Berikut saran dari peneliti untuk untuk memperbaiki kekurangan pada penelitian ini :

1. Menambahkan jumlah skenario yang akan diuji, supaya dapat mengeksplor penggunaan katalon studio dalam menerapkan *automation testing* pada *mobile aplikasi*
2. Menggunakan *device cloud* atau *emulator* agar lebih mudah memilih versi sistem operasi android sesuai kebutuhan.
3. Menambahkan id pada setiap objek di aplikasi Dunia Games agar mempermudah menyeleksi objek.

DAFTAR PUSTAKA

- [1] A. Setiadi, P. Yuliatmojo, and D. Nurhidayat, "Pengembangan Aplikasi *Android* Untuk Pembelajaran Pneumatik," *Jurnal Pendidikan Vokasional Teknik Elektronika (JVoTE)*, vol. 1, no. 1, pp. 1–5, 2018, doi: 10.21009/jvote.v1i1.6886.
- [2] M. Nur Ichsanudin, M. Yusuf, S. Jurusan Rekayasa Sistem Komputer, J. Teknik Industri, I. AKPRIND Yogyakarta, and R. Artikel, "Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode *Black Box Testing* Bagi Pemula Info Artikel Abstrak," vol. 1, no. 2, pp. 1–8, 2022, doi: 10.55123.
- [3] A. Zulianto, A. Purbasari, N. Suryani, A. I. Susanti, F. R. Rinawan, and W. G. Purnama, "Pemanfaatan Katalon Studio untuk Otomatisasi Pengujian *Black-Box* pada Aplikasi *iPosyandu*," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 7, no. 3, p. 370, 2021, doi: 10.26418/jp.v7i3.46954.
- [4] N. Kadek et al., "Rancang Bangun Aplikasi *Mobile* Siska Berbasis *Android*," *Online*, 2018. [Online]. Available: <http://jurnal.stiki-indonesia.ac.id/index.php/sintechjournal>
- [5] A. Arfan and Hendrik, "Penerapan STLC dalam Pengujian Automation Aplikasi *Mobile* (Studi kasus: LMS Amikom Center PT.GIT Solution)," *Jurusan Informatika, Universitas Islam Indonesia*, vol. 3, no. 2, pp. 1–6, 2022, [Online]. Available: <https://jurnal.uui.ac.id/AUTOMATA/article/view/24127/>
- [6] B. Huda and A. Lia Hananto, "Penerapan *Software Testing Life Cycle* Pada Pengujian Otomatisasi Platform *Dzikra Application of Software Testing Life Cycle in Automated Testing of Dzikra Platform*," *CSRID Journal*, vol. 15, no. 1, pp. 1–11, 2023, [Online]. Available: <https://www.doi.org/10.22303/csrid.15.1.2023.01-11>
- [7] F. Ardi and H. P. Putro, "Pengujian *Black Box* Aplikasi *Mobile* Menggunakan Katalon Studio (Studi Kasus: ACC *Partner* PT. Astra Sedaya *Finance*)," *Jurnal Universitas Islam Indonesia*, vol. 2, no. 1, pp. 1–6, 2021.

- [8] Katalon Community, “What is Automation Testing? Ultimate Guide & Best Practices,” <https://katalon.com/resources-center>. Accessed: Mar. 27, 2024. [Online]. Available: <https://katalon.com/resources-center/blog/what-is-automation-testing>
- [9] Katalon Docs, “About Katalon Studio.” Accessed: Mar. 27, 2024. [Online]. Available: <https://docs.katalon.com/katalon-studio/about-katalon-studio>
- [10] D. Katarina, E. W. Ambarsari, and T. Case, “Automation Testing Tool Dalam Pengujian Aplikasi Belajar Tajwid Pada Platform Android,” *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, vol. 4, no. 2, pp. 205–212, 2019.
- [11] A. Indrayanti, B. A. Wardijono, D. Nur, and R. Aulia, “Analisis Pengujian Graphical User Interface E-Commerce Dengan Menggunakan Katalon Studio,” *Seminar Nasional Teknologi Informasi dan Komunikasi STI&K (SeNTIK)*, vol. 5, no. 1, 2021.
- [12] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita, “Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap,” *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, vol. 3, no. 2, p. 206, 2018, doi: 10.30998/string.v3i2.3048.
- [13] R. Hasibuan and D. Suryana, “Pengaruh Metode Eksperimen Sains Terhadap Perkembangan Kognitif Anak Usia 5-6 Tahun,” *Jurnal Obsesi: Jurnal Pendidikan Anak Usia Dini*, vol. 6, no. 3, pp. 1169–1179, Aug. 2021, doi: 10.31004/obsesi.v6i3.1735.
- [14] I. R. Dhaifullah, M. Muttanifudin H, A. Ananda Salsabila, and M. Ainul Yaqin, “Survei Teknik Pengujian Software,” *Journal Automation Computer Information System*, vol. 2, no. 1, pp. 31–38, 2022, doi: 10.47134/jacis.v2i1.42.

LAMPIRAN

Lampiran 1 Report Test ops setiap testcase

Registration

Execution Environment

Host name	user - 192.168.1.6
Local OS	Mac OS X 64bit
Katalon version	8.8.9.208
Browser	null

Summary

ID	Test Suites/Dunia Games/Registration		
Description			
Total	2	Failed	0
Passed	2	Incomplete	0
Skipped	0		
Start	2024-06-20 18:58:46	End	2024-06-20 19:01:09
Elapsed	2m - 22.691s		

#	ID	Description	Status
1	Test Cases/Dunia Games/TC001_fail_Registration		PASSED
2	Test Cases/Dunia Games/TC002_success_Registration		PASSED

TC001_fail_Registration

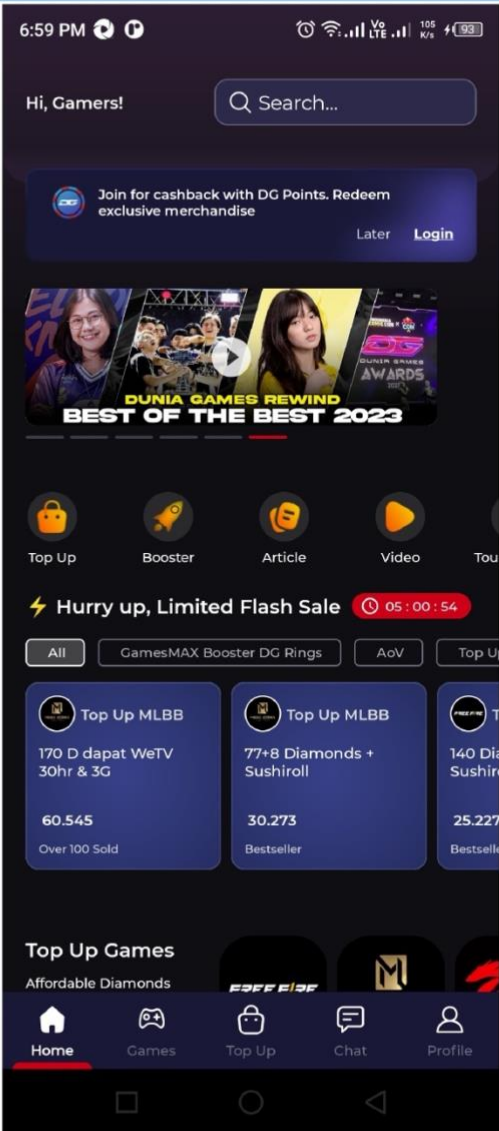
Information

ID	Test Cases/Dunia Games/TC001_fail_Registration		
Description			
Tag			
Start	2024-06-20 18:58:46	End	2024-06-20 18:59:54
Elapsed	1m - 7.927s		
Status	PASSED		


Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.001s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	17.209s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.135s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.088s	PASSED
5	TC001('TC001', device_height, device_width)	0.003s	PASSED
6	comment(tcid + " - Start") TC001 - Start	0.017s	PASSED
7	comment(tcid + " - Expected Result - user can see wrong format email error message") TC001 - Expected Result - user can see wrong format email error message	0.003s	PASSED

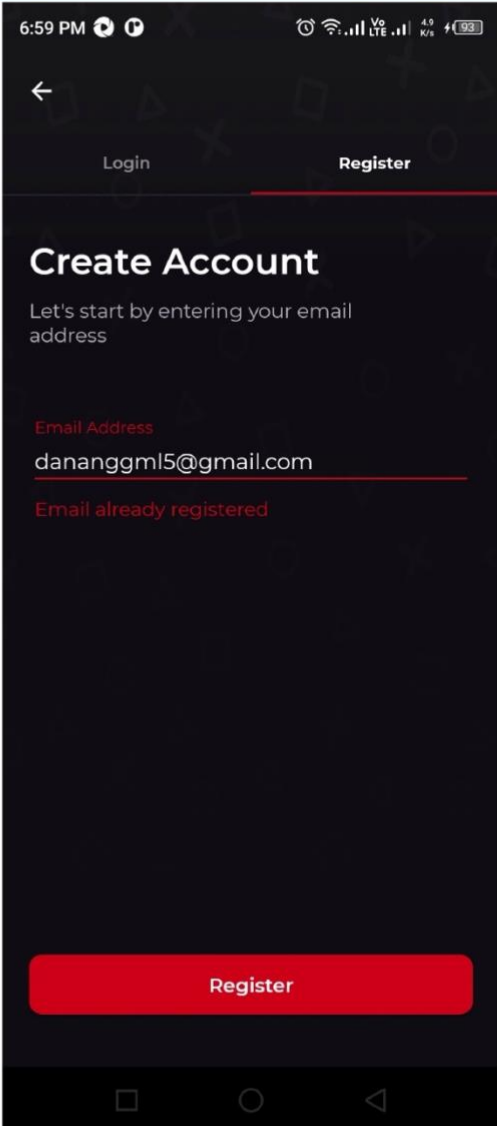
TC001_fail_Registration

#	Description	Elapsed	Status
			

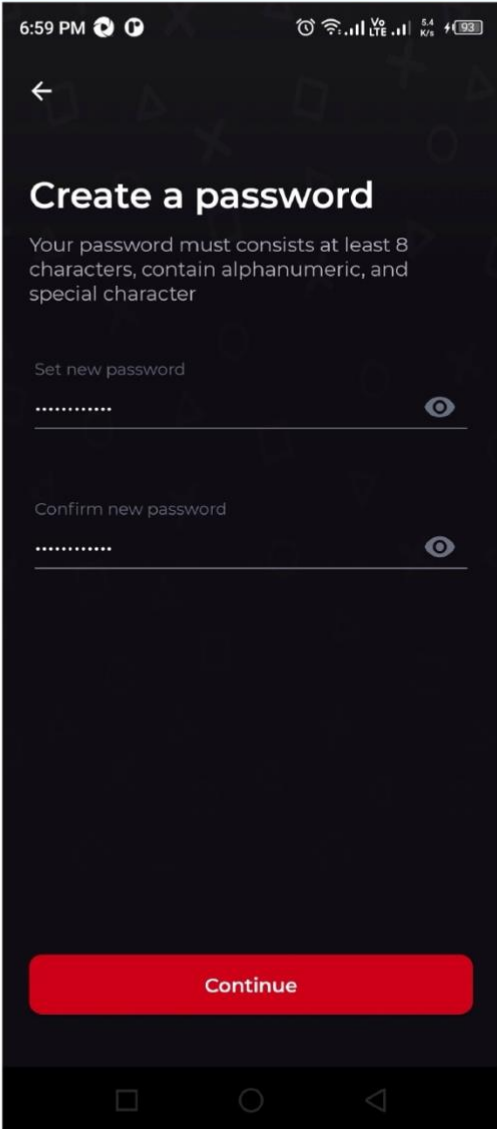
TC001_fail_Registration

#	Description	Elapsed	Status
			


TC001_fail_Registration

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for account registration. At the top, there are navigation options for 'Login' and 'Register', with 'Register' being the active tab. Below this, the heading 'Create Account' is displayed, followed by the instruction 'Let's start by entering your email address'. An input field labeled 'Email Address' contains the text 'dananggml5@gmail.com'. Below the input field, a red error message reads 'Email already registered'. At the bottom of the screen, there is a prominent red button labeled 'Register'. The status bar at the top of the phone shows the time as 6:59 PM and a battery level of 93%.</p>		

TC001_fail_Registration

#	Description	Elapsed	Status
			

TC001_fail_Registration

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for entering a verification code. At the top, the status bar displays the time as 6:59 PM, along with icons for signal strength, Wi-Fi, and battery level at 93%. The app screen has a dark background with a white back arrow in the top left corner. The main heading is "Enter Verify Code" in white. Below it, the text reads "Please input verification code that has been sent to" followed by the email address "adoremoccasin@navalcadets.com". In the center, four white boxes contain the digits "8", "1", "2", and "3" respectively. Below the input boxes, a timer indicates "The code will expire in 00:54". At the bottom, there is a red "Continue" button. A dark grey error message box is overlaid on the bottom of the screen, containing a yellow warning icon and the text "The OTP you have entered is invalid". The Android navigation bar is visible at the very bottom of the screenshot.</p>		

TC002_success_Registration


Information

ID	Test Cases/Dunia Games/TC002_success_Registration		
Description			
Tag			
Start	2024-06-20 19:05:59	End	2024-06-20 19:07:44
Elapsed	1m - 44.868s		
Status	PASSED		

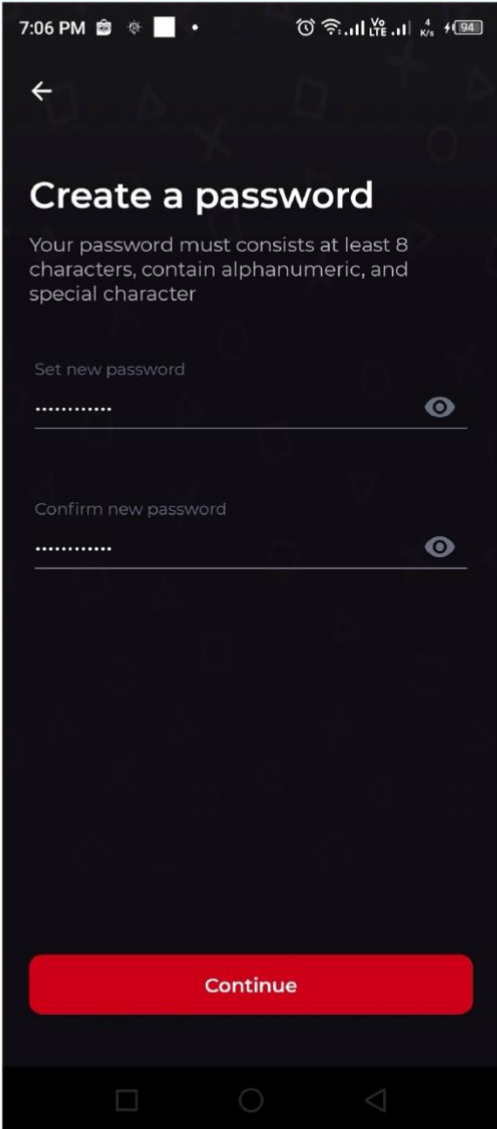
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.003s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 1d.co.duniagames'	20.327s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.136s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.113s	PASSED
5	TC002("TC002", device_height, device_width)	0.002s	PASSED
6	comment(tcid + " - Start") TC002 - Start	0.019s	PASSED
7	comment(tcid + " - Expected Result - success to register and direct to homepage") TC002 - Expected Result - success to register and direct to homepage	0.003s	PASSED


TC002_success_Registration

#	Description	Elapsed	Status
			

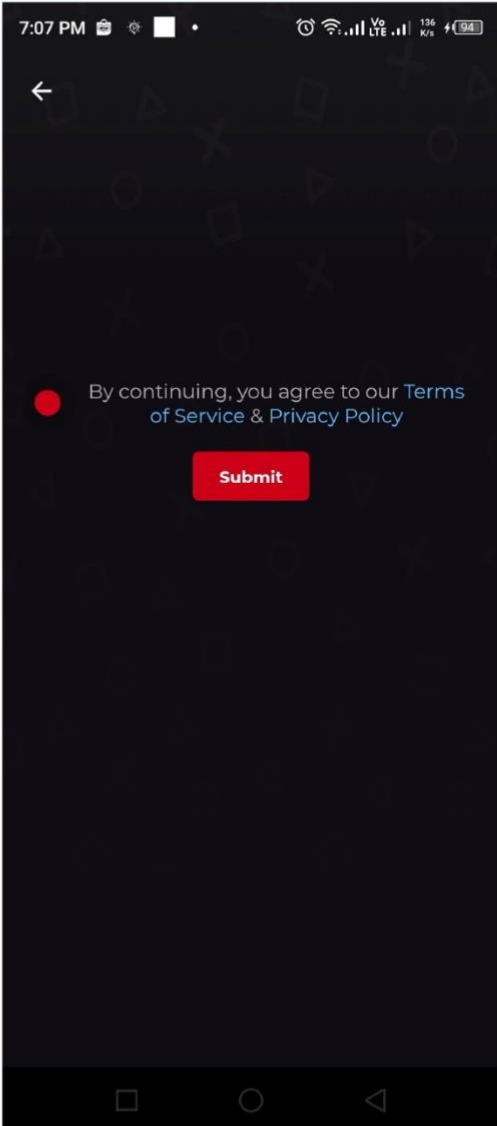
TC002_success_Registration

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for creating a password. At the top, there is a status bar with the time 7:06 PM, signal strength, Wi-Fi, and battery icons. Below the status bar is a navigation bar with a back arrow. The main content area has a dark background with the title "Create a password" in white. Below the title is a subtitle: "Your password must consists at least 8 characters, contain alphanumeric, and special character". There are two input fields: "Set new password" and "Confirm new password", both with masked characters (dots) and a toggle eye icon. At the bottom, there is a red "Continue" button. The Android navigation bar is visible at the very bottom.</p>		

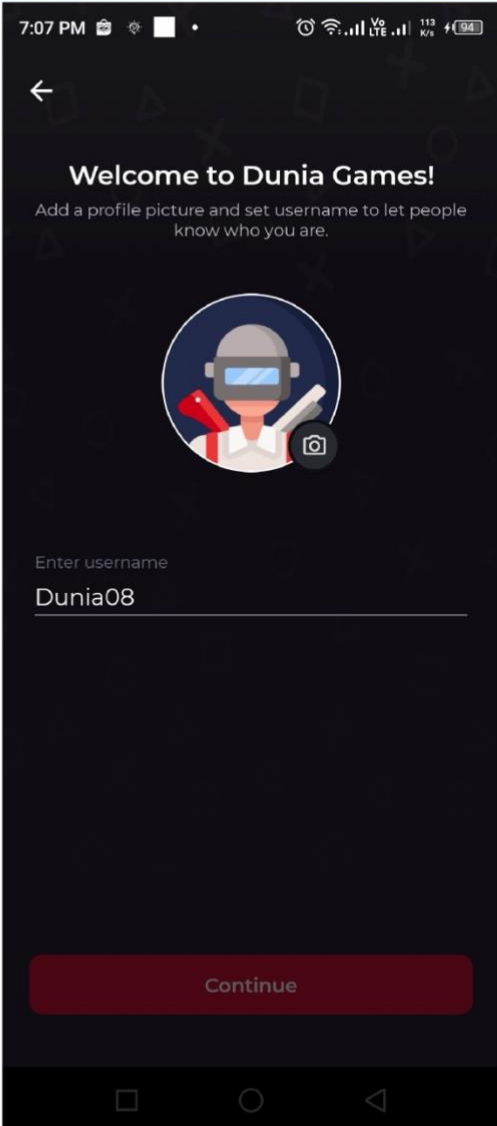
TC002_success_Registration

#	Description	Elapsed	Status
			

TC002_success_Registration

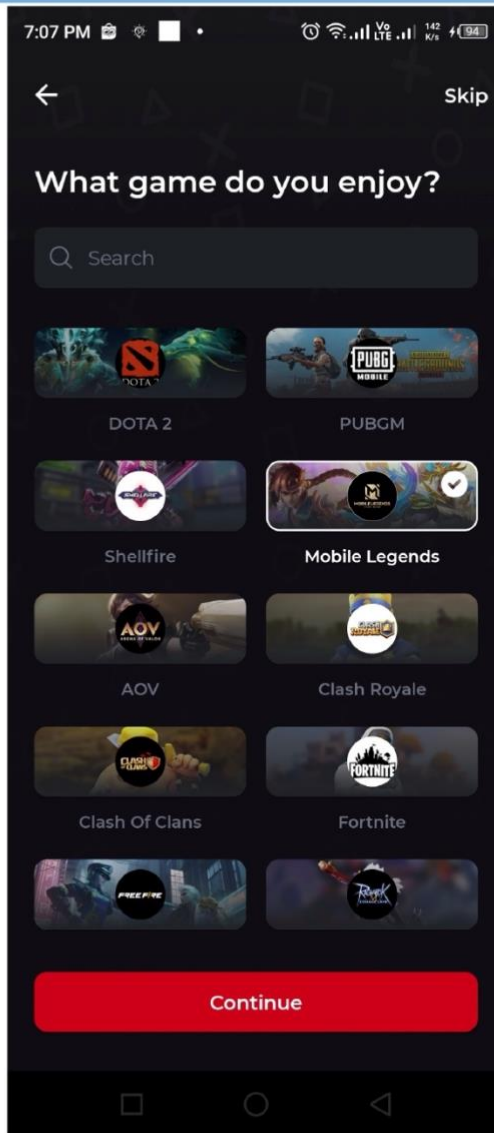
#	Description	Elapsed	Status
			

TC002_success_Registration

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for 'Dunia Games'. At the top, there is a status bar with the time 7:07 PM, signal strength, Wi-Fi, VoLTE, and battery icons. Below the status bar is a dark blue header with a back arrow on the left. The main content area has a dark background with the text 'Welcome to Dunia Games!' in white. Below this, it says 'Add a profile picture and set username to let people know who you are.' There is a circular profile picture placeholder showing a character in a white shirt and red tie, with a camera icon in the bottom right corner. Below the profile picture, there is a text input field with the placeholder text 'Enter username' and the text 'Dunia08' entered. At the bottom of the screen, there is a red button labeled 'Continue'. The Android navigation bar is visible at the very bottom.</p>		

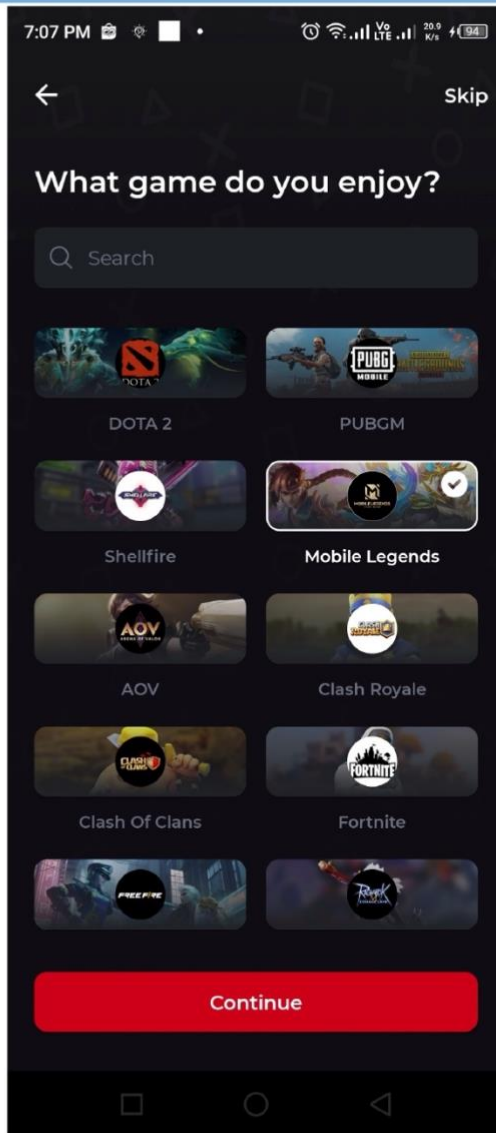
TC002_success_Registration

#	Description	Elapsed	Status
---	-------------	---------	--------



TC002_success_Registration

#	Description	Elapsed	Status
---	-------------	---------	--------



Login

Execution Environment

Host name user - 192.168.1.6
Local OS Mac OS X 64bit
Katalon version 8.8.9.208
Browser null

Summary

ID Test Suites/Dunia Games/Login
Description
Total 2
Passed 2 Failed 0
Error 0 Incomplete 0
Skipped 0
Start 2024-06-20 16:18:54 End 2024-06-20 16:20:06
Elapsed 1m - 11.750s

#	ID	Description	Status
1	Test Cases/Dunia Games/TC003_fail_Login		PASSED
2	Test Cases/Dunia Games/TC004_success_Login		PASSED

TC003_fail_Login

Information

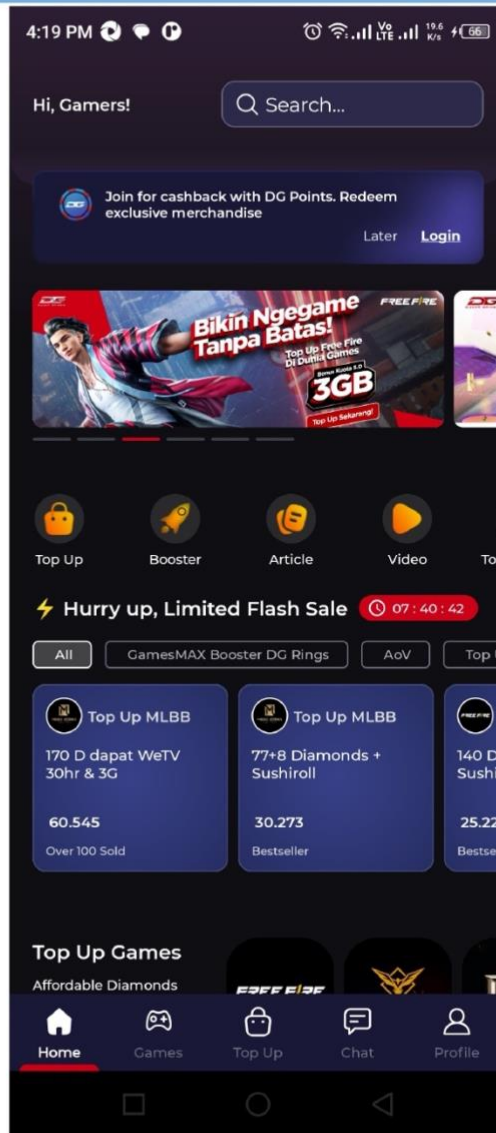
ID Test Cases/Dunia Games/TC003_fail_Login
Description
Tag
Start 2024-06-20 16:18:55 End 2024-06-20 16:19:37
Elapsed 41.918s
Status PASSED

Details

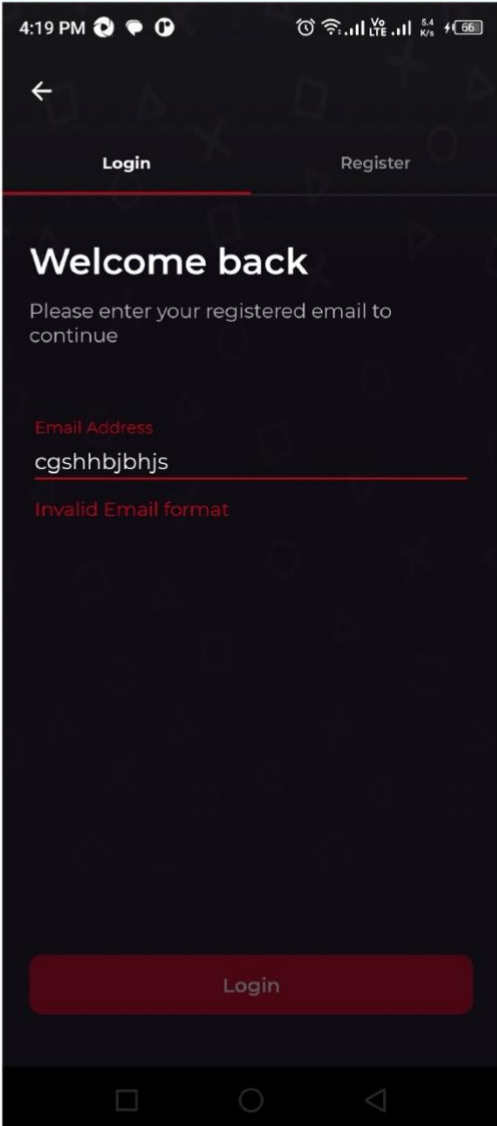
#	Description	Elapsed	Status
1	startLoginApplication()	0.002s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: '1d.co.duniagames'	21.146s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.595s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.103s	PASSED
5	TC003("TC003", device_height, device_width)	0.002s	PASSED
6	comment(tcId + " - Start") TC003 - Start	0.015s	PASSED
7	comment(tcId + " - Expected Result - can see toast error wrong email or password, failed to login") TC003 - Expected Result - can see toast error wrong email or password, failed to login	0.002s	PASSED

TC003_fail_Login

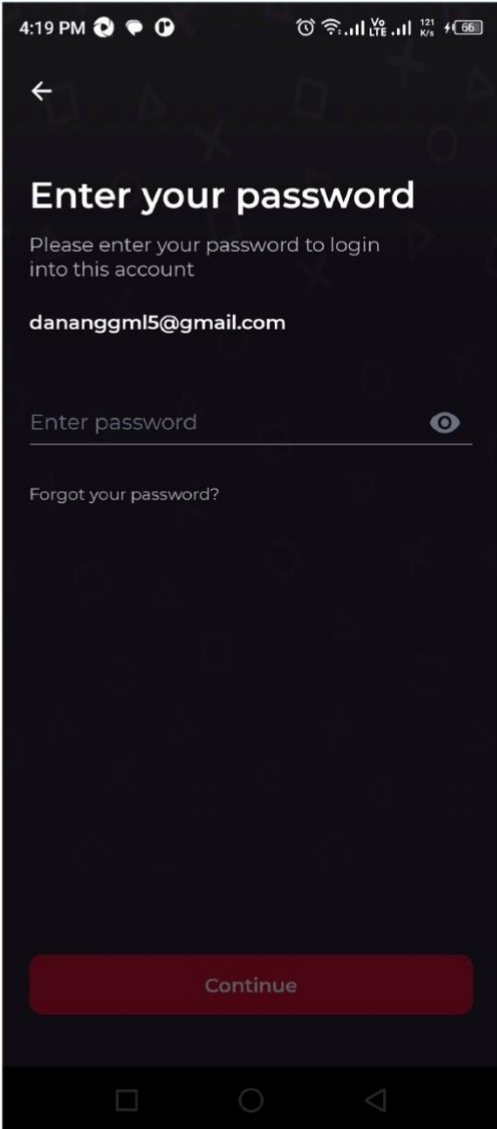
#	Description	Elapsed	Status
---	-------------	---------	--------



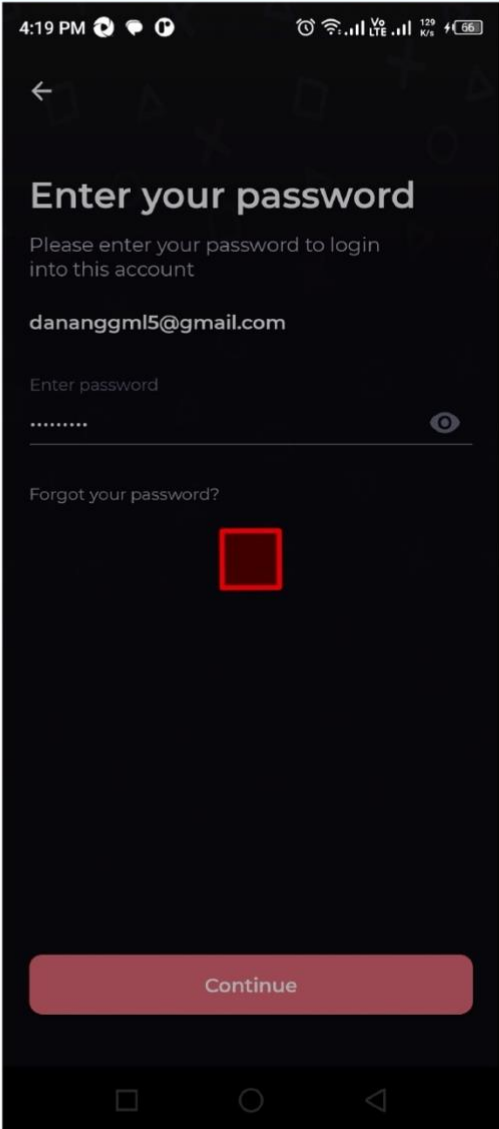
TC003_fail_Login

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for login. At the top, there is a navigation bar with a back arrow on the left and two tabs: "Login" (which is selected and underlined) and "Register". Below the navigation bar, the text "Welcome back" is displayed in a large font, followed by the instruction "Please enter your registered email to continue". There is an input field for the email address with the placeholder text "Email Address" and the value "cgshhbjbhjs". Below the input field, a red error message reads "Invalid Email format". At the bottom of the screen, there is a large red button labeled "Login". The status bar at the very top of the phone shows the time as 4:19 PM, along with various system icons like signal strength, Wi-Fi, and battery level.</p>		

TC003_fail_Login

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for password entry. At the top, the status bar displays the time as 4:19 PM, along with icons for notifications, Wi-Fi, cellular signal, and battery. The app's navigation bar features a back arrow. The main heading is "Enter your password" in a large, bold font. Below it, a subtitle reads "Please enter your password to login into this account". The email address "dananggml5@gmail.com" is displayed. A text input field is labeled "Enter password" and includes a toggle icon for password visibility. A link for "Forgot your password?" is positioned below the input field. At the bottom, a prominent red "Continue" button is visible. The Android navigation bar is shown at the very bottom of the screen.</p>		

TC003_fail_Login

#	Description	Elapsed	Status
			

TC004_success_Login


Information

ID	Test Cases/Dunia Games/TC004_success_Login		
Description			
Tag			
Start	2024-06-20 16:19:37	End	2024-06-20 16:20:06
Elapsed	29.032s		
Status	PASSED		

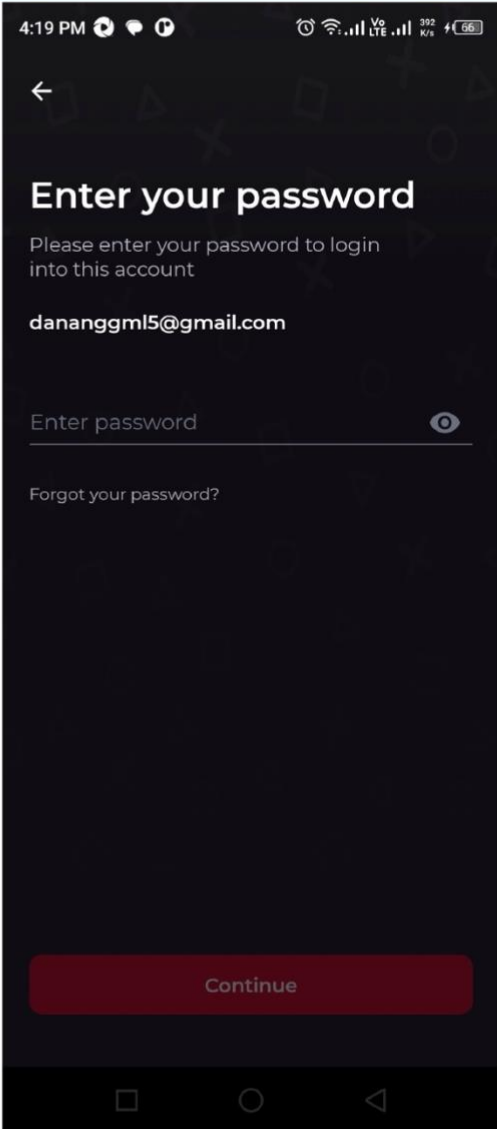
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.000s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	10.190s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.132s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.118s	PASSED
5	TC004("TC004", device_height, device_width)	0.002s	PASSED
6	comment(tcid + " - Start") TC004 - Start	0.002s	PASSED
7	comment(tcid + " - Expected Result - successfull login & direct to homepage ") TC004 - Expected Result - successfull login & direct to homepage	0.001s	PASSED

TC004_success_Login

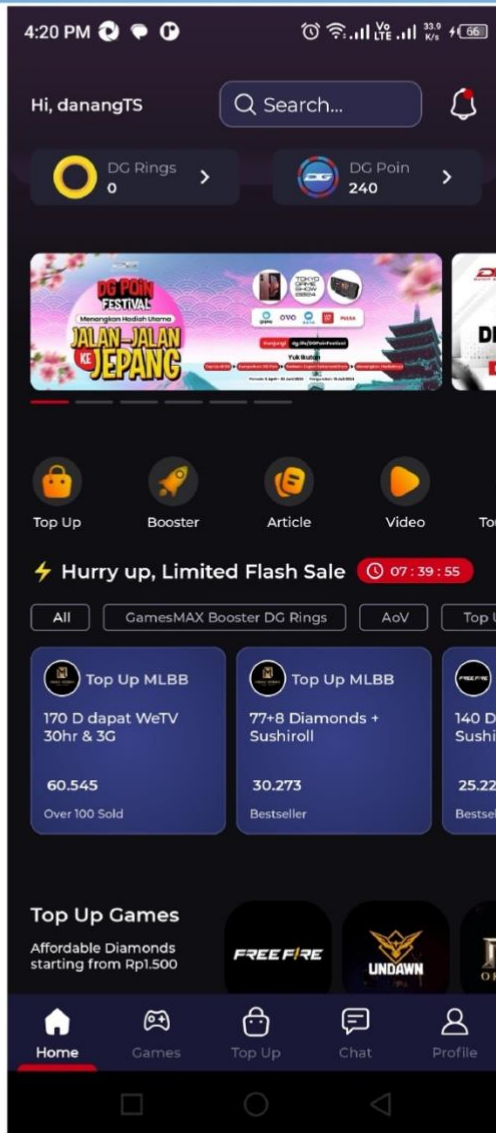
#	Description	Elapsed	Status
			

TC004_success_Login

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for password entry. At the top, the status bar displays the time as 4:19 PM, along with icons for notifications, cellular signal, Wi-Fi, and battery level at 56%. Below the status bar is a dark navigation bar with a white back arrow. The main content area has a dark background with white text. It features a title 'Enter your password', a subtitle 'Please enter your password to login into this account', and the email address 'dananggml5@gmail.com'. A password input field is present with the placeholder text 'Enter password' and a toggle icon for visibility. Below the input field is a link that says 'Forgot your password?'. At the bottom of the screen is a large, rounded red button with the text 'Continue'. The Android navigation bar is visible at the very bottom.</p>		

TC004_success_Login

#	Description	Elapsed	Status
---	-------------	---------	--------



Article

Execution Environment

Host name user - 192.168.1.6
Local OS Mac OS X 64bit
Katalon version 8.6.9.208
Browser null

Summary

ID Test Suites/Dunia Games/Article
Description
Total 1
Passed 1 Failed 0
Error 0 Incomplete 0
Skipped 0
Start 2024-06-20 16:43:02 End 2024-06-20 16:44:09
Elapsed 1m - 7.395s

#	ID	Description	Status
1	Test Cases/Dunia Games/TC005_like_bookmark_and_comment the article		PASSED

TC005_like_bookmark_and_comment the article

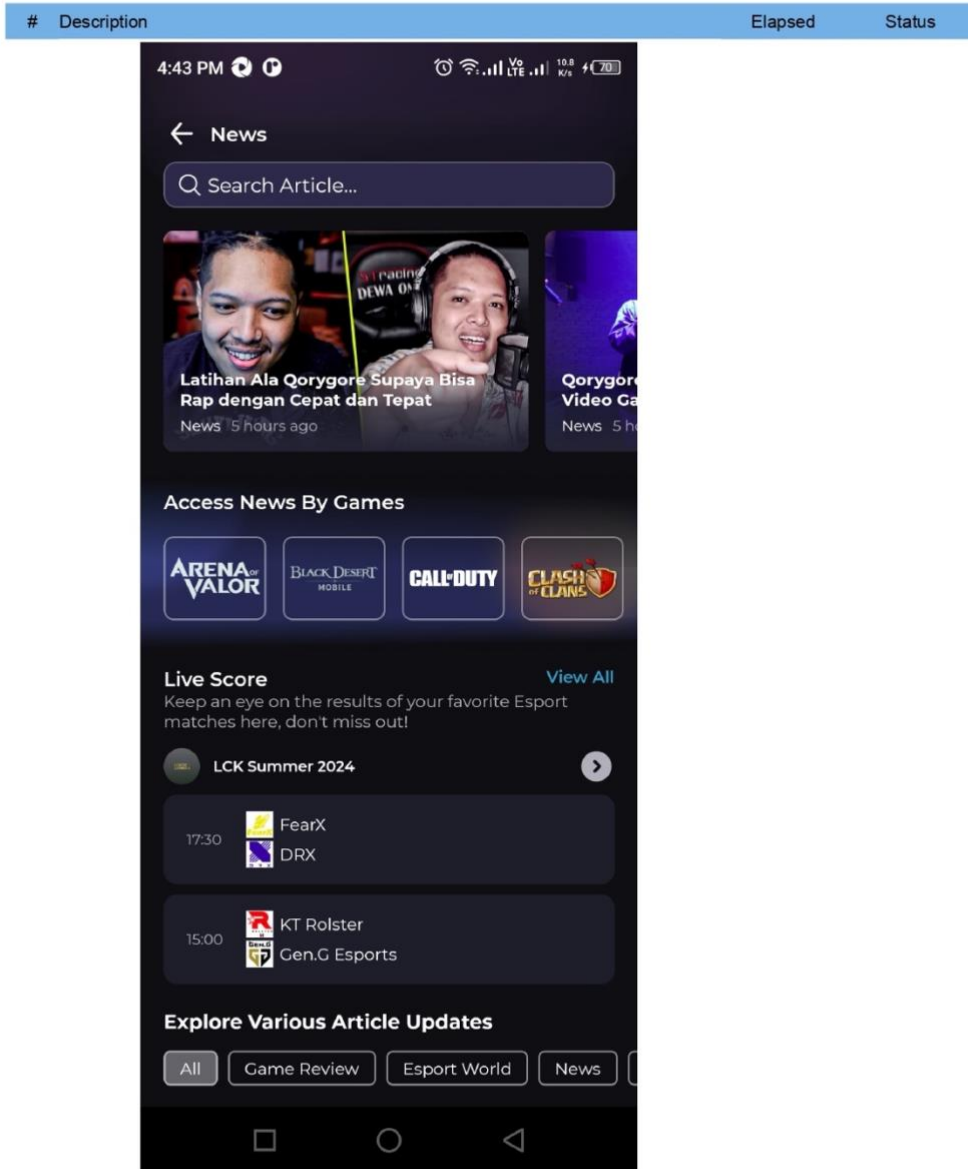
Information

ID Test Cases/Dunia Games/TC005_like_bookmark_and_comment the article
Description
Tag
Start 2024-06-20 16:43:02 End 2024-06-20 16:44:09
Elapsed 1m - 6.481s
Status PASSED

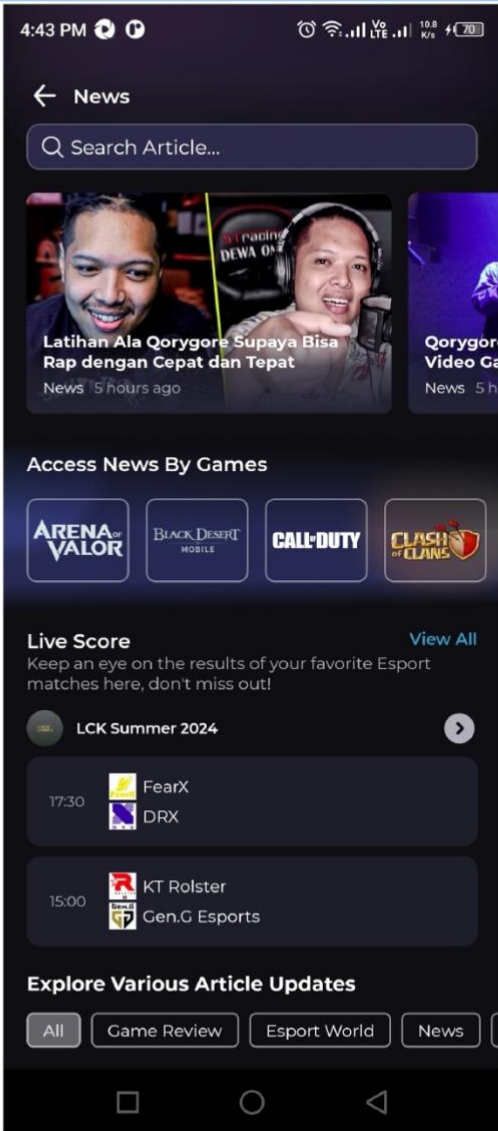
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.001s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: '1d.co.duniagames'	18.557s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.155s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.136s	PASSED
5	TC005("TC005", device_height, device_width)	0.001s	PASSED
6	comment(tcid + " - Start") TC005 - Start	0.016s	PASSED
7	comment(tcid + " - Expected Result - button like will active &button bookmark will active") TC005 - Expected Result - button like will active &button bookmark will active	0.001s	PASSED


TC005_like_bookmark_and_comment the article



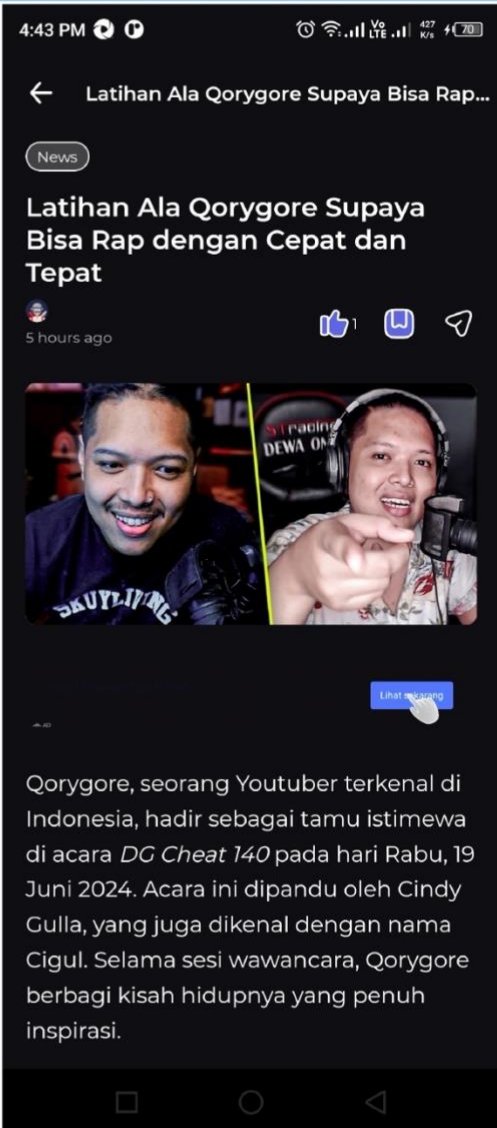

TC005_like_bookmark_and_comment the article

#	Description	Elapsed	Status
	 <p>4:43 PM</p> <p>← News</p> <p>Search Article...</p> <p>Latihan Ala Qorygore Supaya Bisa Rap dengan Cepat dan Tepat News 5 hours ago</p> <p>Qorygore Video Ga News 5 h</p> <p>Access News By Games</p> <p>ARENA OF VALOR BLACK DESERT MOBILE CALL OF DUTY CLASH OF CLANS</p> <p>Live Score View All Keep an eye on the results of your favorite Esport matches here, don't miss out!</p> <p>LCK Summer 2024</p> <p>17:30 FearX vs DRX</p> <p>15:00 KT Rolster vs Gen.G Esports</p> <p>Explore Various Article Updates</p> <p>All Game Review Esport World News</p>		

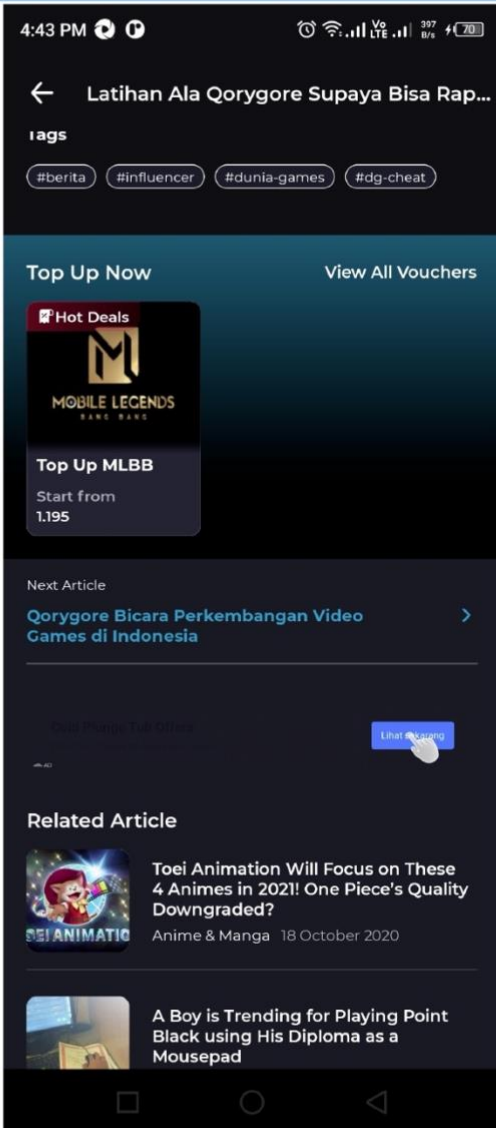
TC005_like_bookmark_and_comment the article

#	Description	Elapsed	Status
	<p>4:43 PM</p> <p>← Latihan Ala Qorygore Supaya Bisa Rap...</p> <p>News</p> <h3>Latihan Ala Qorygore Supaya Bisa Rap dengan Cepat dan Tepat</h3> <p>5 hours ago</p>  <p>Qorygore, seorang Youtuber terkenal di Indonesia, hadir sebagai tamu istimewa di acara <i>DG Cheat 140</i> pada hari Rabu, 19 Juni 2024. Acara ini dipandu oleh Cindy Gulla, yang juga dikenal dengan nama Cigul. Selama sesi wawancara, Qorygore berbagi kisah hidupnya yang penuh inspirasi.</p>		

TC005_like_bookmark_and_comment the article

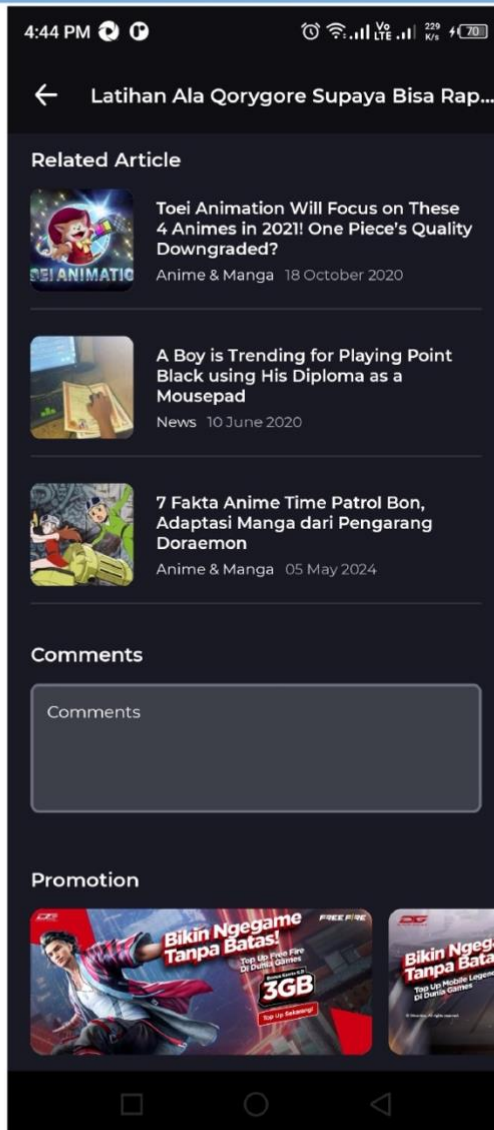
#	Description	Elapsed	Status
	 <p>4:43 PM</p> <p>← Latihan Ala Qorygore Supaya Bisa Rap...</p> <p>News</p> <h3>Latihan Ala Qorygore Supaya Bisa Rap dengan Cepat dan Tepat</h3> <p>5 hours ago</p>  <p>Lihat selengkapnya</p> <p>Qorygore, seorang YouTuber terkenal di Indonesia, hadir sebagai tamu istimewa di acara <i>DG Cheat 140</i> pada hari Rabu, 19 Juni 2024. Acara ini dipandu oleh Cindy Gulla, yang juga dikenal dengan nama Cigul. Selama sesi wawancara, Qorygore berbagi kisah hidupnya yang penuh inspirasi.</p>		

TC005_like_bookmark_and_comment the article

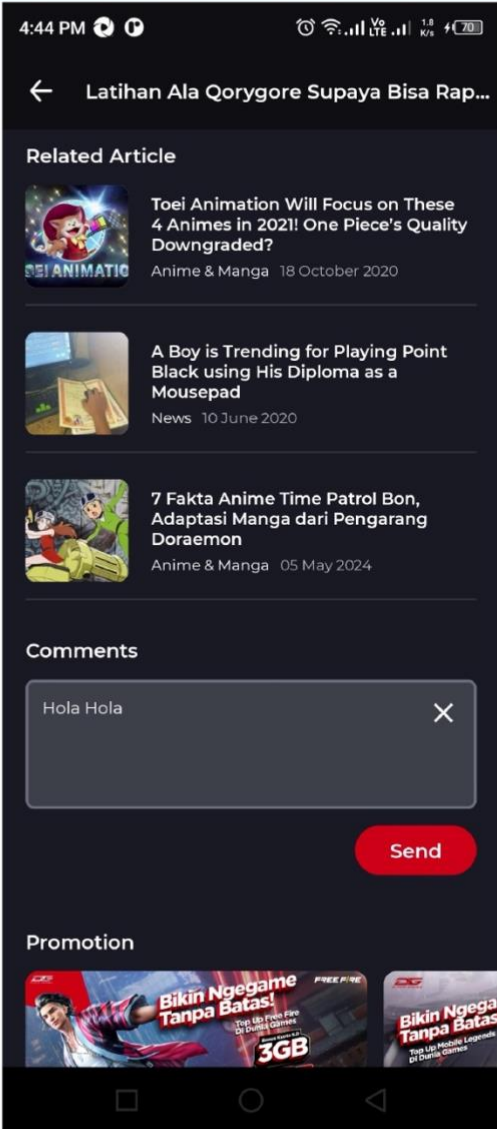





#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface. At the top, the time is 4:43 PM. The article title is "Latihan Ala Qorygore Supaya Bisa Rap...". Below the title are tags: #berita, #influencer, #dunia-games, and #dg-cheat. A "Top Up Now" section features a "Hot Deals" banner for "MOBILE LEGENDS" with a "Top Up MLBB" button and a price of "Start from 1.195". A "Next Article" section is titled "Qorygore Bicara Perkembangan Video Games di Indonesia". A "Related Article" section includes two items: "Toei Animation Will Focus on These 4 Animes in 2021! One Piece's Quality Downgraded?" and "A Boy is Trending for Playing Point Black using His Diploma as a Mousepad".</p>		

TC005_like_bookmark_and_comment the article

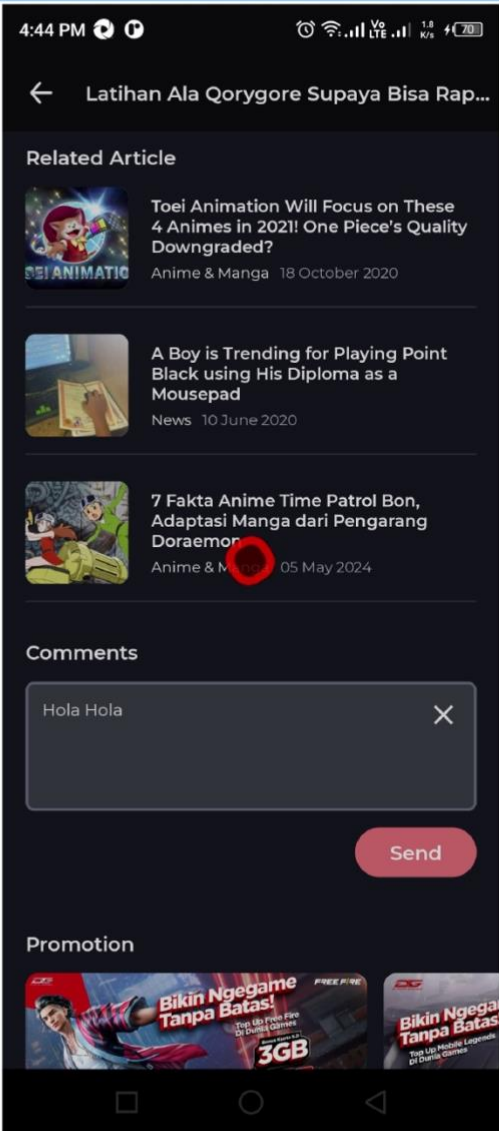
#	Description	Elapsed	Status
---	-------------	---------	--------



TC005_like_bookmark_and_comment the article

#	Description	Elapsed	Status
	 <p>4:44 PM</p> <p>← Latihan Ala Qorygore Supaya Bisa Rap...</p> <p>Related Article</p> <p> Toei Animation Will Focus on These 4 Animes in 2021! One Piece's Quality Downgraded? Anime & Manga · 18 October 2020</p> <p> A Boy is Trending for Playing Point Black using His Diploma as a Mousepad News · 10 June 2020</p> <p> 7 Fakta Anime Time Patrol Bon, Adaptasi Manga dari Pengarang Doraemon Anime & Manga · 05 May 2024</p> <p>Comments</p> <p>Hola Hola</p> <p>Send</p> <p>Promotion</p> <p> </p>		

TC005_like_bookmark_and_comment the article

#	Description	Elapsed	Status
			

Gamefeed

Execution Environment

Host name user - 192.168.1.6
Local OS Mac OS X 64bit
Katalon version 8.6.9.208
Browser null

Summary

ID Test Suites/Dunia Games/Gamefeed
Description
Total 2
Passed 2 Failed 0
Error 0 Incomplete 0
Skipped 0
Start 2024-06-20 16:24:03 End 2024-06-20 16:25:34
Elapsed 1m - 31.885s

#	ID	Description	Status
1	Test Cases/Dunia Games/TC006_create post gamefeed		PASSED
2	Test Cases/Dunia Games/TC007_like and Comment the post		PASSED

TC006_create post gamefeed

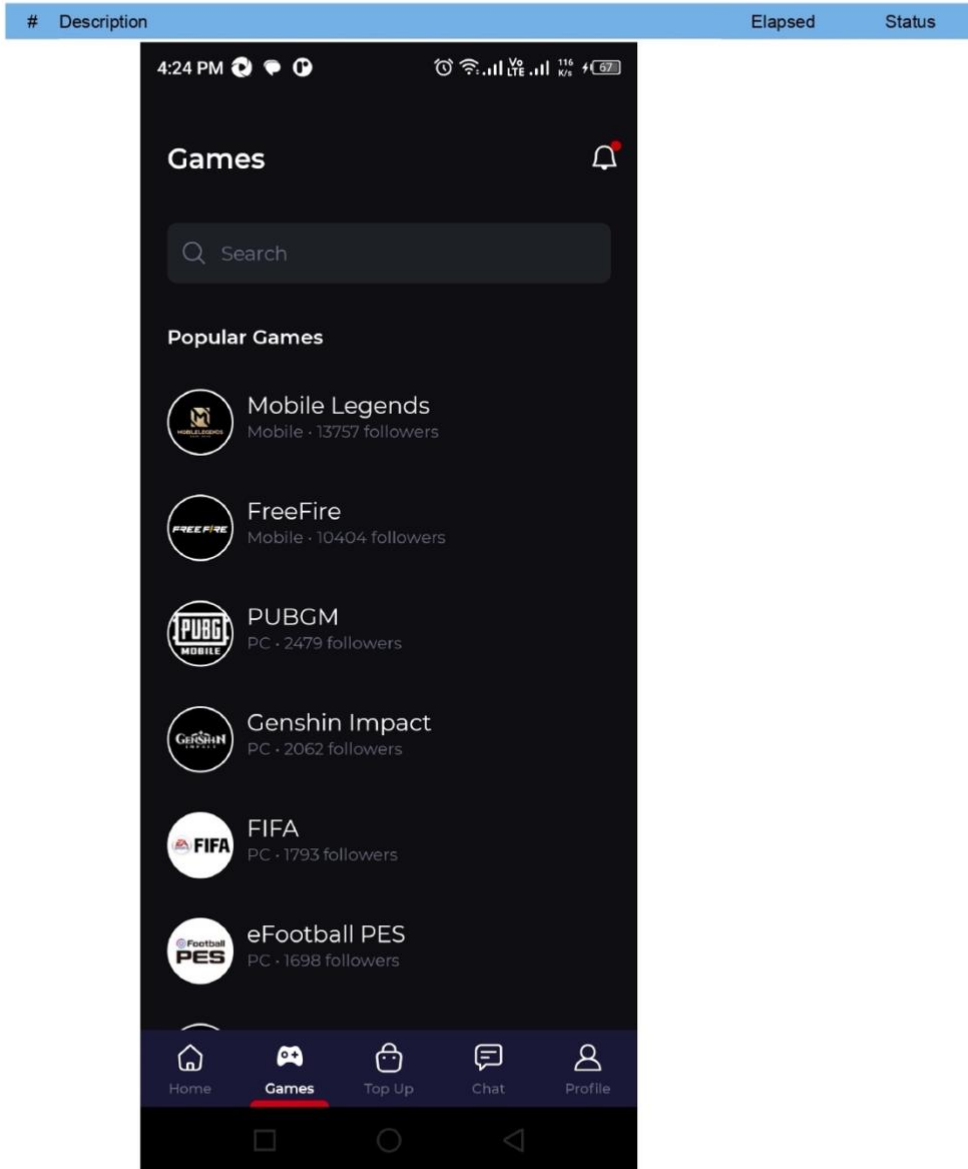
Information

ID Test Cases/Dunia Games/TC006_create post gamefeed
Description
Tag
Start 2024-06-20 16:24:03 End 2024-06-20 16:24:41
Elapsed 37.750s
Status PASSED

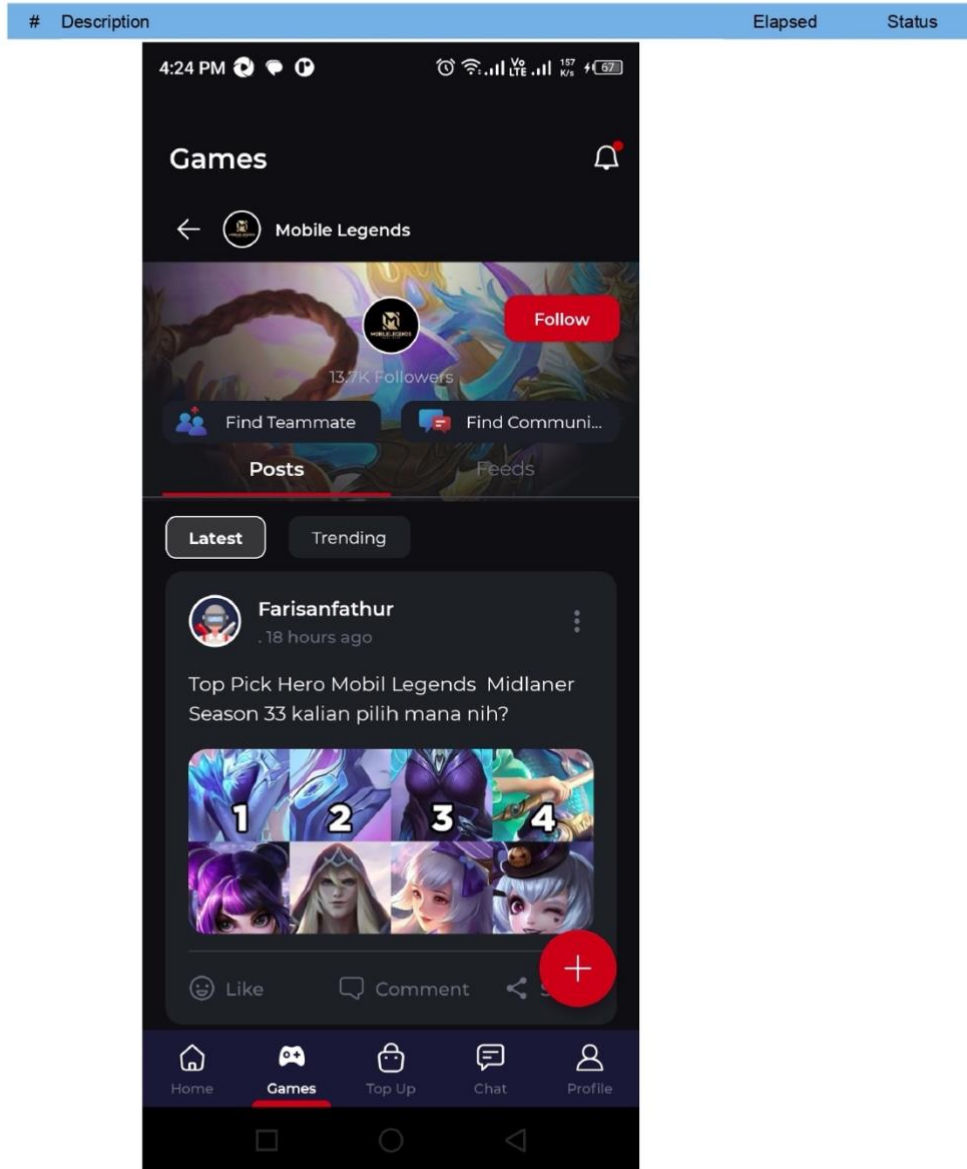
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.002s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	18.601s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.195s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.136s	PASSED
5	TC006("TC006", device_height, device_width)	0.002s	PASSED
6	comment(tcid + " - Start") TC006 - Start	0.016s	PASSED
7	comment(tcid + " - Expected Result - user succesfully create the post") TC006 - Expected Result - user succesfully create the post	0.003s	PASSED

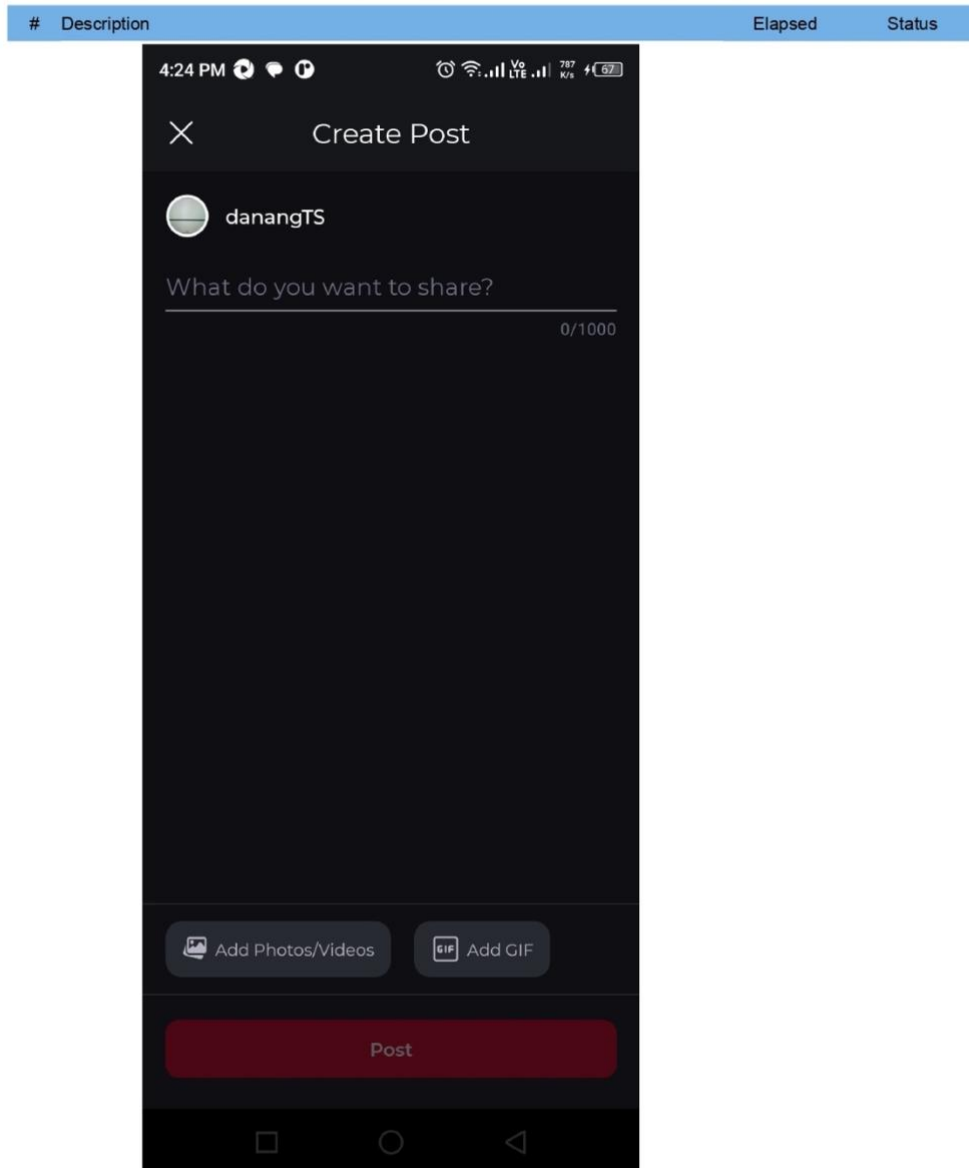
TC006_create post gamefeed



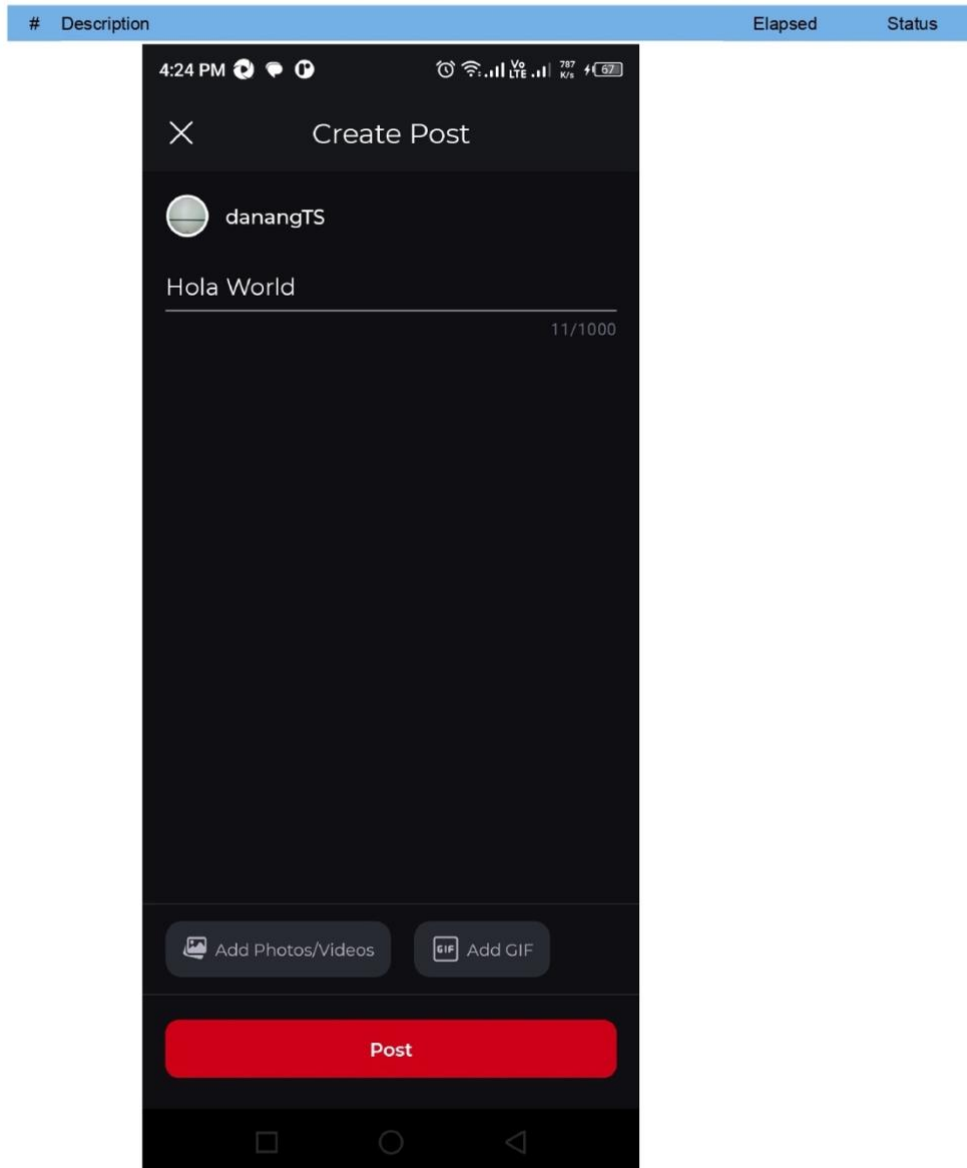
TC006_create post gamefeed



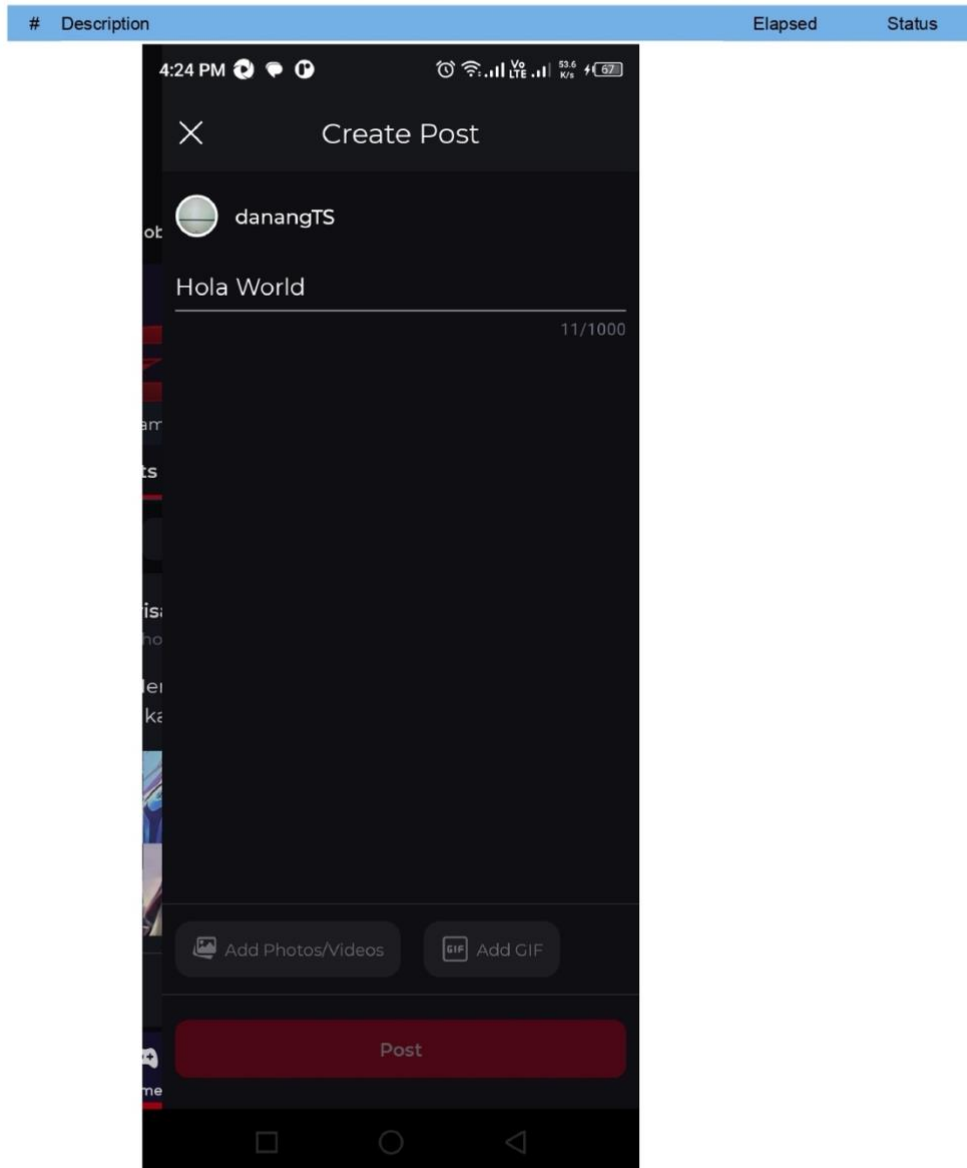
TC006_create post gamefeed



TC006_create post gamefeed



TC006_create post gamefeed



TC007_like and Comment the post

Information

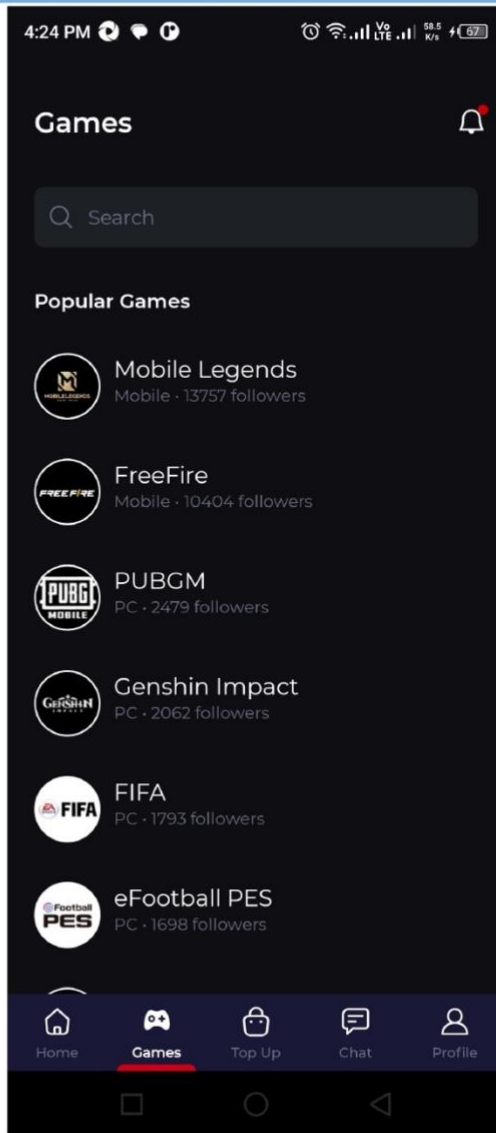
ID	Test Cases/Dunia Games/TC007_like and Comment the post		
Description			
Tag			
Start	2024-06-20 16:24:41	End	2024-06-20 16:25:34
Elapsed	53.335s		
Status	PASSED		

Details

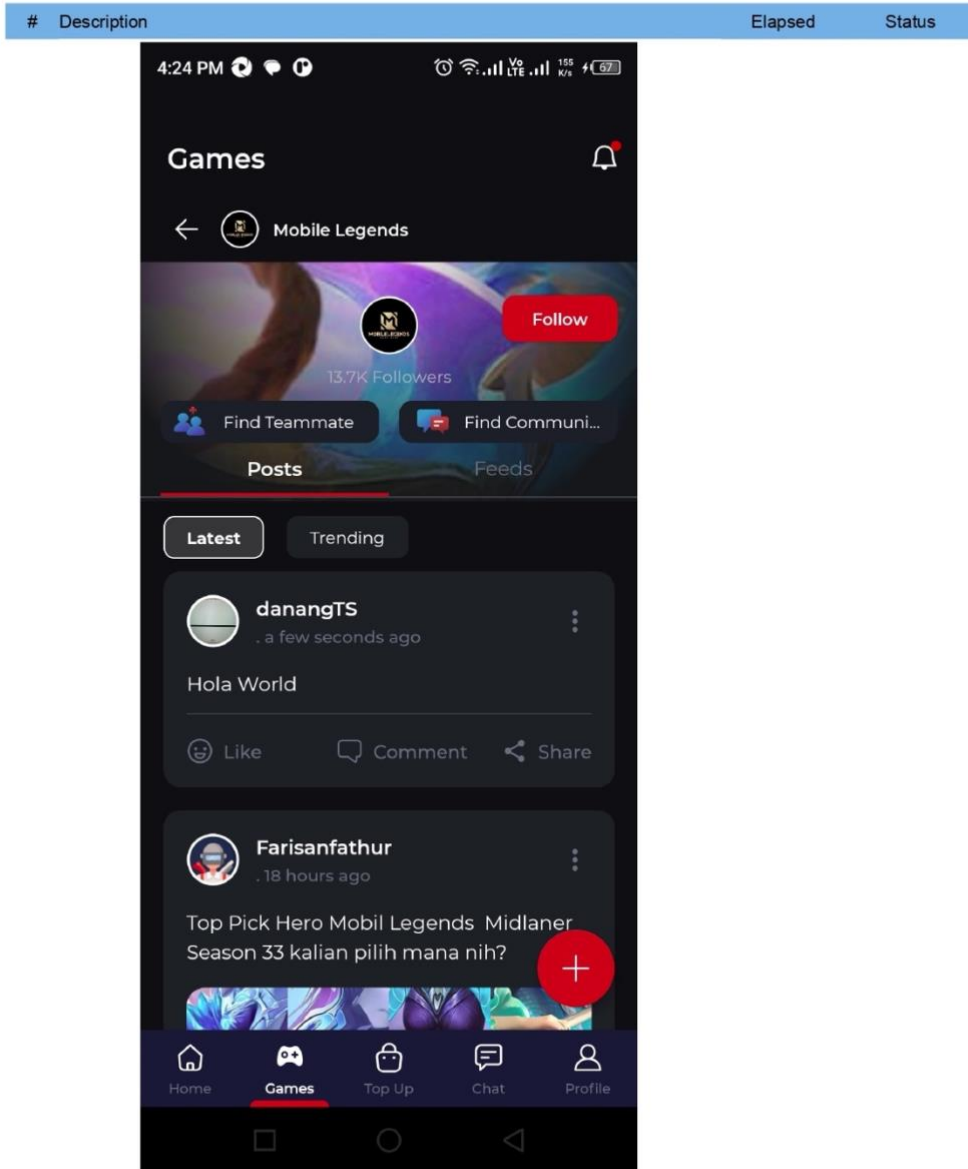
#	Description	Elapsed	Status
1	startLoginApplication()	0.000s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	10.649s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.142s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.118s	PASSED
5	TC007("TC007", device_height, device_width)	0.002s	PASSED
6	comment(tcid + " - Start") TC007 - Start	0.002s	PASSED
7	comment(tcid + " - Expected Result - user succesfully like and comment the post") TC007 - Expected Result - user succesfully like and comment the post	0.001s	PASSED

TC007_like and Comment the post

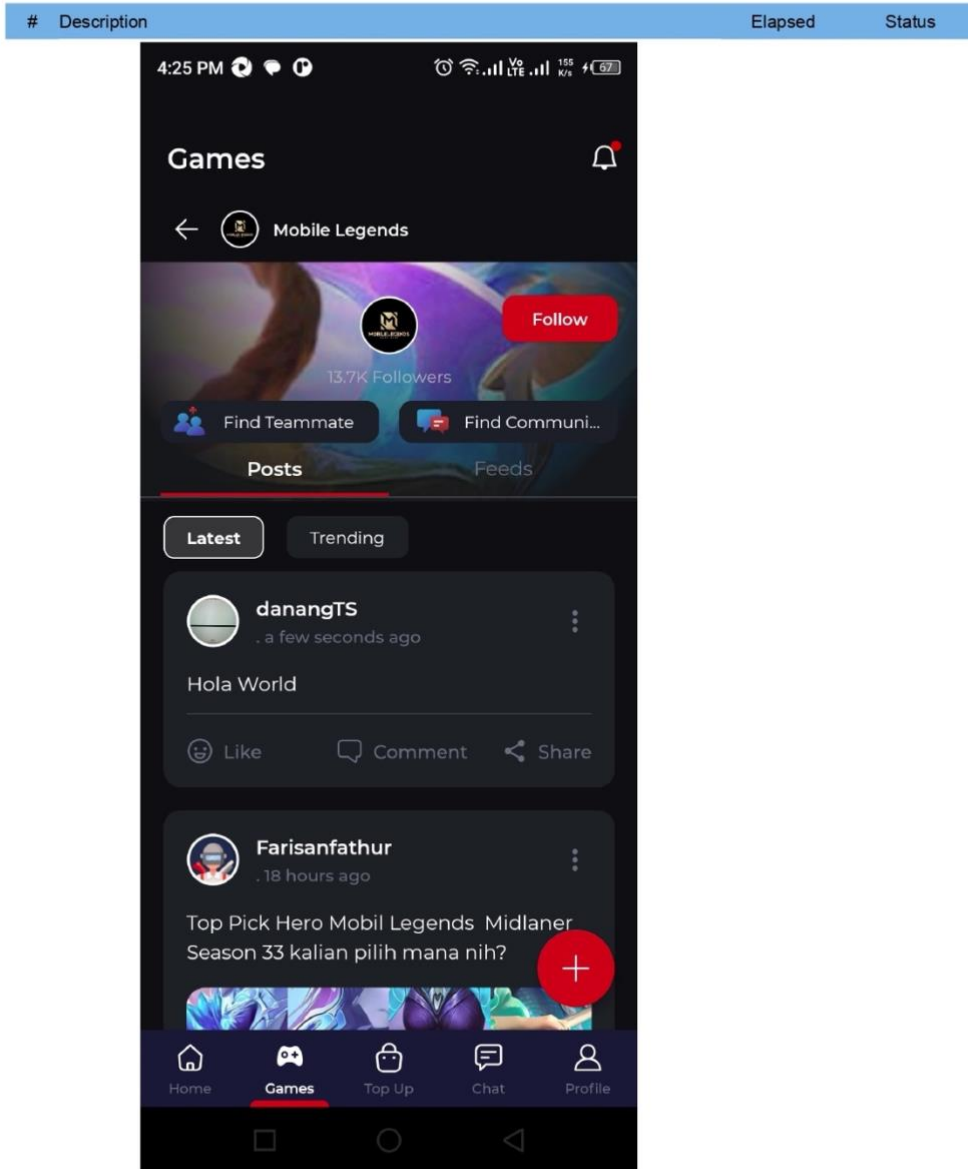
Description Elapsed Status



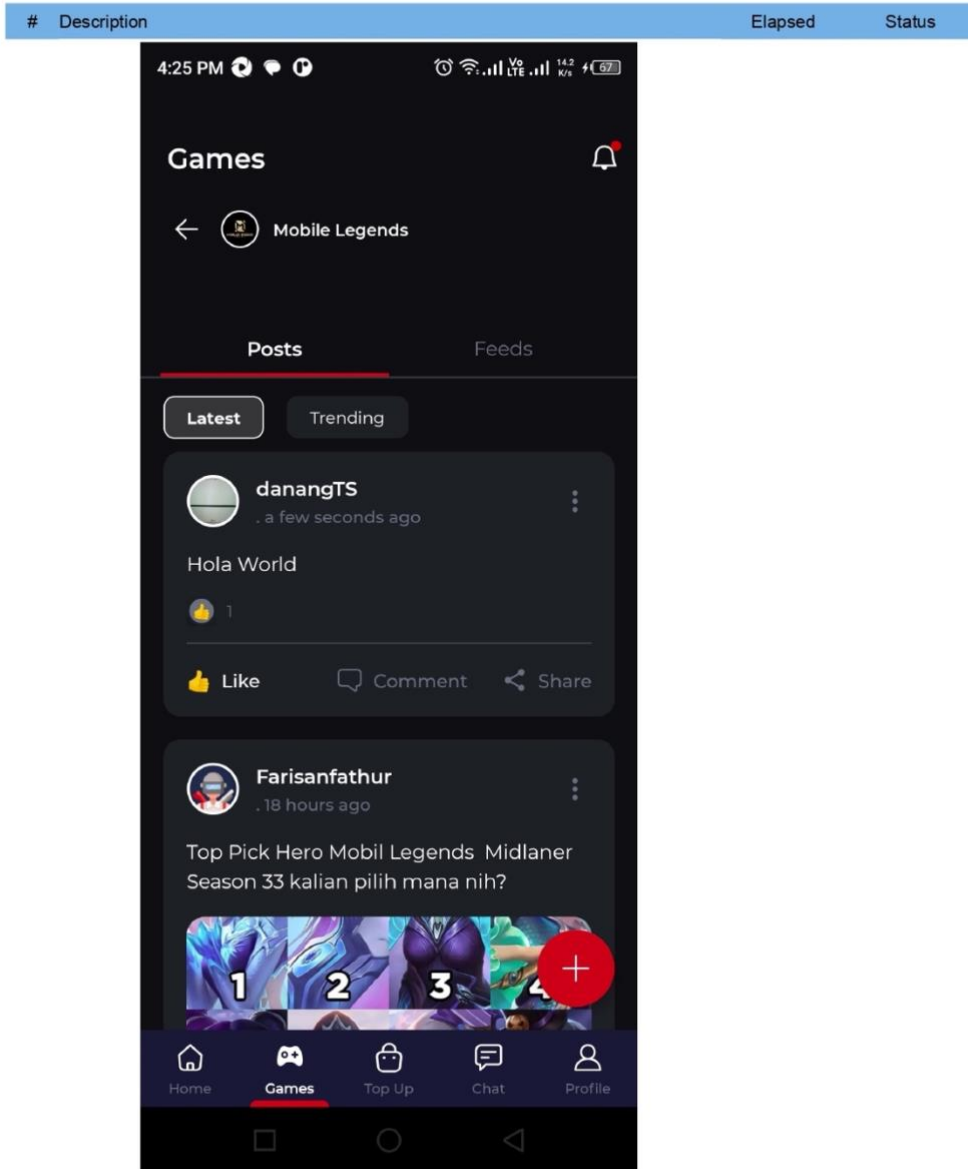
TC007_like and Comment the post



TC007_like and Comment the post



TC007_like and Comment the post



Tournament

Execution Environment

Host name user - 192.168.1.6
Local OS Mac OS X 64bit
Katalon version 8.6.9.208
Browser null

Summary

ID Test Suites/Dunia Games/Tournament
Description
Total 1
Passed 1 Failed 0
Error 0 Incomplete 0
Skipped 0
Start 2024-06-20 19:11:35 End 2024-06-20 19:12:58
Elapsed 1m - 22.887s

#	ID	Description	Status
1	Test Cases/Dunia Games/TC008_create Tournament		PASSED

TC008_create Tournament

Information

ID Test Cases/Dunia Games/TC008_create Tournament
Description
Tag
Start 2024-06-20 19:11:36 End 2024-06-20 19:12:58
Elapsed 1m - 22.197s
Status PASSED

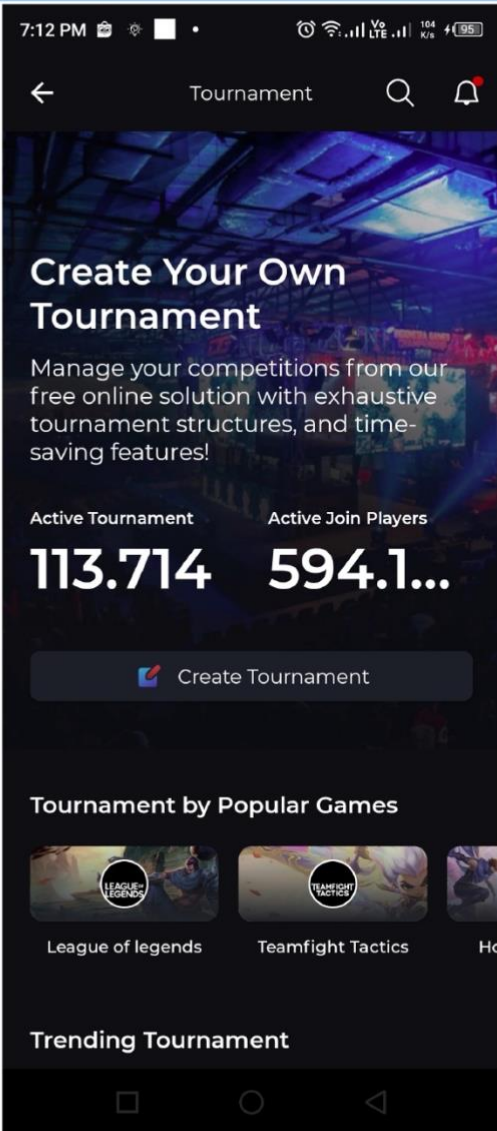
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.001s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	18.082s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.137s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.085s	PASSED
5	TC008("TC008", device_height, device_width)	0.002s	PASSED
6	comment(tcId + " - Start") TC008 - Start	0.018s	PASSED
7	comment(tcId + " - Expected Result - there button join tournament & there registration date") TC008 - Expected Result - there button join tournament & there registration date	0.001s	PASSED

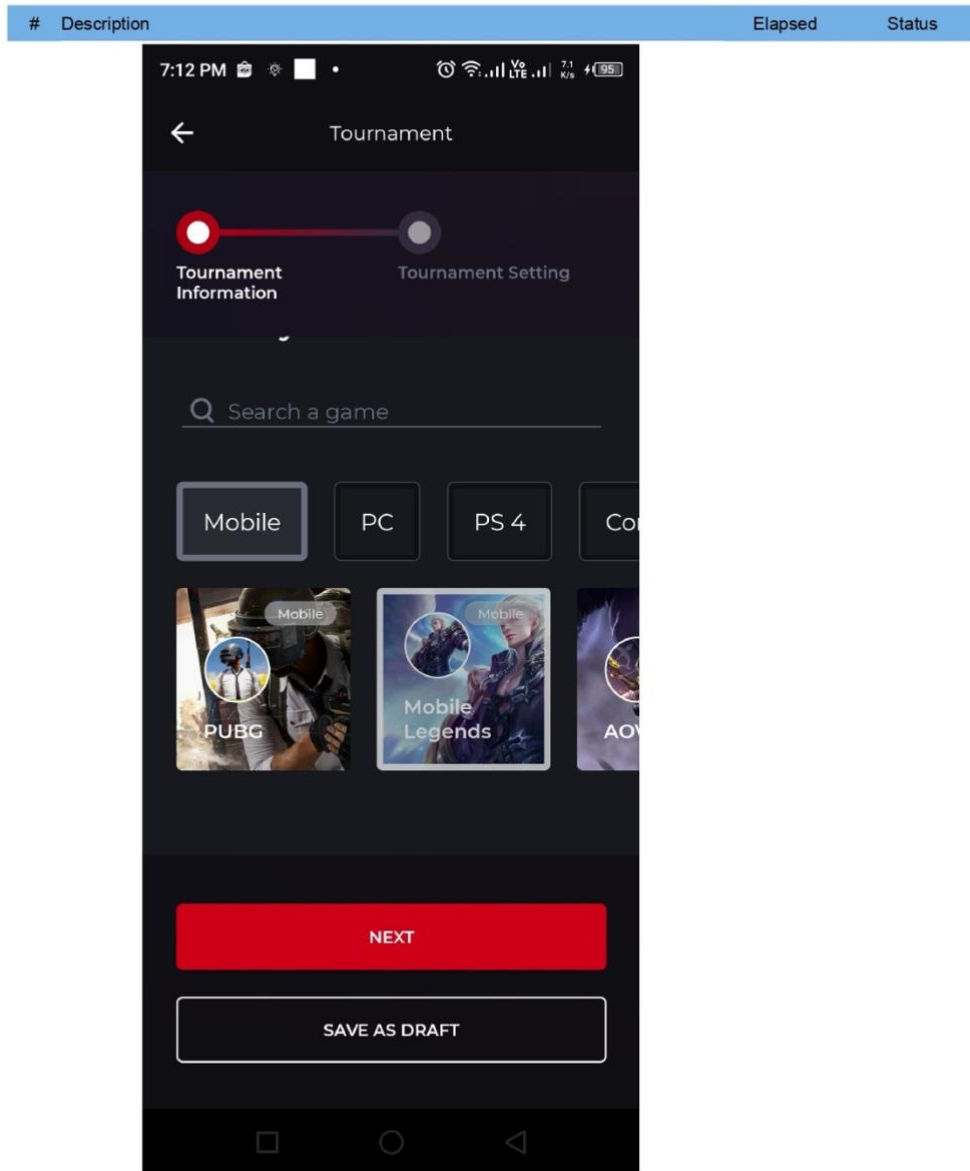
TC008_create Tournament

#	Description	Elapsed	Status
9	<pre> try Failed to check for element 'Object Repository/Tournament/android.widget.TextView - Participants' exist (Root cause: com.kms.katalon.core.exception.StepFailedException: Element 'Object Repository/Tournament/android.widget.TextView - Participants' not found at com.kms.katalon.core.keyword.internal.KeywordMain.stepFailed(KeywordMain.groovy:51) at com.kms.katalon.core.mobile.keyword.internal.MobileKeywordMain.stepFailed(MobileKeywordMain.groovy:40) at com.kms.katalon.core.mobile.keyword.builtin.VerifyElementExistKeyword\$_verifyElementExist_closure1.doCall(VerifyElementExistKeyword.groovy:77) at com.kms.katalon.core.mobile.keyword.builtin.VerifyElementExistKeyword\$_verifyElementExist_closure1.call(VerifyElementExistKeyword.groovy) at com.kms.katalon.core.mobile.keyword.internal.MobileKeywordMain.runKeyword(MobileKeywordMain.groovy:21) at com.kms.katalon.core.mobile.keyword.builtin.VerifyElementExistKeyword.verifyElementExist(VerifyElementExistKeyword.groovy:80) at com.kms.katalon.core.mobile.keyword.builtin.VerifyElementExistKeyword.execute(VerifyElementExistKeyword.groovy:84) at com.kms.katalon.core.keyword.internal.KeywordExecutor.executeKeywordForPlatform(KeywordExecutor.groovy:74) at com.kms.katalon.core.mobile.keyword.MobileBuiltInKeywords.verifyElementExist(MobileBuiltInKeywords.groovy:1739) at com.kms.katalon.core.mobile.keyword.MobileBuiltInKeywords.verifyElementExist\$3.call(Unknown Source) at TC008_create Tournament.TC008(TC008_create Tournament:124) at Script1718638673658\$TC008.callCurrent(Unknown Source) at TC008_create Tournament.run(TC008_create Tournament:27) at com.kms.katalon.core.main.ScriptEngine.run(ScriptEngine.java:194) at com.kms.katalon.core.main.ScriptEngine.runScriptAsRawText(ScriptEngine.java:119) at com.kms.katalon.core.main.TestCaseExecutor.runScript(TestCaseExecutor.java:448) at com.kms.katalon.core.main.TestCaseExecutor.doExecute(TestCaseExecutor.java:439) at com.kms.katalon.core.main.TestCaseExecutor.processExecutionPhase(TestCaseExecutor.java:418) at com.kms.katalon.core.main.TestCaseExecutor.accessMainPhase(TestCaseExecutor.java:410) at com.kms.katalon.core.main.TestCaseExecutor.execute(TestCaseExecutor.java:285) at com.kms.katalon.core.common.CommonExecutor.accessTestCaseMainPhase(CommonExecutor.java:65) at com.kms.katalon.core.main.TestSuiteExecutor.accessTestSuiteMainPhase(TestSuiteExecutor.java:148) at com.kms.katalon.core.main.TestSuiteExecutor.execute(TestSuiteExecutor.java:106) at com.kms.katalon.core.main.TestCaseMain.startTestSuite(TestCaseMain.java:187) at com.kms.katalon.core.main.TestCaseMain\$startTestSuite\$0.call(Unknown Source) at TempTestSuite1718885485937.run(TempTestSuite1718885485937.groovy:35)) </pre>	1m - 3.074s	FAILED

TC008_create Tournament

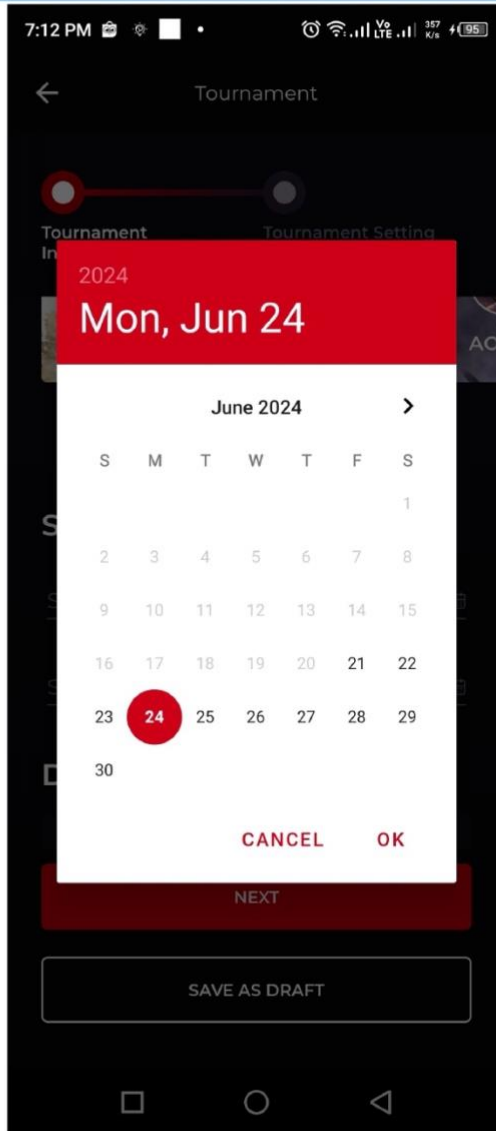
#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for creating tournaments. At the top, the status bar displays the time as 7:12 PM, along with various system icons. The app's navigation bar includes a back arrow, the title 'Tournament', a search icon, and a notification bell. The main content area features a large header 'Create Your Own Tournament' with a sub-header 'Manage your competitions from our free online solution with exhaustive tournament structures, and time-saving features!'. Below this, two statistics are shown: 'Active Tournament' with the value '113.714' and 'Active Join Players' with the value '594.1...'. A prominent 'Create Tournament' button is located below the statistics. The interface also includes a section titled 'Tournament by Popular Games' with three game icons: 'League of legends', 'Teamfight Tactics', and 'Ho...'. At the bottom, there is a 'Trending Tournament' section. The Android navigation bar is visible at the very bottom of the screen.</p>		

TC008_create Tournament



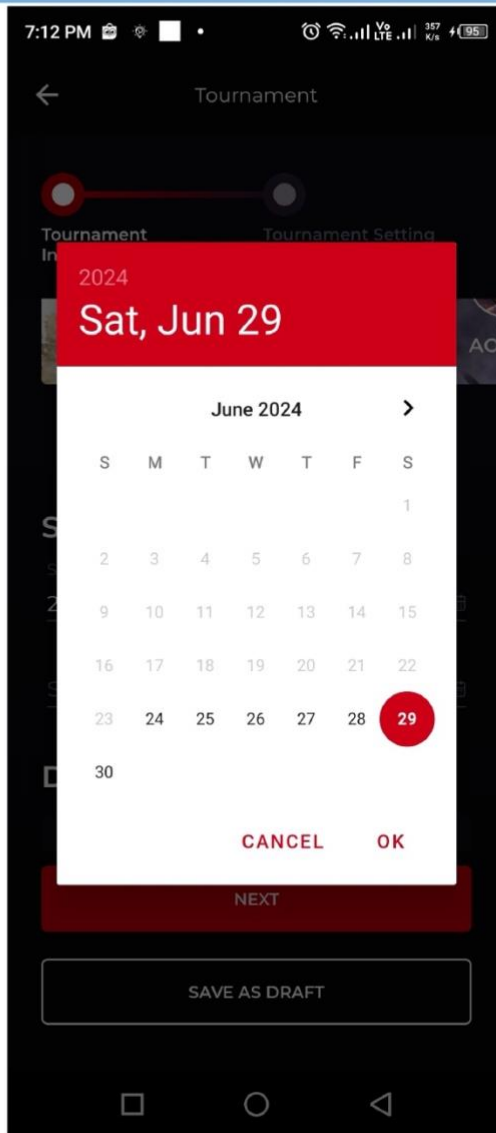
TC008_create Tournament

#	Description	Elapsed	Status
---	-------------	---------	--------

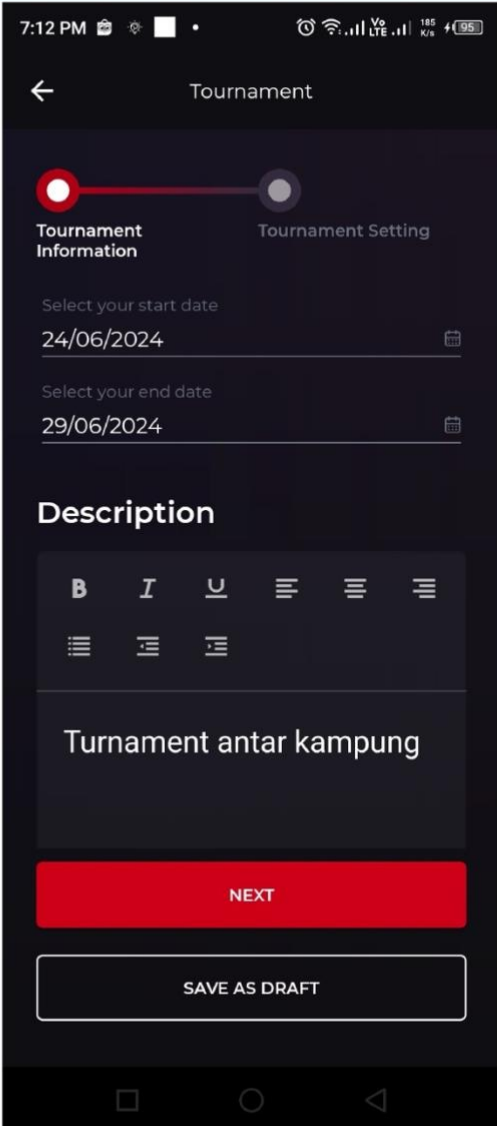


TC008_create Tournament

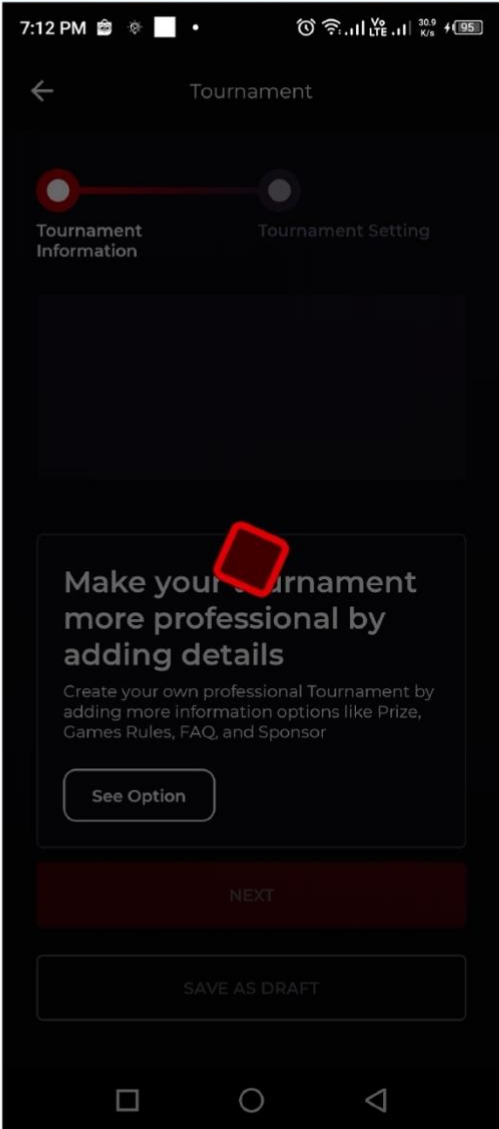
#	Description	Elapsed	Status
---	-------------	---------	--------



TC008_create Tournament

#	Description	Elapsed	Status
			

TC008_create Tournament

#	Description	Elapsed	Status
			

RoomChat

Execution Environment

Host name user - 192.168.1.6
Local OS Mac OS X 64bit
Katalon version 8.6.9.208
Browser null

Summary

ID Test Suites/Dunia Games/RoomChat
Description
Total 2
Passed 2 Failed 0
Error 0 Incomplete 0
Skipped 0
Start 2024-06-20 16:36:10 End 2024-06-20 16:37:35
Elapsed 1m - 24.421s

#	ID	Description	Status
1	Test Cases/Dunia Games/TC009_create Roomchat		PASSED
2	Test Cases/Dunia Games/TC010_send message to room chat		PASSED

TC009_create Roomchat

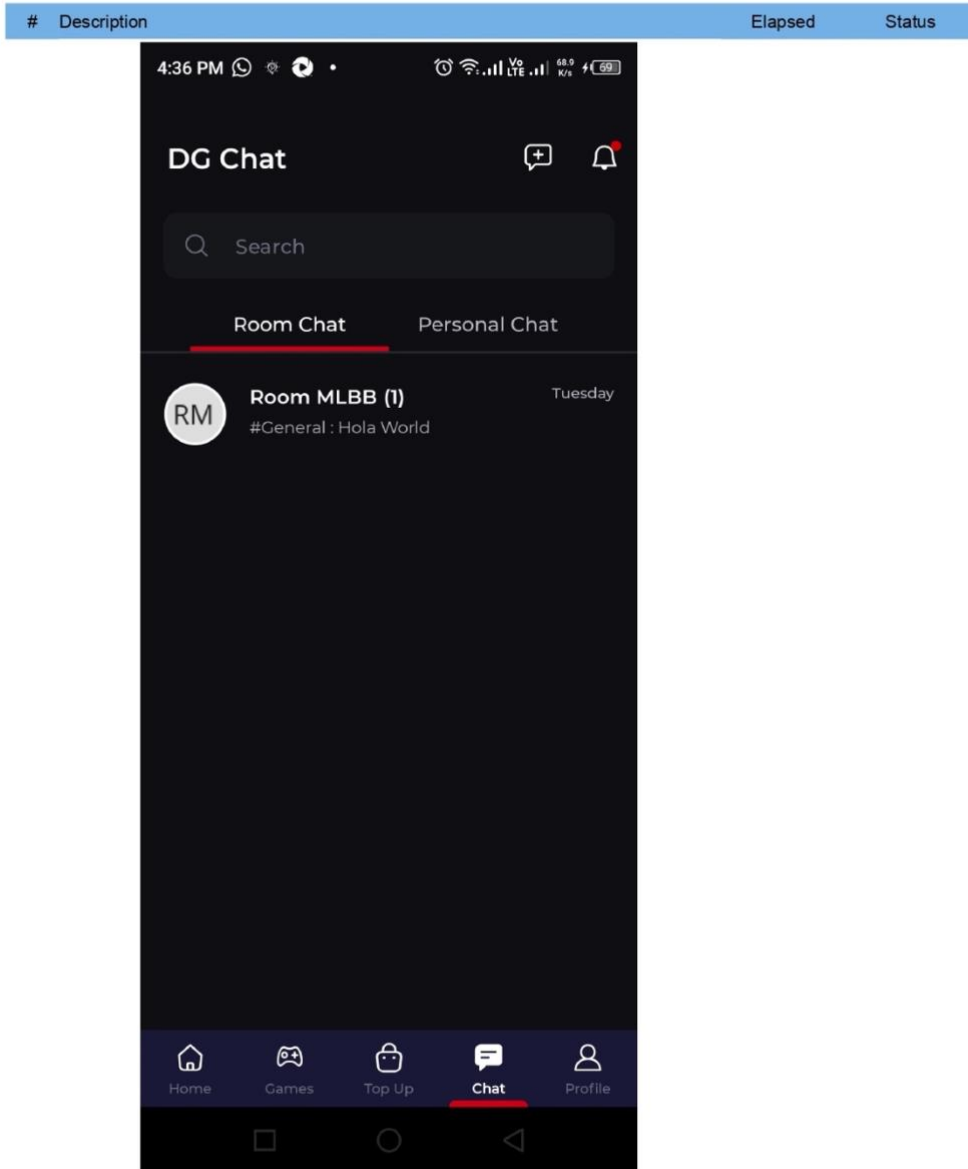
Information

ID Test Cases/Dunia Games/TC009_create Roomchat
Description
Tag
Start 2024-06-20 16:36:11 End 2024-06-20 16:37:07
Elapsed 56.062s
Status PASSED

Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.002s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: '1d.co.duniagames'	17.123s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.129s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.117s	PASSED
5	TC009("TC009", device_height, device_width)	0.002s	PASSED
6	comment(tcid + " - Start") TC009 - Start	0.016s	PASSED
7	comment(tcid + " - Expected Result - user succesfully create room chat") TC009 - Expected Result - user succesfully create room chat	0.003s	PASSED

TC009_create Roomchat



TC010_send message to room chat

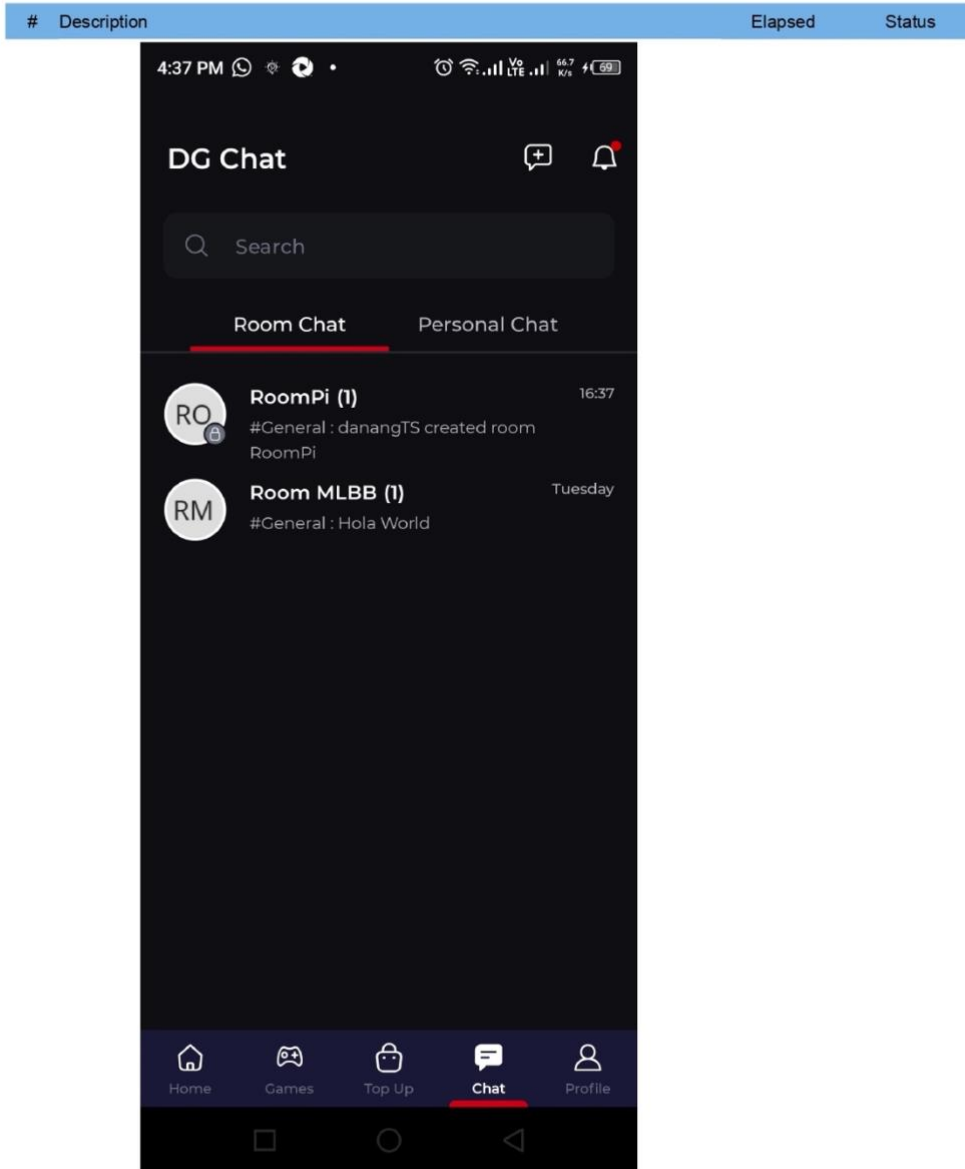
Information

ID	Test Cases/Dunia Games/TC010_send message to room chat		
Description			
Tag			
Start	2024-06-20 16:37:07	End	2024-06-20 16:37:34
Elapsed	27.630s		
Status	PASSED		

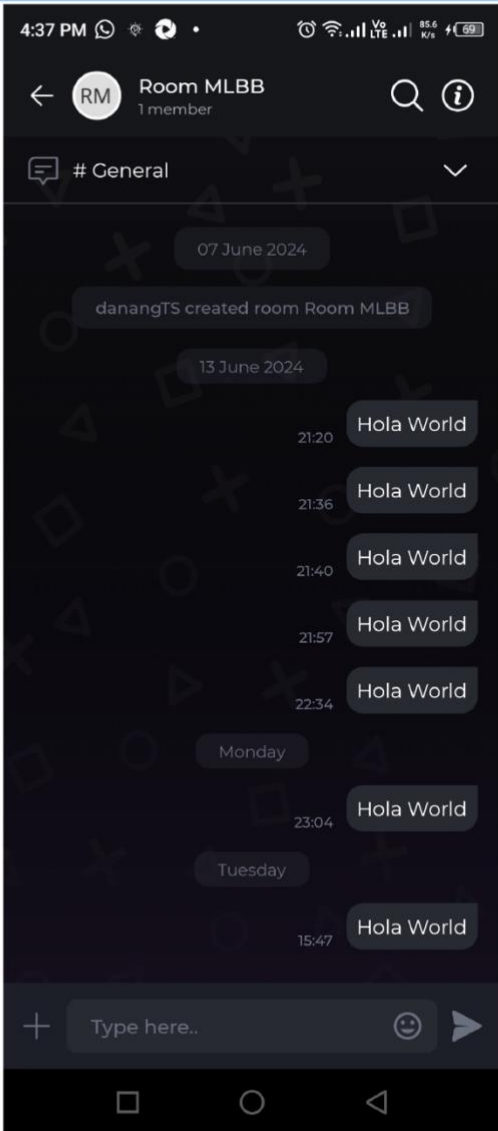
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.000s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	10.561s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.083s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.098s	PASSED
5	TC010("TC010", device_height, device_width)	0.001s	PASSED
6	comment(tcid + " - Start") TC010 - Start	0.001s	PASSED
7	comment(tcid + " - Expected Result - user succesfully send message to roomchat") TC010 - Expected Result - user succesfully send message to roomchat	0.002s	PASSED

TC010_send message to room chat

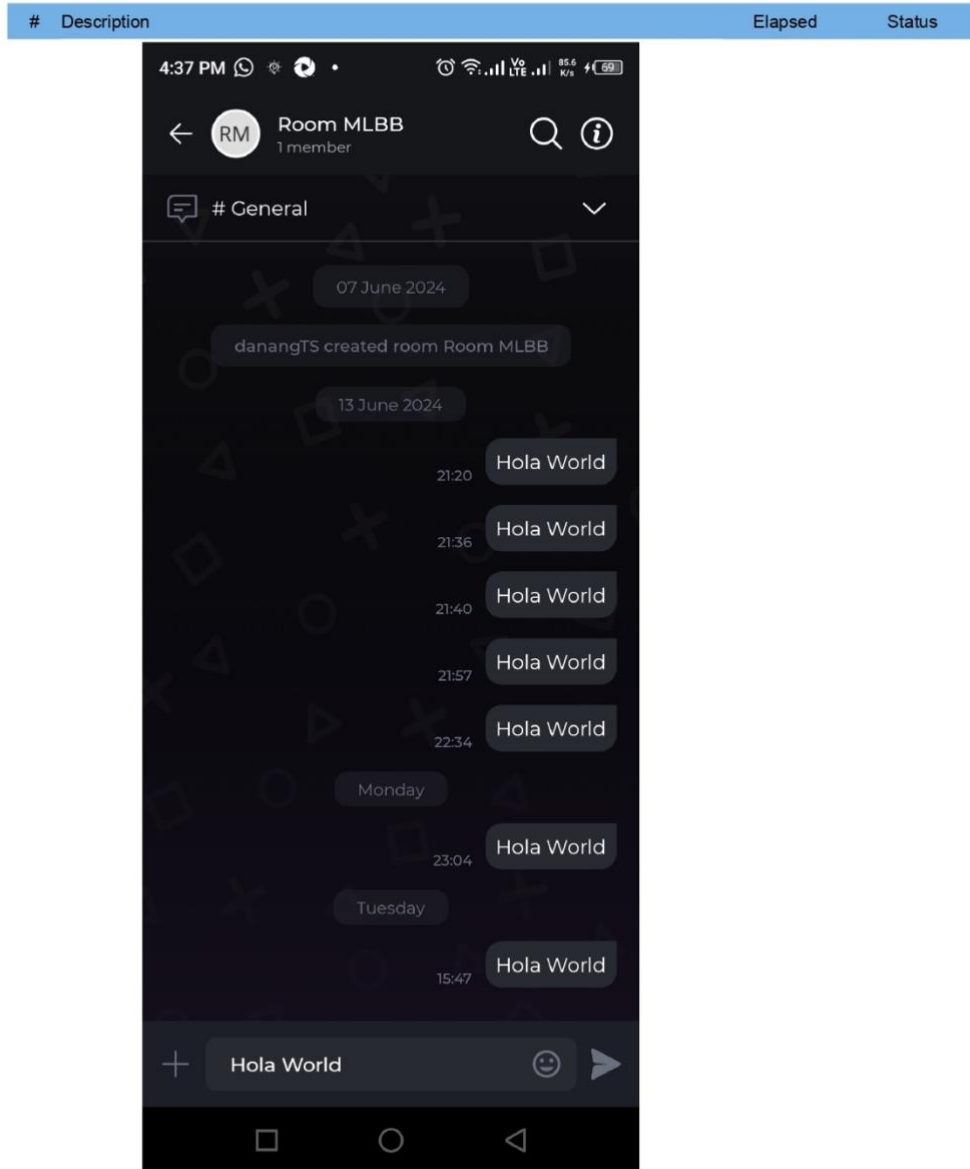


TC010_send message to room chat

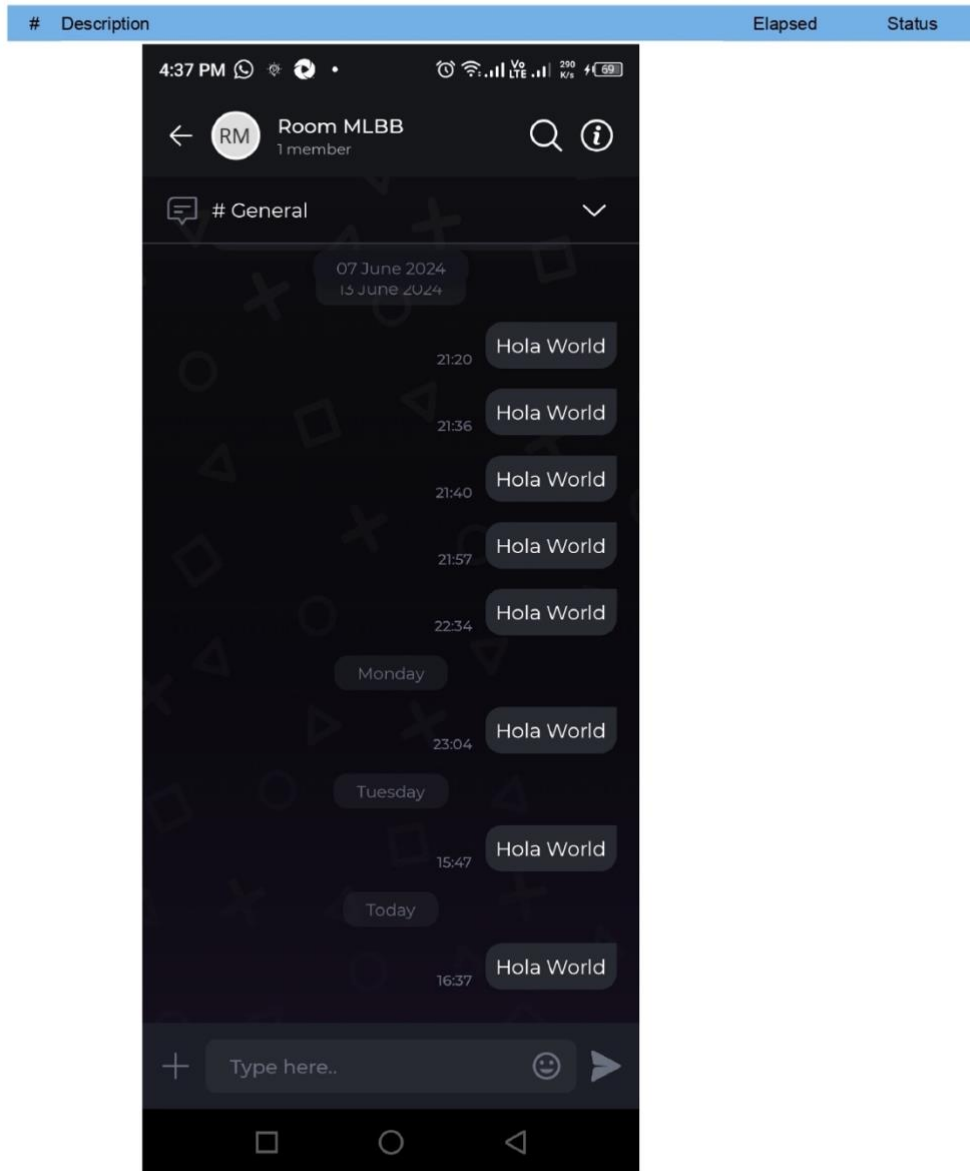
#	Description	Elapsed	Status
			

The screenshot shows a WhatsApp chat interface for a room named "Room MLBB" with 1 member. The chat history includes a system message from "danangTS" stating "created room Room MLBB" on June 7, 2024. On June 13, 2024, a series of "Hola World" messages are sent at the following times: 21:20, 21:36, 21:40, 21:57, 22:34, 23:04 (Monday), and 15:47 (Tuesday). The bottom of the screen shows the input field with the placeholder text "Type here..".

TC010_send message to room chat



TC010_send message to room chat



Topup

Execution Environment

Host name user - 192.168.1.6
Local OS Mac OS X 64bit
Katalon version 8.6.9.208
Browser null

Summary

ID Test Suites/Dunia Games/Topup
Description
Total 2
Passed 2
Error 0
Skipped 0
Failed 0
Incomplete 0
Start 2024-06-20 17:31:51 End 2024-06-20 17:34:45
Elapsed 2m - 53.532s

#	ID	Description	Status
1	Test Cases/Dunia Games/TC011_fail Topup Diamond		PASSED
2	Test Cases/Dunia Games/TC012_success_Topup Diamond		PASSED

TC011_fail Topup Diamond

Information

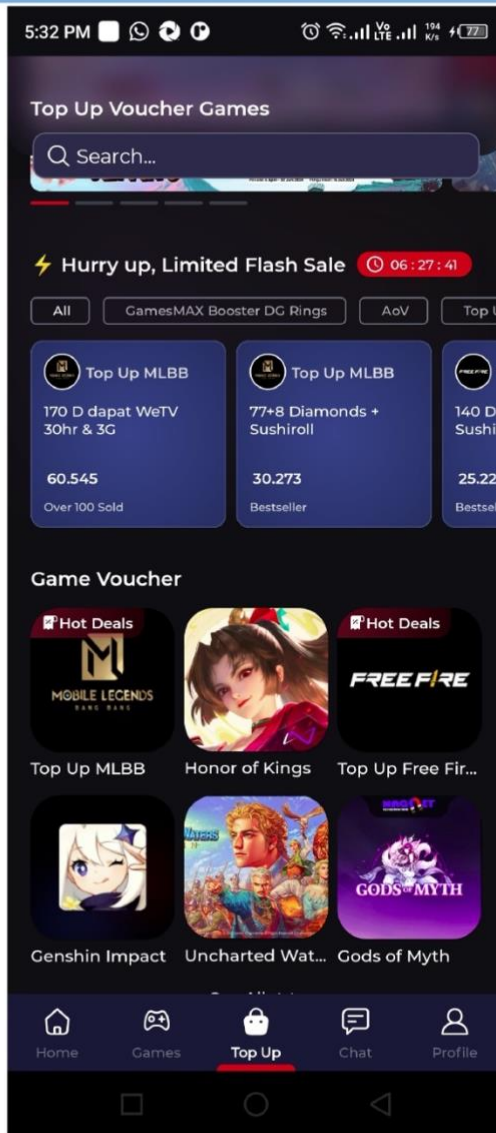
ID Test Cases/Dunia Games/TC011_fail Topup Diamond
Description
Tag
Start 2024-06-20 17:31:52 End 2024-06-20 17:33:19
Elapsed 1m - 27.199s
Status PASSED

Details

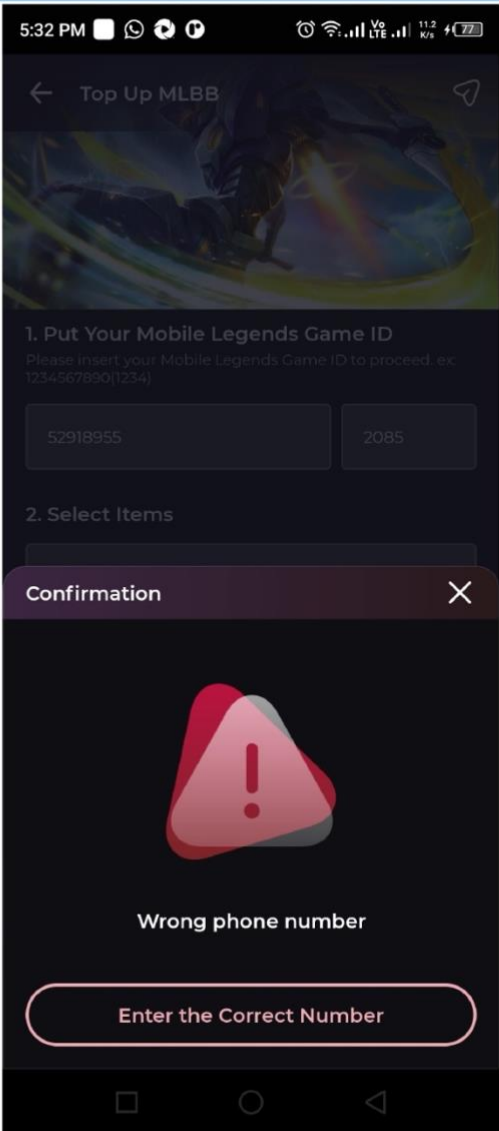
#	Description	Elapsed	Status
1	startLoginApplication()	0.002s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	19.506s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.140s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.116s	PASSED
5	TC011("TC011", device_height, device_width)	0.002s	PASSED
6	comment(tcid + " - Start") TC011 - Start	0.022s	PASSED
7	comment(tcid + " - Expected Result - can view several error message like: wrong id games, wrong phone number") TC011 - Expected Result - can view several error message like: wrong id games, wrong phone number	0.002s	PASSED

TC011_fail Topup Diamond

#	Description	Elapsed	Status
---	-------------	---------	--------

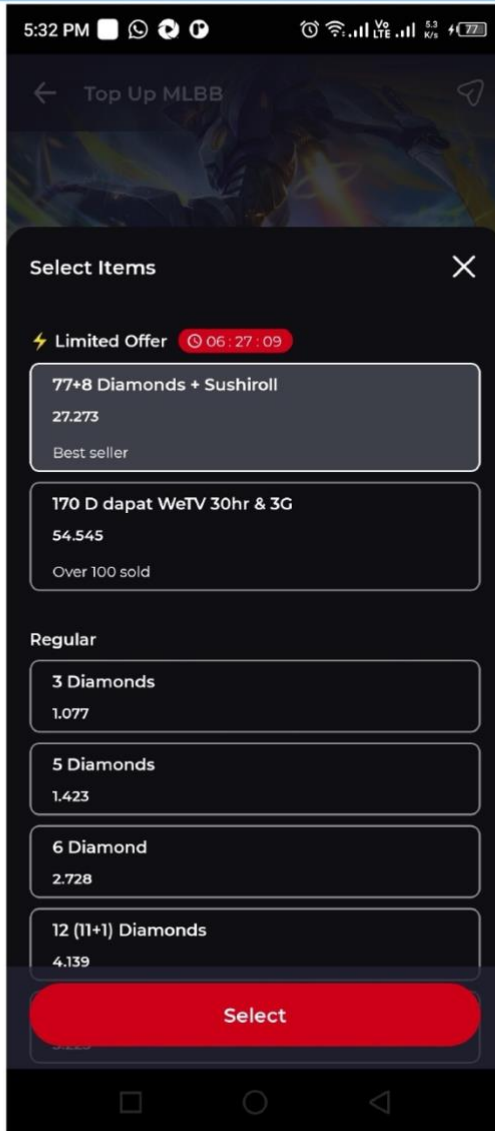


TC011_fail Topup Diamond

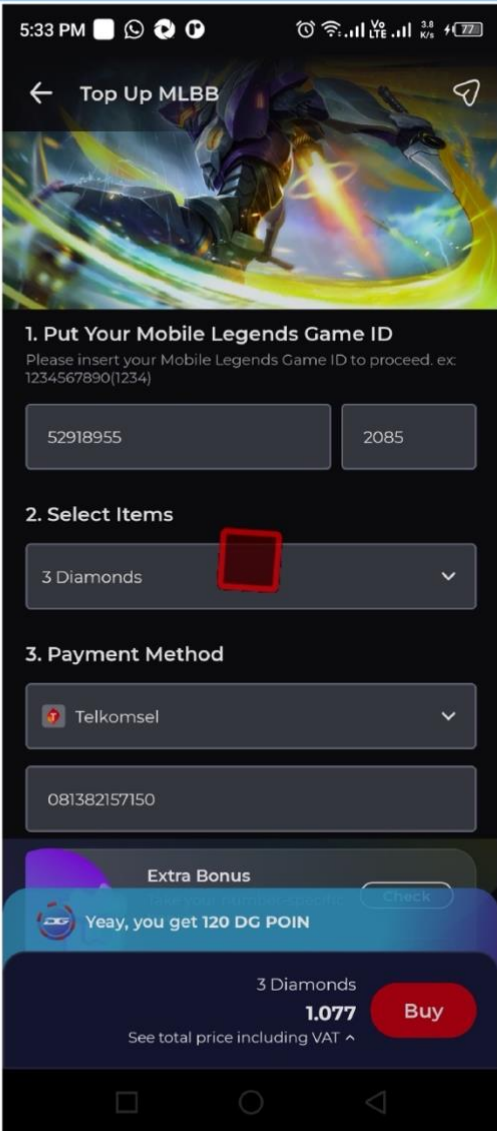
#	Description	Elapsed	Status
			

TC011_fail Topup Diamond

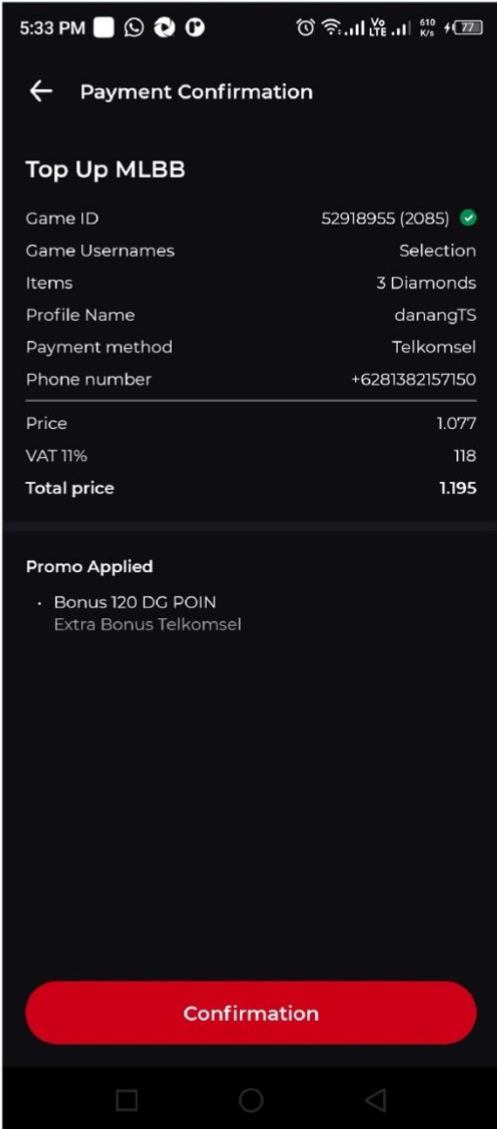
#	Description	Elapsed	Status
---	-------------	---------	--------



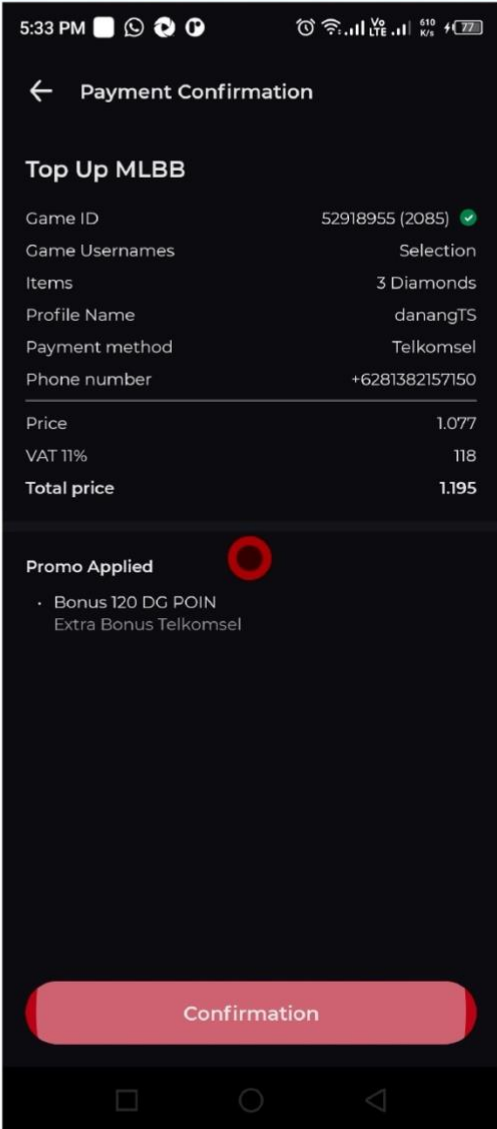
TC011_fail Topup Diamond

#	Description	Elapsed	Status
	 <p>5:33 PM</p> <p>← Top Up MLBB</p> <p>1. Put Your Mobile Legends Game ID Please insert your Mobile Legends Game ID to proceed. ex: 1234567890(1234)</p> <p>52918955 2085</p> <p>2. Select Items</p> <p>3 Diamonds</p> <p>3. Payment Method</p> <p>Telkomsel</p> <p>081382157150</p> <p>Extra Bonus Yeay, you get 120 DG POIN</p> <p>3 Diamonds 1.077 Buy See total price including VAT ^</p>		

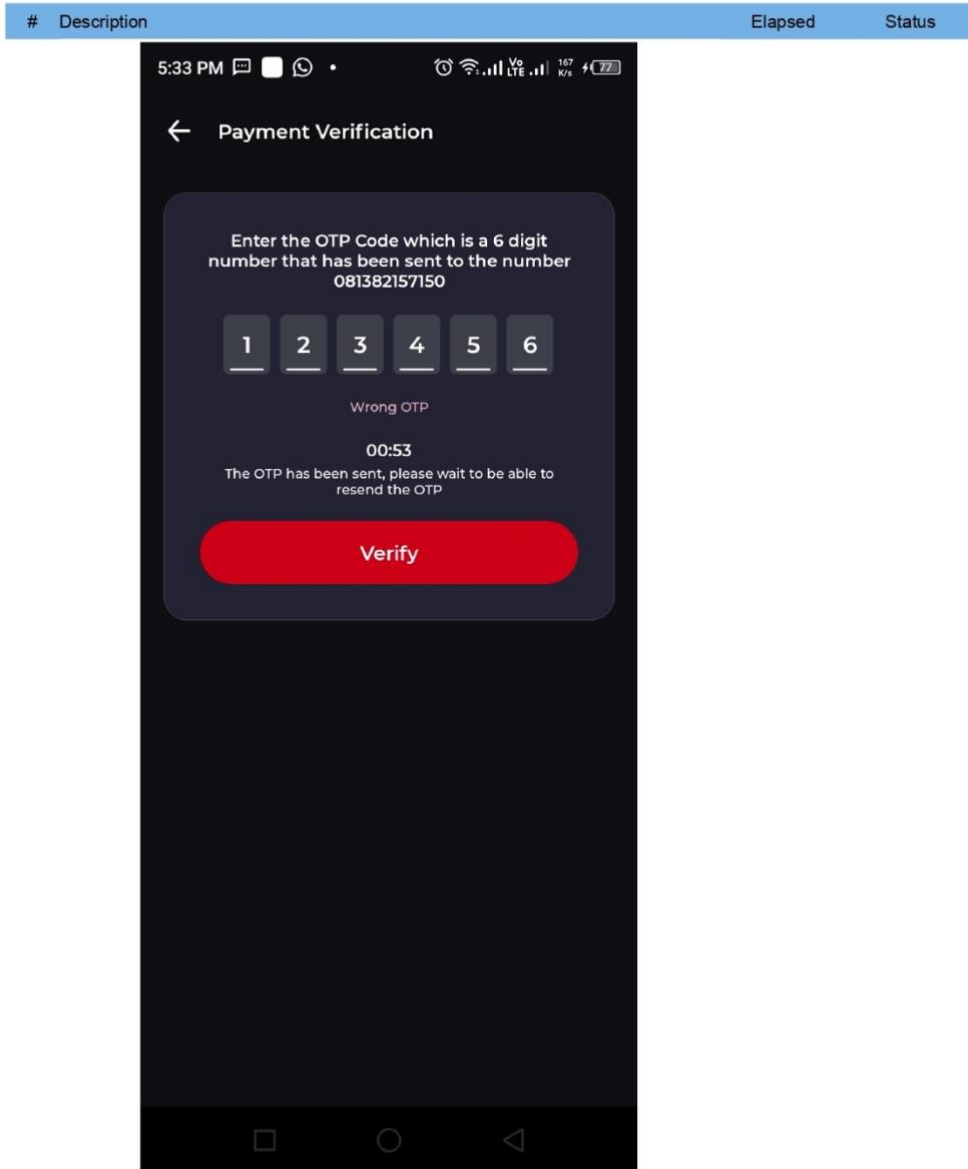
TC011_fail Topup Diamond

#	Description	Elapsed	Status																		
	 <p>The screenshot shows a mobile application interface for a payment confirmation. At the top, the time is 5:33 PM. The screen title is "Payment Confirmation" with a back arrow. Below that, the main heading is "Top Up MLBB". The details are as follows:</p> <table border="1"><tr><td>Game ID</td><td>52918955 (2085) ✓</td></tr><tr><td>Game Usernames</td><td>Selection</td></tr><tr><td>Items</td><td>3 Diamonds</td></tr><tr><td>Profile Name</td><td>danangTS</td></tr><tr><td>Payment method</td><td>Telkomsel</td></tr><tr><td>Phone number</td><td>+6281382157150</td></tr></table> <p>Below the details, a price breakdown is shown:</p> <table border="1"><tr><td>Price</td><td>1,077</td></tr><tr><td>VAT 11%</td><td>118</td></tr><tr><td>Total price</td><td>1,195</td></tr></table> <p>A "Promo Applied" section lists:</p> <ul style="list-style-type: none">- Bonus 120 DG POINExtra Bonus Telkomsel <p>At the bottom, there is a large red button labeled "Confirmation".</p>	Game ID	52918955 (2085) ✓	Game Usernames	Selection	Items	3 Diamonds	Profile Name	danangTS	Payment method	Telkomsel	Phone number	+6281382157150	Price	1,077	VAT 11%	118	Total price	1,195		
Game ID	52918955 (2085) ✓																				
Game Usernames	Selection																				
Items	3 Diamonds																				
Profile Name	danangTS																				
Payment method	Telkomsel																				
Phone number	+6281382157150																				
Price	1,077																				
VAT 11%	118																				
Total price	1,195																				

TC011_fail Topup Diamond

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for a payment confirmation. At the top, the status bar displays the time as 5:33 PM and various system icons. Below the status bar is a navigation bar with a back arrow and the text "Payment Confirmation". The main content area is titled "Top Up MLBB" and contains a list of transaction details: Game ID (52918955 (2085) with a green checkmark), Game Usernames (Selection), Items (3 Diamonds), Profile Name (danangTS), Payment method (Telkomsel), and Phone number (+6281382157150). A horizontal line separates this information from a pricing section: Price (1,077), VAT 11% (118), and Total price (1,195). Below the pricing is a "Promo Applied" section with a red circular icon, listing "Bonus 120 DG POIN" and "Extra Bonus Telkomsel". At the bottom of the screen is a large red button labeled "Confirmation". The Android navigation bar is visible at the very bottom.</p>		

TC011_fail Topup Diamond



TC012_success_Topup Diamond

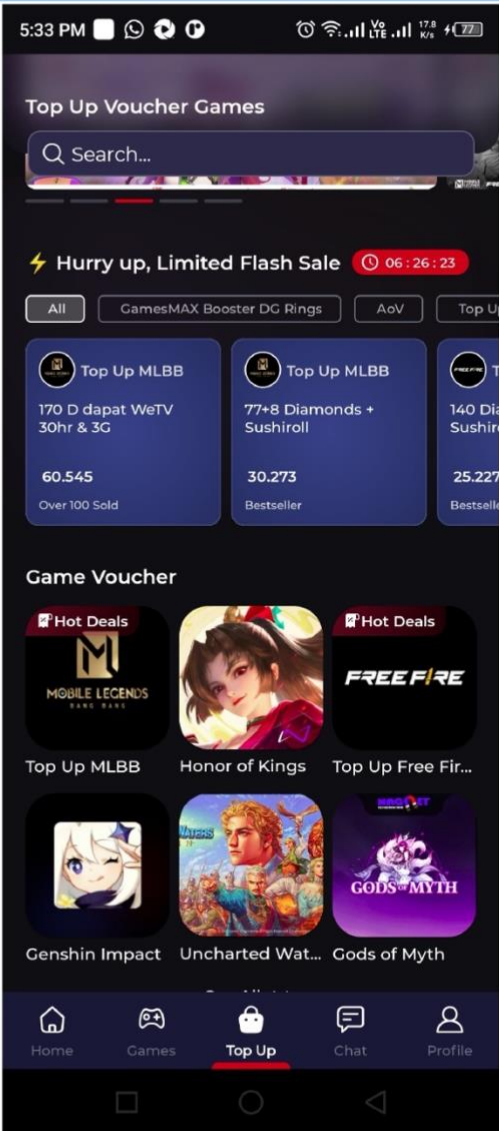
Information

ID	Test Cases/Dunia Games/TC012_success_Topup Diamond		
Description			
Tag			
Start	2024-06-20 17:33:19	End	2024-06-20 17:34:44
Elapsed	1m - 25.322s		
Status	PASSED		

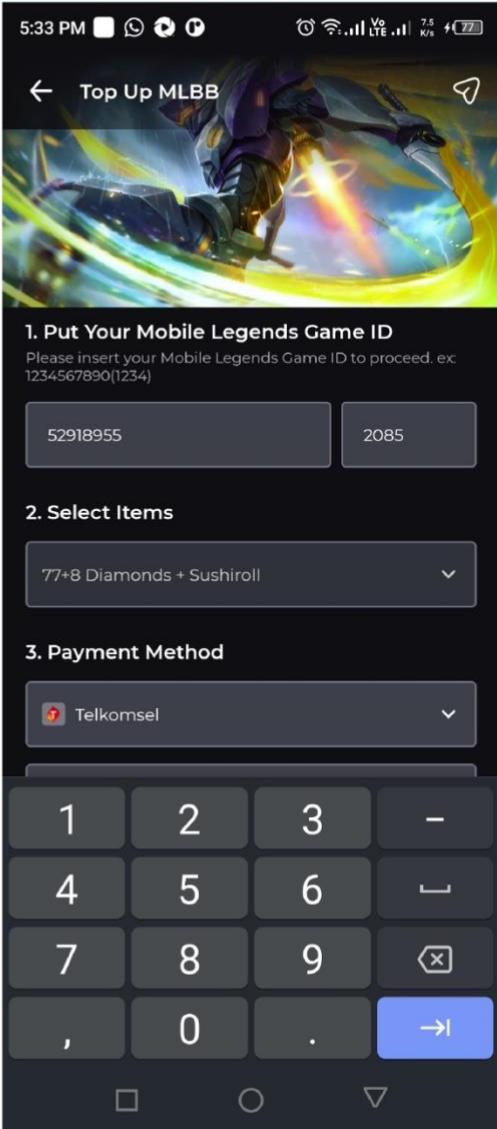
Details

#	Description	Elapsed	Status
1	startLoginApplication()	0.000s	PASSED
2	startExistingApplication(idAppDG, STOP_ON_FAILURE) Starting app with application ID: 'id.co.duniagames'	10.248s	PASSED
3	device_height = getDeviceHeight() Current device physical height is 2208	0.117s	PASSED
4	device_width = getDeviceWidth() Current device physical width is 1080	0.094s	PASSED
5	TC012("TC012", device_height, device_width)	0.001s	PASSED
6	comment(tcid + " - Start") TC012 - Start	0.002s	PASSED
7	comment(tcid + " - Expected Result - user direct to success payment page") TC012 - Expected Result - user direct to success payment page	0.001s	PASSED

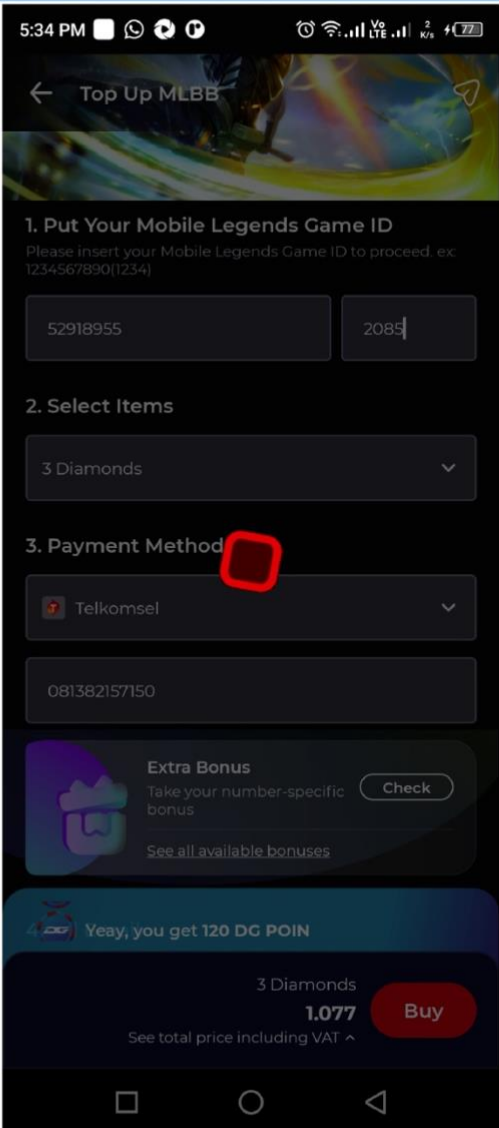
TC012_success_Topup Diamond

#	Description	Elapsed	Status
			

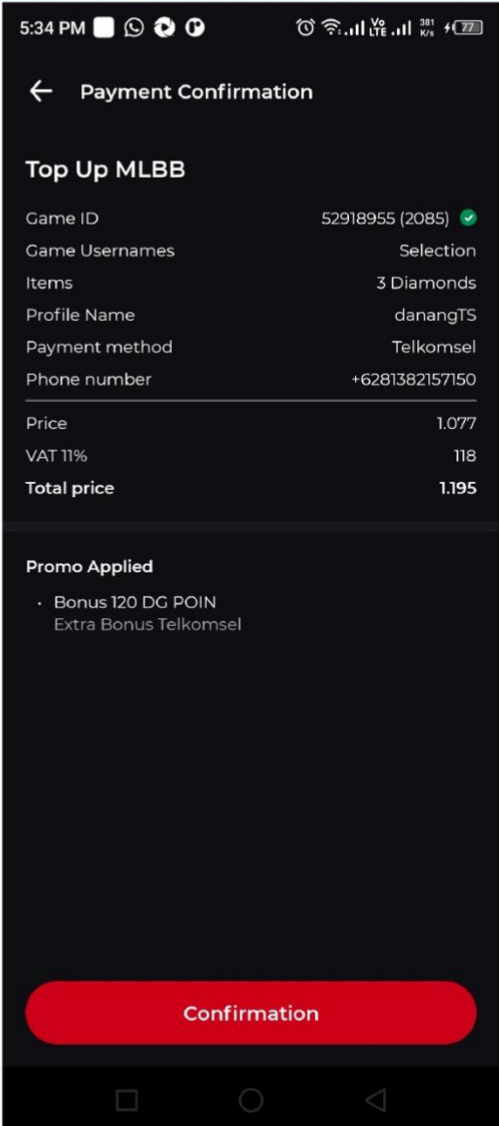
TC012_success_Topup Diamond

#	Description	Elapsed	Status
	 <p>5:33 PM</p> <p>← Top Up MLBB</p> <p>1. Put Your Mobile Legends Game ID Please insert your Mobile Legends Game ID to proceed. ex: 1234567890(1234)</p> <p>52918955 2085</p> <p>2. Select Items 77+8 Diamonds + Sushiroll</p> <p>3. Payment Method Telkomsel</p> <p>1 2 3 - 4 5 6 ↵ 7 8 9 ✕ , 0 . →</p>		

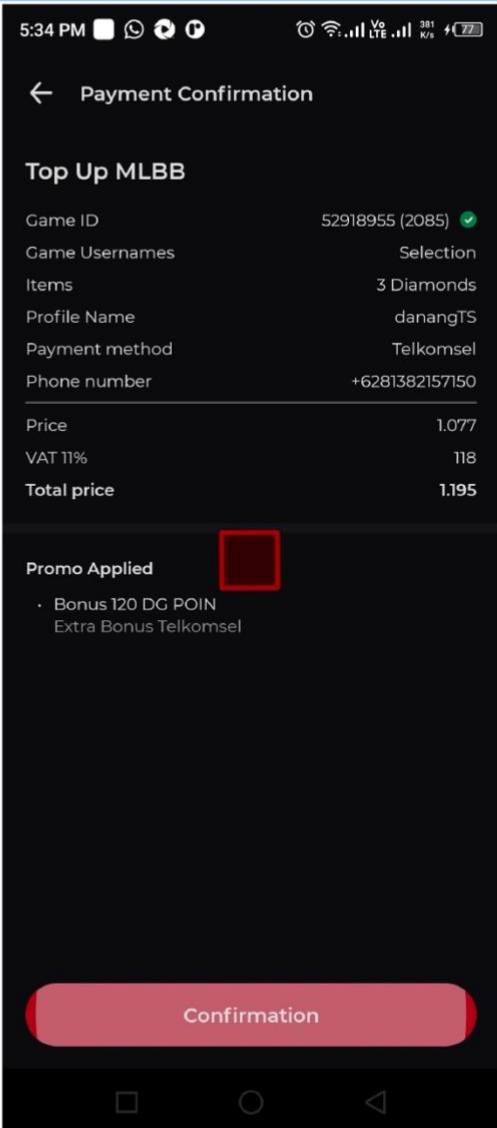
TC012_success_Topup Diamond

#	Description	Elapsed	Status
	 <p>The screenshot shows the 'Top Up MLBB' interface. It includes a header with the time 5:34 PM and various status icons. The main content is divided into three steps: 1. Put Your Mobile Legends Game ID, where the user has entered '52918955' and '2085'. 2. Select Items, where '3 Diamonds' is selected. 3. Payment Method, where 'Telkomsel' is selected and highlighted with a red square. Below the payment method, there is a field for the phone number '081382157150'. An 'Extra Bonus' section offers a 'Check' button. A promotional banner at the bottom states 'Yeay, you get 120 DG POIN'. The final purchase summary shows '3 Diamonds' for '1.077' with a 'Buy' button. The total price includes VAT.</p>		

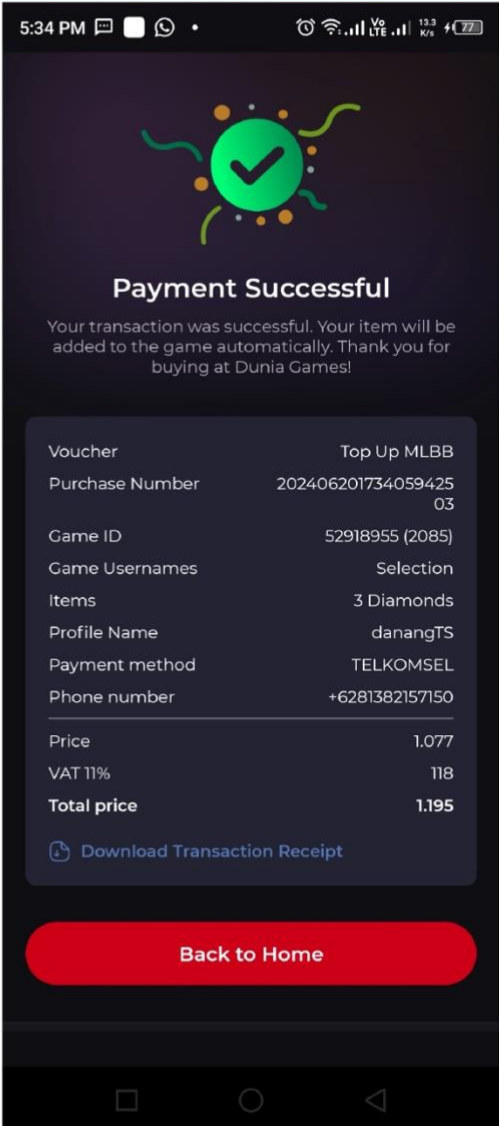
TC012_success_Topup Diamond

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for a 'Payment Confirmation' screen. At the top, there is a status bar with the time 5:34 PM, signal strength, Wi-Fi, and battery icons. Below the status bar is a navigation bar with a back arrow and the text 'Payment Confirmation'. The main content area is titled 'Top Up MLBB' and contains a list of transaction details: Game ID (52918955 (2085) with a green checkmark), Game Usernames (Selection), Items (3 Diamonds), Profile Name (danangTS), Payment method (Telkomsel), and Phone number (+6281382157150). A horizontal line separates this from a pricing section: Price (1,077), VAT 11% (118), and Total price (1,195). Below the pricing is a 'Promo Applied' section with a bullet point: 'Bonus 120 DG POIN Extra Bonus Telkomsel'. At the bottom of the screen is a large red button labeled 'Confirmation'. The Android navigation bar is visible at the very bottom.</p>		

TC012_success_Topup Diamond

#	Description	Elapsed	Status
	 <p>The screenshot shows a mobile application interface for a payment confirmation. At the top, the status bar displays the time as 5:34 PM and various system icons. Below the status bar is a navigation bar with a back arrow and the text 'Payment Confirmation'. The main content area is titled 'Top Up MLBB' and contains a list of transaction details: Game ID (52918955 (2085) with a green checkmark), Game Usernames (Selection), Items (3 Diamonds), Profile Name (danangTS), Payment method (Telkomsel), and Phone number (+6281382157150). A horizontal line separates this from a pricing section: Price (1,077), VAT 11% (118), and Total price (1,195). Below the pricing is a 'Promo Applied' section with a red square icon and a list of bonuses: 'Bonus 120 DG POIN' and 'Extra Bonus Telkomsel'. At the bottom of the screen is a large red button labeled 'Confirmation'. The Android navigation bar is visible at the very bottom.</p>		

TC012_success_Topup Diamond

#	Description	Elapsed	Status
			

No	Feature	Name	Description	Type	Precondition	Test Script(Step)	Expected Result
TC001	Registration	Fail to registration	as user fail to regist new account (email or password invalid, wrong otp)	Negative	non login had a internet connection	<ol style="list-style-type: none"> 1. Open Dunia Games application 2. click profile menu on bar menu 3. click icon mail 4. Click tab Register 5. at email form input incorrect format email 6. Input incorrect format password (not contains capital alphabet, number and character) 7. input incorrect OTP 	*can view several error like: - wrong format email -password not match -wrong otp *button register disable
TC002	Registration	Successfully registration new account	as user success to registration new account	Positive	non login had a internet connection	<ol style="list-style-type: none"> 1. Open Dunia Games application 2. click profile menu on bar menu 3. click icon mail 4. Click tab register 5. at email form input correct format email 6. click register button 7. Input correct format password (contains capital alphabet, number and character) 8. input match confirmation password 9. click button register 10. input correct OTP 11. click continue button 	*success to register and direct to homepage
TC003	Login	Fail to login	as user cant input incorrect credential	Negative	non login had a internet connection	<ol style="list-style-type: none"> 1. Open Dunia Games application 2. Click profile menu on menu bar 3. Click icon mail 4. input email 5. click button login 6. input incorrect password 7. click continue button 	*can see toast error wrong email or password •failed to login
TC004	login	Success to login	as user can input correct credential	Positive	non login had a internet connection	<ol style="list-style-type: none"> 1. Open Dunia Games application 2. Click profile menu on menu bar 3. Click icon mail 4. input correct email 5. click button login 6. input correct password 7. click continue button 	*successful login *direct to homepage
TC005	Article	Like and bookmark the article	user can like and bookmark the article	Positive	had internet connection	<ol style="list-style-type: none"> 1. Open Dunia Games application and do login 2. tap article submenu 3. tap bar one of the article from list 4. tap button like 5. tap button bookmark 	*button like will active *button bookmark will active
TC006	gamefeed	create post in gamefeed	user can create post	Positive	had internet connection	<ol style="list-style-type: none"> 1. Open Dunia Games application and do login 2. tap Menu game 3. tap one of game 	*user succesfully create the post

						4. tap post tab 5. tap create post 6. write post in post field 7. tap submit	
TC007	gamefeed	like one of the post	user can like the post	Positive	had internet connection	1. Open Dunia Games application and do login 2. tap Menu game 3. tap one of game 4. tap post tab 5. tap like button from one of the post 6. Write comment in comment section 7. Tap send button	*user succesfully like and comment the post
TC008	Tournament	create tournament	user can create Tournament	Positive	had internet connection	1. Open Dunia Games application and do login 2. tap tournamet submenu 3. tap create tournament button 4. input all field 5. input all field setting tournament 6. tap submit 7. view detail tournamet page	*Make sure there are several tab like : - Highlight - Participant - Watch * there button join tournament *there registration date
TC009	Room chat	create roomchat	user can create roomchat	Positive	had internet connection	1. Open Dunia Games application and do login 2. tap roomchat menu 3. tap create roomchat 4. input name room chat 5. choose games 6. set member gouta 7. set room as privat/general 8. tap submit button	user succesfully create room chat
TC010	Room chat	send message to one of room chat	user can send message to one of room chat	Positive	had internet connection	1. Open Dunia Games application and do login 2. tap roomchat menu 3. tap one of room chat 4. input message inf chat field 5. tap send	user succesfully send message to roomchat
TC011	Topup	Fail to topup diamond	user fail to topup diamond game	Negative	had internet connection	1. Open Dunia Games application and do login 2. tap Menu Topup 3. tap one of game voucher 4. input wrong id games 5. input wrong phone number	*button buy disable *can view several error message like: -wrong id games -wrong phone number
TC012	Topup	topup diamond one of games	user can topup diamond	Positive	had internet connection	1. Open Dunia Games application and do login 2. tap Menu Topup 3. tap one of game voucher	user direct to success payment page

						<i>4. input id games</i> <i>5. input phone number</i> <i>6. input Email (optional)</i> <i>7. tap buy button</i> <i>8. tap confirmation button</i>	
--	--	--	--	--	--	---	--