

Nama: Erlyna Audya Safitri
QA
06/12/2024

1. Studi kasus 1

Test Case ID	Test Description	Precondition	Steps	Expected Result
TC001	Validasi Harga Booking	Data pada tabel booking dan jadwal_harga tersedia di database.	1. Ambil data dari tabel booking. 2. Ambil data dari tabel jadwal_harga. 3. Cocokkan start_time, end_time, dan price pada kedua tabel. 4. Identifikasi jika terdapat perbedaan harga.	Harga pada tabel booking sesuai dengan price pada tabel jadwal_harga.
TC002	Deteksi Double Booking	Data pada tabel booking tersedia.	1. Ambil data dari tabel booking. 2. Kelompokkan data berdasarkan waktu (start_time dan end_time). 3. Periksa apakah ada waktu yang sama digunakan oleh lebih dari satu booking.	Tidak ada double booking untuk waktu yang sama (start_time dan end_time).
TC003	Validasi Data Jadwal yang Tidak Ada	Data pada tabel booking dan jadwal_harga tersedia.	1. Ambil data dari tabel booking. 2. Periksa apakah start_time dan end_time dari setiap booking memiliki entri yang sesuai di tabel jadwal_harga.	Semua booking memiliki jadwal yang valid pada tabel jadwal_harga.
TC004	Validasi Format Data pada Tabel Booking	Data tabel booking tersedia.	1. Periksa apakah price hanya mengandung nilai numerik. 2. Validasi apakah start_time dan end_time dalam format waktu yang benar (HH:MM:SS). 3. Pastikan booking_id unik untuk setiap entri.	Semua data pada tabel booking memiliki format yang benar, dan tidak ada duplikasi pada kolom booking_id.

TC005	Validasi Perhitungan Harga Akumulasi Booking	Data pada tabel booking tersedia.	1. Ambil semua booking pada tanggal yang sama (2022-12-10). 2. Hitung total harga berdasarkan booking per waktu tertentu. 3. Bandingkan total harga booking dengan perhitungan manual untuk memvalidasi kebenarannya.	Total harga akumulasi pada booking sesuai dengan data yang dihitung manual berdasarkan waktu masing-masing.
-------	--	-----------------------------------	---	---

Automation:

```

import sqlite3

# Koneksi ke database
conn = sqlite3.connect('database.db') # Sesuaikan dengan database Anda
cursor = conn.cursor()

# Query data booking
booking_query = """
SELECT * FROM booking
WHERE date = '2022-12-10' AND venue_id = 15;
"""
cursor.execute(booking_query)
bookings = cursor.fetchall()

# Query data jadwal dan harga
jadwal_query = """
SELECT * FROM jadwal_harga
WHERE date = '2022-12-10' AND venue_id = 15;
"""
cursor.execute(jadwal_query)
jadwals = cursor.fetchall()

# Validasi data
errors = []
time_slots = set()

for booking in bookings:
    start_time, end_time, price = booking[4], booking[5], booking[6]
    matched = False

    # Cek kesesuaian harga
    for jadwal in jadwals:
        if start_time == jadwal[3] and end_time == jadwal[4]:
            matched = True
            if price != jadwal[5]:
                errors.append(f"Error harga booking ID {booking[1]}: Harusnya {jadwal[5]}, ditemukan {price}")
            break

    # Jika jadwal tidak ditemukan
    if not matched:

```

```

errors.append(f"Tidak ada jadwal untuk booking ID {booking[1]}")

# Cek double booking
if (start_time, end_time) in time_slots:
    errors.append(f"Double booking terdeteksi: {start_time} - {end_time}")
else:
    time_slots.add((start_time, end_time))

# Output hasil validasi
if errors:
    for error in errors:
        print("ERROR:", error)
    else:
        print("Semua booking valid.")

# Tutup koneksi database
conn.close()

```

2. Studi kasus 2

Test Case: Website AYO				
Test Case ID	Test Title	Precondition	Steps	Expected Result
TC001	Validasi Form Pencarian Venue	Website AYO dapat diakses	1. Masukkan nama venue yang valid pada form pencarian. 2. Klik tombol "Cari".	Hasil pencarian menampilkan venue yang sesuai dengan input.
TC002	Validasi Pagination Hasil	Website AYO dapat diakses dan memiliki banyak data pencarian.	1. Masukkan nama venue yang valid. 2. Klik tombol "Cari". 3. Klik tombol navigasi halaman berikutnya.	Halaman navigasi berikutnya menampilkan data pencarian yang valid dan sesuai dengan hasil sebelumnya.
TC003	Validasi Responsif Website	Website AYO dapat diakses di berbagai perangkat.	1. Akses website dari perangkat desktop. 2. Akses website dari perangkat mobile. 3. Bandingkan tampilan.	Tampilan website menyesuaikan ukuran layar perangkat tanpa layout yang rusak.
Test Case: Aplikasi Mobile AYO				

Test Case ID	Test Title	Precondition	Steps	Expected Result
TC004	Validasi Login dengan Data Valid	Aplikasi AYO sudah terinstal, akun pengguna tersedia.	1. Masukkan email dan password yang valid. 2. Klik tombol "Login".	Pengguna berhasil masuk ke dashboard utama.
TC005	Validasi Login dengan Data Invalid	Aplikasi AYO sudah terinstal.	1. Masukkan email yang tidak terdaftar. 2. Klik tombol "Login".	Pesan error muncul, seperti "Email tidak ditemukan".
TC006	Validasi Alur Registrasi	Aplikasi AYO sudah terinstal.	1. Klik "Daftar". 2. Masukkan email dan password baru. 3. Klik "Simpan".	Pengguna berhasil registrasi dan diarahkan ke halaman login dengan pesan "Registrasi berhasil".
TC007	Validasi Pembatalan Booking	Akun pengguna memiliki booking aktif.	1. Masuk ke aplikasi. 2. Pilih menu "Booking Saya". 3. Klik "Batalkan" pada salah satu booking.	Booking berhasil dibatalkan, dan status berubah menjadi "Dibatalkan".

Dokumentasi Penggunaan Automation Test Script

Yang diperlukan:

- Python sudah terinstal.
- Database sudah terkonfigurasi dengan tabel booking dan jadwal_harga.

Cara Penggunaan:

- Instalasi Library
- Jika menggunakan SQLite, tidak perlu instalasi tambahan.
- Jika menggunakan database lain, pastikan menginstal konektor yang sesuai dengan database tersebut.
- Menjalankan Script
- Simpan script Python pada folder yang diinginkan.
- Buka terminal atau command prompt dan navigasikan ke folder tempat script disimpan.
- Jalankan script dengan perintah:

bash

Copy code

python nama_script.py

Output

Jika ada kesalahan (misalnya harga tidak sesuai atau terjadi double booking), akan muncul pesan kesalahan.

Jika tidak ada masalah, akan muncul pesan yang menyatakan bahwa semua booking valid.

Penyesuaian

Pastikan koneksi database sudah benar dan sesuai dengan jenis database yang digunakan.

