# DA\_JMeter\_L1\_Use\_Case\_01

## JMeter Test Case: Appointment Booking Performance and Functionality Test

Objective: To validate the performance, functionality, and reliability of the Appointment Booking module in the DanpheEMR system under various load conditions.

# Table of Contents

# Introduction

# Conduct Guidelines

# Use Case Template

# Use Case Specification: UC- Appointment Booking01

# Introduction

This document details the specifications for use cases and guidelines for proper conduct during assessments. It is aimed at ensuring a thorough understanding and compliance with the best practices in Performance testing, focusing on JMeter tool generate the Performance Metrics.

# Conduct Guidelines

## Important Notice to All Assessors:

Prior to commencing the assessment, please ensure that you have read and understood the following conduct guidelines. Failure to adhere to these guidelines may result in disqualification from the assessment process.

## Resource Restricted Usage:

Access to resources like GMAIL, Google ChatGPT, Quora, Stack Overflow, official emails, search engines, and AI tools will be restricted or monitored.

## Monitoring and Compliance:

Your activities will be monitored. Accessing blocked sites or using unauthorized tools will be recorded and may lead to disqualification.

## JMeter Standards:

Use descriptive names for elements, documenting test plans and components, and structuring tests logically for clarity and maintainability. Emphasize parameterization, efficient listener use, and assertions for validation, while managing error handling and logging meticulously. Ensure resource-conscious configurations for optimal performance testing

# Use Case Details

# **Use Case ID: UC-HealthEMR01**

# Title: Appointment Booking Actors: User, SASA Health EMR System

# Description: This JMeter test case aims to validate the Appointment Booking feature of the SASA Health EMR System. The test will emulate user actions for booking a new patient appointment, ensuring the system handles the process accurately under various load conditions. The test will cover patient registration, appointment scheduling, and validation of the booking process.

# Test Steps to follow Main Success Scenario (Basic Flow):

1. Login Workflow

Step 1: The hospital staff member navigates to the HMS login page.

URL: https://healthapp.yaksha.com/

Step 2: They enter their credentials and are granted access to the system.

Username: admin

Password: pass123

Note : Add all the cookies necessary for maintaining the session state and authentication. Ensure you have proper correlation mechanisms in place to handle dynamic tokens like "\_\_RequestVerificationToken" for subsequent requests.

2. Appointment Booking Workflow:

## Step 1: Upon successful login, the staff member selects the 'Appointment' section and chooses 'Book Appointment' to initiate a new appointment entry.

## URL: https://healthapp.yaksha.com/Home/Index#/Appointment/CreateAppointment

## Step 2: They input the patient's personal and contact details:

## First Name: Brock

## Last Name: Roy

## Gender: Male

## Age: 34

## Contact Number: 1234567891

## Reason for visit: [Enter the reason]

## Step 3: The staff member selects the desired appointment date and time, assigns the patient to the appropriate department and doctor.

## Appointment Date: 27-05-2024

## Appointment Time: 15:55:57

## Department: [Select department]

## Doctor: [Select doctor]

## Step 4: After reviewing the details, they submit the booking request by clicking on 'Add Appointment.'

## Step 5: The system processes the request and confirms the appointment, which is then visible in the 'Appointment Booking List'

## Pre-Test Setup

Verify JMeter Installation and Environment Setup (Test Case 1): Ensure that JMeter is installed correctly and is operational.

Verify that all necessary plugins and dependencies for the test are in place.

**Preconditions:**

The patient registration system is fully operational.  
Hospital staff have secure access and are authorized to operate the registration module.

Use the URL – https://healthapp.yaksha.com/

Username: admin / Password: pass123

## Test Preparation

## Single User Response Testing

Set up a Test Plan to execute a single thread representing one user.

Collect key performance parameters such as response time, latency, connect time, and size of the response data.

Record and report the exact sequence of requests and responses for detailed analysis.

## Stress Test - Peak Load Time

Design a stress test to identify the system's peak load capacity.

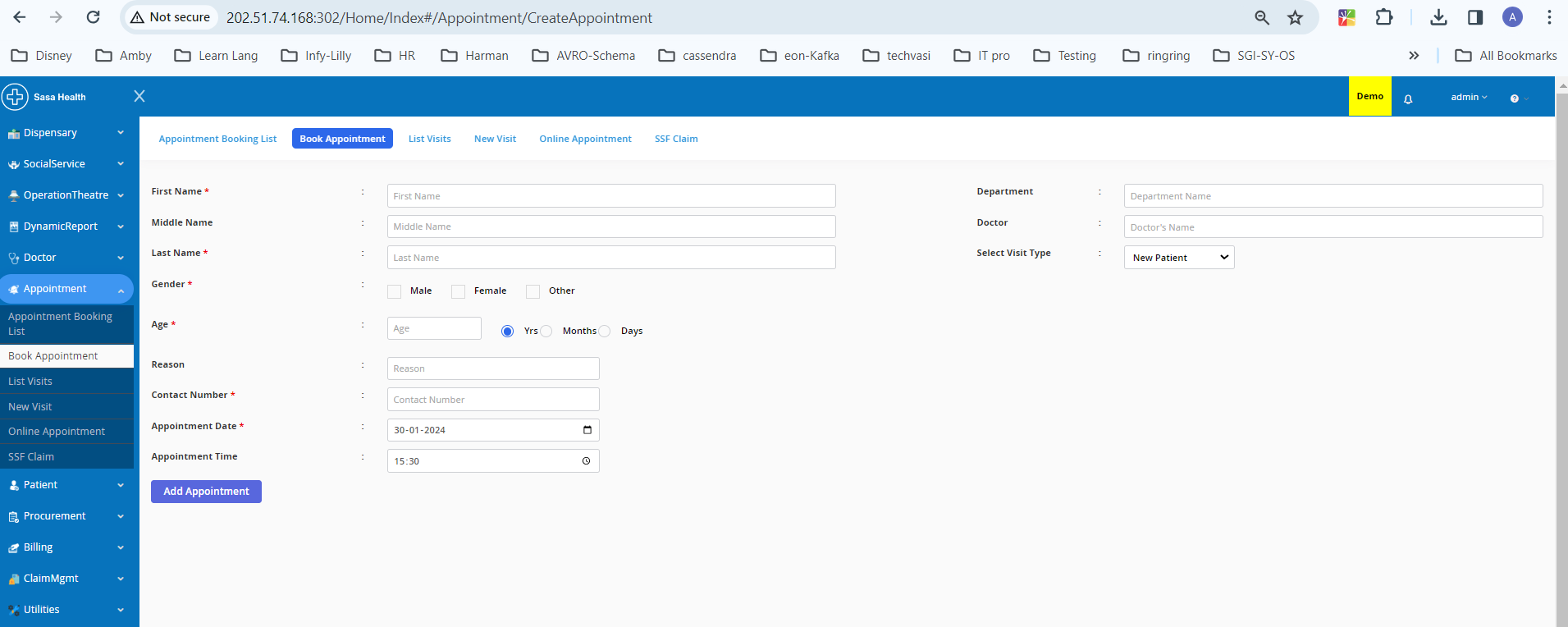
Incrementally increase the load until you reach or exceed the system's breaking point to understand its behavior under extreme stress.

Monitor system metrics like CPU, memory, disk I/O, and network I/O in addition to JMeter's performance metrics.

Document the response times, throughput, error rates, and server resource utilization at peak load time.

## Test Execution

1. Load Creation and Analysis
2. Simulate virtual users for appointment booking. Configure varying numbers of threads, ramp-up periods, and loop counts to reflect different user load scenarios.
3. Add Reports for View Results Tree, Summary Report, Aggregate Report , Graph Report , Aggregate Graph and Response Times Over Time to monitor and analyze performance metrics.
4. Direct the load towards the Appointment Booking endpoint/function within the DanpheEMR system.
5. Implement/Simulate real-time and varied load patterns for complex load scenarios. A) Ramup in steps for 150 concurrent connections B) Ramup with gradual load(1-150-500) in 30 sec and ramp down with in 3 sec
6. Implement HTTP to manage and simulate 150 real user sessions during appointment booking. Test the application's session management capabilities, including handling new user sessions, session timeouts, and persistent sessions.
7. Use Response Assertion to validate texts in the response, such as confirmation of appointment booking. Implement assertions for JSON or XML , Duration , Size to validate Booking ID , Doctor name , Booking slot from the response.
8. Group multiple related requests (like selecting a doctor, choosing a time slot, confirming an appointment) as a single transaction. Implement JMeter Controller for organizing requests and to manage the order of requests.



1. Stress Testing and Dynamic Behavior in Appointment Booking:

Execute stress tests to assess system performance under peak load conditions.

Start with a baseline of 150 virtual users and scale up to 500 users to simulate the extreme load during peak hours.

Ensure that during peak load times, the response times do not exceed 3 seconds for a satisfactory user experience.

Use Conditional , loop and Runtime Controllers to for rescheduling the appointment to next day for Patient listed on <http://localhost:5000/Home/Index#/Appointment/ListAppointment>

Search Three Patients when the patient is available then Book Appointment.

Apply a runtime check point for Patient Search at runtime for 15 sec.

1. Targeted Throughput and User Behavior Timing: Apply content timers for two thread and Throughput timers to achieve desired (1000 hits/min) throughput levels and introduce random delays between appointment booking actions for existing patient within the SASA Health EMR system.
2. In the SASA Health EMR System, the test will mimic actual patient engagements by dynamically capturing key appointment details during interactions and reusing them in subsequent steps, ensuring the system’s ability to manage a realistic patient booking flow from doctor selection to final confirmation. Utilize Regular Expression Extractor, extractors to set up this task.
3. Employ data-driven methods to input diverse patient information into the EMR system, utilizing a predefined JSON structure for appointment bookings to reflect the varied scenarios a real user might encounter.



1. Manually configure HTTP Request for the appointment booking endpoint with appropriate parameters, headers, and body data. Validate the correctness of the requests and responses from the server.
2. Add Listener as Assertion Result , Graph Result , Aggregate Report Summary Report View Result Tree Aggregate Report Aggregate Graph View result in table where these reports are required.

## Post-Test Analysis

Analyze the collected data for response time, throughput, error rate, and other performance metrics.

Compare results from different test scenarios to understand the impact of load variations and different configurations.

Document findings and make recommendations for system optimization or adjustments.

Save all the logs and Screen Shots of complex matrix for review

## Assessment Criteria:

Demonstrate the ability to set up and configure all the JMeter components.

Create a test plan that reflects a thorough understanding of both basic and advanced JMeter functionalities.

The test plan must be robust enough to simulate a realistic load scenario on the SASA Health EMR System.

Identify potential performance bottlenecks and suggest optimizations.

## Expected Outcomes:

The SASA Health EMR System should handle 150 concurrent user sessions without significant performance degradation.

All aspects of the system's functionality should be verified under load, including user authentication, appointment booking, and administrative functions.

The test should reveal how the system behaves under peak loads and provide insights into the system's scalability.

## Delivery:

The use case should be presented as a fully documented JMeter test plan, along with a detailed report of the findings and recommendations for performance improvements.