Performance Testing Approach

For

MOSIP Pre-Registration Module

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision History** | | | |
| **Revision Number** | **Date** | **Author(s)** | **Description** |
| V1.0 | 13-Feb-2019 | Prashanth Thimmavajjala | Initial Draft |
|  |  |  |  |
|  |  |  |  |

TABLE OF CONTENTS

[1.0 EXECUTIVE SUMMARY 4](#_Toc1134304)

[2.0 INTRODUCTION 5](#_Toc1134305)

[2.1 PURPOSE 5](#_Toc1134306)

[2.2 OBJECTIVES 6](#_Toc1134307)

[2.3 REFERENCE DOCUMENTS 6](#_Toc1134308)

[3.0 SCOPE 6](#_Toc1134309)

[3.1 IN SCOPE 6](#_Toc1134310)

[3.2 OUT OF SCOPE 7](#_Toc1134311)

[4.0 PERFORMANCE ACCEPTANCE CRITERIA 7](#_Toc1134312)

[4.1 PERFORMANCE TESTING GOALS 7](#_Toc1134313)

[4.2 ENGAGEMENT COMPLETION CRITERIA 7](#_Toc1134314)

[5.0 WORKLOAD DISTRIBUTION 8](#_Toc1134315)

[6.0 TEST SCRIPTING 9](#_Toc1134316)

[7.0 DATA GENERATION NEEDS 10](#_Toc1134317)

[8.0 PROPOSED TEST EXECUTION APPROACH 10](#_Toc1134318)

[8.0.1 BASELINE TESTS 10](#_Toc1134319)

[8.0.2 TESTS WITH INCREASING USER LOAD 10](#_Toc1134320)

[8.0.3 Resource Monitoring 11](#_Toc1134321)

[9.0 DELIEVERABLES 11](#_Toc1134322)

[10.0 TEST ENVIRONMENT 12](#_Toc1134323)

[11.0 RESULT ANALYSIS & REPORTING 12](#_Toc1134324)

[12.0 ENTRY, EXIT AND TEST STOP CRITERIA 13](#_Toc1134325)

[13.0 DEPENDENCIES 13](#_Toc1134326)

[14.0 ROLE AND RESPONSIBILITIES 14](#_Toc1134327)

[15.0 RISK ASSESSMENT 15](#_Toc1134328)

[16.0 LIMITATIONS & ASSUMPTIONS 17](#_Toc1134329)

[17.0 APPENDIX 17](#_Toc1134330)

# EXECUTIVE SUMMARY

Performance testing needs to be conducted on the MOSIP Pre-Registration module of the overall platform to identify performance issues and analyze them. This Test Plan is a technical document that provides guidelines on how performance test planning activities should be carried out within the project. This is a living document that may be refined as the project progresses.  Manager, Test Lead, Product Manager, Project Manager, and Development Manager etc. shall review and approve the final version of the Test Plan document.

It is planned that Apache JMeter will be used for the scripting and load testing effort. Monitoring the server performance during the tests will be done using native monitoring tools on the servers or using Azure based monitoring dashboards. As per the project need, any additional tools required for result analysis and monitoring may be identified and recommended.

The high-level risks for this effort would be related to environment availability for test execution, access to the servers and logs for monitoring, license and hardware availability.

# INTRODUCTION

This document describes the Performance Test plan that covers the goals, scope, and high level approach and detailed performance testing scenarios for MOSIP Pre-Registration module listed below:

|  |  |
| --- | --- |
| **No.** | **Scenario Name** |
| 1 | Appointment Booking Full Flow |
| 2 | Enter/Edit Demographic Detail |
| 3 | Appointment booking for an existing application |
| 4 | Delete an application |
| 5 | Book Appointment on existing application |
| 6 | Cancel Appointment |
| 7 | Download acknowledgement |
| 8 | Rebook appointment |

## PURPOSE

The document mentions various phases that will be followed during the performance testing exercise for MOSIP Pre-Registration Module. These phases are mentioned below:

* Introduction
  + Application functionality
  + Objectives and scope of performance testing
  + Scenarios for performance testing
  + Acceptance criteria/goals
* Performance Test Execution Approach
  + Tools to be used
  + Test environment and data generation needs
  + Test execution Approach
  + Reporting and deliverables
  + Timelines for testing
* Administrative activities
  + Customer dependencies
  + Risks
  + Assumptions
  + Roles and Responsibilities

## OBJECTIVES

This performance testing exercise is split across two phases based on the number of modules covered and goals of the testing.

* Primary objective is to execute the performance tests for the identified scenarios in isolation for the expected load levels, perform basic analysis of the performance test reports, server monitoring logs and find out bottleneck if there is any. Tests will be repeated to achieve various transactions per second levels.
* After each scenario level performance issues are fixed, all the scenarios will be run together based on the workload model to simulate a production-like workload on the target system

## REFERENCE DOCUMENTS

Documents mentioned below have been used as reference while creating this strategy document:

* High level Non-Functional requirements specified
* Workload model document
* Outcomes of sprint planning discussions

# SCOPE

Scope of the engagement is divided into two phases depending on the modules covered in each phase.

## IN SCOPE

Following scenarios will be covered in the performance testing for the pre-registration module:

|  |  |
| --- | --- |
| **No.** | **Scenario Name** |
| 1 | Appointment Booking Full Flow |
| 2 | Enter/Edit Demographic Detail |
| 3 | Appointment booking for an existing application |
| 4 | Delete an application |
| 5 | Book Appointment on existing application |
| 6 | Cancel Appointment |
| 7 | Download acknowledgement |
| 8 | Rebook appointment |

Testing types covered in this phase are

* Load testing for individual scenarios
* Load testing of mix of scenarios for production-like workload

## OUT OF SCOPE

Following areas are not present in the current scope of the testing

* Functional testing
* Security Testing
* If Performance Testing uncovers any bottlenecks, testing team will provide results and recommendations. However testing team would not be directly performing any tuning activities

# PERFORMANCE ACCEPTANCE CRITERIA

Performance efforts have two sets of criteria associated with them. The first are Performance criteria (requirements and goals) and second is engagement completion criteria. This effort will be considered as complete when either of the criteria is completed.

## PERFORMANCE TESTING GOALS

Performance testing goals or objectives are defined as follows:

* Load testing of identified scenarios and analysis
* Load testing of identified scenarios with mix workload

## ENGAGEMENT COMPLETION CRITERIA

Performance testing effort is assumed to be complete when :

* Testing is completed for all the scenarios listed under scope section.
* Transactions per second requirements of 10 TPS, 20 TPS, 30 TPS, 50 TPS, 70 TPS and 100 TPS are achieved
* All API response times are within the SLA of 2 second
* The target requirement of 2222 transactions per hour is met

# WORKLOAD DISTRIBUTION

Workload model provides classification in percentages which will be used to identify total number of users across scenarios when they are executed together to create production-like user load.

The user distribution across all the scenarios for different user loads will be as below:

|  |  |
| --- | --- |
| **Scenario** | **User Distribution** |
| Appointment Booking Full Flow | 25% |
| Enter/Edit Demographic Detail | 20% |
| Appointment booking for an existing application | 15% |
| Delete an application | 5% |
| Book Appointment on existing application | 5% |
| Cancel Appointment | 5% |
| Download acknowledgement | 10% |
| Rebook appointment | 15% |

The same percentage distribution will be used for varying concurrent user loads. Currently it is assumed that the user load is 100 on the current development environment and on the actual performance test environment the same workload distribution will be used to arrive at the target of 2222 transactions per hour.

The details of each of the scenarios considered in scope are given below:

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Steps** | **Description** |
| Appointment Booking Full Flow | |  | | --- | | Login | | Enter demographic details | | Upload the documents | | Book appointment | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Enter/Edit Demographic Detail | |  | | --- | | Login | | Enter/Edit demographic details | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Appointment booking for an existing application | |  | | --- | | Login | | Open existing demographic detail | | Upload the documents | | Book appointment | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Delete an application | |  | | --- | | Login | | Delete an application(whole data) | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Book Appointment on existing application | |  | | --- | | Login | | Select application (having demographic detail and documents pre-uploaded) | | Book appointment | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Cancel Appointment | |  | | --- | | Login | | View booked appointments | | Cancel an appointment | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Download acknowledgement | |  | | --- | | Login | | Download acknowledgement | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |
| Rebook appointment | |  | | --- | | Login | | View appointment | | Edit the appointment to rebook | | View acknowledgement | | Logout | | Flow for updating the booked time slots, The existing appointment slot is updated with a new one |

# TEST SCRIPTING

Test scripts will be created for the below sceanarios. For each scenario, the script will be a sequence of multiple API calls to each of the sub-modules.

Scripts will be developed for following scenarios:

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Category** | **Estimate(Hrs)** |
| Appointment Booking Full Flow | Complex | 6 Hrs |
| Enter/Edit Demographic Detail | Medium | 3 Hrs |
| Appointment booking for an existing application | Complex | 6 Hrs |
| Delete an application | Medium | 3 Hrs |
| Book Appointment on existing application | Medium | 3 Hrs |
| Cancel Appointment | Medium | 3 Hrs |
| Download acknowledgement | Medium | 3 Hrs |

# 

# DATA GENERATION NEEDS

Before executing the scripts, the data which is required to run the script should be populated in the database and also there is a requirement for input data. Performance team will create utilities using Java and JMeter to create test data wherever possible for document upload, demographic details etc. For few of the test data, the team will take help from the Automation team and other teams as needed.

# PROPOSED TEST EXECUTION APPROACH

Initially, the scripts will be created for all the individual APIs and will be combined as needed for the eight scenarios considered as scope. All the scripts will be run with full user load (100 users in dev) each for a fixed duration. This will be the first phase of testing which will cover all scenarios individually to uncover any performance issues.

In the second phase of testing, all the scripts will be run together for the required concurrent users with user distribution as determined by the workload model. This phase will be done in multiple iterations, each iteration being configured in such a way to meet the required TPS of 10 TPS, 20 TPS, 30 TPS, 50 TPS, 70 TPS and 100 TPS. The iterations may be stopped if it is found that the application cannot scale to higher TPS. Depending on the performance issues identified, this phase may also include running individual scenarios or a combination.

## BASELINE TESTS

Baseline tests for all the scenarios (listed in scripting section) will be carried out in the initial stage of testing to validate the test environment and to check the performance scripts for single user for all applications.

## TESTS WITH INCREASING USER LOAD

To identify the maximum throughput and maximum # of users supported by the application for particular scenario, tests will be carried out with increasing user load for all scenarios (listed in scripting section).

Tools

Following toolset will during the performance testing:

**Apache JMeter:** This will be used for scripting, execution and analysis

Custom built utilities built by the performance team or other teams within the project will be used to create test data and automate any steps required.

## 8.0.3 Resource Monitoring

Critical performance data that typically needs to be collected from load testing. For the pre-registration module load testing, metrics will be collected through native monitoring commands available with the OS. Microsoft Azure based monitoring will also be used if available. Metrics will be collected from Azure Application Insights if it is configured and made available on the performance test environment. Any kind of logs being generated will also be analyzed to check for any abnormalities or errors.

The below metrics will be monitored for the Application and Database server:

* CPU Utilization
* Memory
* Free Disk Space
* I/O operations
* Network I/O
* Application / Web / Database counters

Below are the SLA metrics for the CPU and Memory utilization on the servers

|  |  |  |
| --- | --- | --- |
| **Server Tier** | **SLA** | |
|
|
| **Tier** | **CPU** | **Memory** |
|  |
| Application Server | <65% | <75% |
| Database Server | <65% | <75% |

# DELIEVERABLES

Following deliverables will be sent to the customer, which are classified into two categories

* Test planning phase
* Performance Test Plan
* Work load model
* Effort Estimation
* Test execution phase
* Performance test script for all scenarios using Apache JMeter
* Intermediate performance test reports
* Comprehensive performance test report at the end of engagement

# TEST ENVIRONMENT

The load testing will be executed on Performance Testing environment. The details of which are below: <This needs to be updated based on the environment that is provisioned>

|  |  |  |  |
| --- | --- | --- | --- |
|  | **WebServer** | **AppServer** | **Database** |
|  |  |  |  |
| **Number Of nodes** |  |  |  |
| **RAM** |  |  |  |
| **PROCESSOR** |  |  |  |

The tests will be run from a JMeter instance within the VPC of the Cloud environment provided for testing.

# RESULT ANALYSIS & REPORTING

Basic analysis of the performance tests executed for all scenarios is expected in the phase-I of performance testing of Pre-Registration Module.

Following types of reports will be provided during the course of the engagement

* Intermediate performance test reports

Intermediate performance test reports will be generated after executing load tests for each scenario

* Comprehensive performance test report

At the end of the engagement comprehensive performance test report will be generated consolidating the summary of whole performance testing effort

# ENTRY, EXIT AND TEST STOP CRITERIA

Following are the entry, exit and test stop criteria’s specified for the Pre-Registration performance testing

**ENTRY CRITERIA**

The performance testing for Pre-Registration Module will start upon the completion of the following:

* All the required APIs are functionally stable and error-free
* Performance Test environment for testing is configured and ready
* The test environment should be uploaded with the latest application code to be tested
* The application should be populated with base data for testing

**EXIT CRITERIA**

The Exit Criteria will be as follows:

* Completion of load testing for all individual scenarios and mix of scenarios

**TEST STOP CRITERIA**

Following conditions are specified as test stop criteria and respective people are responsible for the decision of it:

| Description | Responsible |
| --- | --- |
| If the test objectives were not met and customer equests additional testing, the Performance Test Team will stop testing in order to allow Performance Test Team, customers and other partners to update the test plan and the schedule. | Mindtree Performance Team |
| Hardware Failure during Load testing | Mindtree Infrastructure Team |
| The application is not functional | Mindtree Application Development team |

# DEPENDENCIES

Following are the dependencies while executing the performance testing project. These dependencies should be highlighted in the initial stage and should be agreed upon by client and performance testing team for smooth execution of the engagement

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Dependency** | **Details** | **Priority** |
| 1 | Stability of Application Functionally | All modules, sub-modules and APIs of the Pre-Registration Module should be functionally stable before the starting the script for the scenarios | High |
| 2 | Unique version of the build should be present in the test environment | There should be unique build of Pre-Registration Module should be present in the test environment for first pass of testing | High |
| 3 | Build deployment | Stable build of Pre-Registration Module should be present for performance testing in the test environment | High |
| 4 | Pre-Registration Module build : provisioning | Before starting testing build deployed in the test environment should be populated with the reference data needed to execute the test | High |

# ROLE AND RESPONSIBILITIES

Following table lists out the roles and responsibilities defined for the Pre-Registration performance testing

|  |  |  |
| --- | --- | --- |
| **Role** | **Responsibility** | **Resource** |
|  |  |  |
| Program Manager | Reviews and Signs-off Performance Test Plan, summary report | Gita |
| Review all test deliverables and Sign-off |
| Manage the delivery |
| Overall responsibility for the Performance testing delivery |
| Performance Test Engineers | Prepare test summary report, metrics and share it with stakeholders | Gaurav Sharan/Shankar Narayanaswamy |
| Identify project risks, escalate and report to stakeholders |
| Understands test requirements and develop the test scripts accordingly |
| Execute performance tests and monitor the servers to identify potential bottlenecks |
| Provides intermediate report after each run to stakeholders |
| Test Manager | Review Performance Test Scenarios prepared by Performance team and Test Environment setup | Rekha N |
| Supports all other test phases by providing Business clarifications |
| Review result report after each execution and provide recommendation |
| Mindtree Infra Team | Access to servers | <TBD> |
| Access to server consoles |
| Monitoring the Servers if required |
| Providing metrics |

# RISK ASSESSMENT

Following risks are identified for Pre-Registration performance testing.

* 1. **SCHEDULE RISKS**

|  |  |  |
| --- | --- | --- |
| **Schedule Risk** | | |
| **Sr. No.** | **Risk** | **Mitigation** |
| 1 | In case of server failure during the course of load testing, the testing will come to an abrupt halt. Additional efforts are required to bring the environment back up and running | Contact infra team to bring up the server up |
| 2 | Any critical bottleneck will prevent further test execution till the issue is resolved by the development team. Additional time and effort may be required in detailed bottleneck analysis and its resolution | * Issue will be raised to respective support team * Development team will use the test environment for debugging purpose, hence respective delay should be considered in the performance testing schedule |
| 3 | Dependency on environment support team for issue resolution | Predefined SLAs for issue resolution. In case issue is not resolved within identified timelines; impact on schedule will be determined and a mutually agreeable mitigation will be arrived at. |
| 4 | Query Resolution and Issue resolution: In case, queries or issues raised by performance test team are not resolved within mutually agreed timelines it will have impact on schedule | * Dedicated point of contact should be available for performance test team’s query resolution * Definite timelines for issue resolution. In case issue is not resolved within identified timelines; impact on schedule will be determined and a mutually agreeable mitigation |

* 1. **RESOURCE RISKS**

|  |  |  |
| --- | --- | --- |
| **Resource Risk** | | |
| **Sr. No.** | **Risk** | **Mitigation** |
| 1 | Failure of server hardware | Identify alternate hardware configuration for testing purpose after consultation with client |
| 2 | Performance Test team will require clarifications from the Project Managers and the Development team during the process of strategy planning and execution. The availability of the above people at the right time is subject to risk. | The Performance Test Team will co-ordinate with team for discussions and clarifications |
| 3 | The initial number of load test machines may be inadequate for generating the requisite load levels during the course of performance testing engagement | Mechanism to scale up the required number of load testing machines. |

# LIMITATIONS & ASSUMPTIONS

* The tests will be executed on the test environment and not the exact replica of production environment or QA environment. This may have considerable impact on the final results
* The functional accuracy of the modules cannot be tested during the performance testing engagement. It is assumed that all the modules are functionally correct
* Data available in the current test environment will be used as base data for all the tests.
* There will be no major application failure. In case of an unrecoverable failure of the Pre-Registration Module Server, respective Infrastructure person will be responsible to make fully provisioned system available during the test execution phase in a reasonable timeframe. The timelines in this scenario may get affected accordingly

# APPENDIX