AT2

Movie Database Application

Performance report

Team Name:

Team Bare Maximum

Members:

Reece Pieri, Rico Imbang, Say Hon Lee

Course:

Diploma of Software Development – Rapid Application design

Contents

[1 Introduction 3](#_Toc56458457)

[2 Code Optimization 3](#_Toc56458458)

[2.1 About code optimization 3](#_Toc56458459)

[2.2 Advantages of code optimization 3](#_Toc56458460)

[2.3 Code optimization tool 3](#_Toc56458461)

[3 Software requirement 4](#_Toc56458462)

[4 Performance Testing 5](#_Toc56458463)

[4.1 What is performance testing 5](#_Toc56458464)

[4.2 Type of performance testing 5](#_Toc56458465)

[4.3 Common performance issue 6](#_Toc56458466)

[4.4 Performance Testing Process 7](#_Toc56458467)

[4.5 Performance Testing Metrics 8](#_Toc56458468)

# Introduction

The performance report is to collect information regarding the performance of the software, analyzing it, creating report documentation to present it to the stakeholders which is involved in the project. Performance report is part of the communication management plan.

# 2 Code Optimization

2.1 About code optimization

Code optimization is a programming technique which tries to help improve the code by helping it to reduce the consumption of the resources such as CPU and memory and also to allow the software to be delivered in a higher speed.

2.2 Advantages of code optimization

The advantages of the code optimization (Monus, 2020):

* Gives you cleaner code base
* Increase the consistency of the software
* Allow the software to run faster
* Increase code readability to allow the team to improve on better workflow
* Increase refactoring efficiency
* Easier code maintenance and allowing quicker feature development in future

2.3 Code optimization tool

For coding of the software, we could install and use the PHP CodeSniffer to implement in every file of the software as it is a application. PHP CodeSniffer is a application which could be installed in visual studio code which helps to do a code optimization.

# 3 Software requirement

The movie website should have the following function

* Home

The home page of the website supposed to display all the movie from the database.

* Search

The search functions allow you to search for a particular movie using by indicating the title of the movie or genre or rating or year of the movie.

* Membership subscription

Users should be able to subscribe for monthly newsletter and breaking newsflash.

* Admin login

The administrator should be able to login on the admin page to view the members which subscribed and members which request to unsubscribe for the newsletter or breaking newsflash.

# 4 Performance Testing

4.1 What is performance testing

Performance testing is a software testing process. The purpose of it is to test the speed, response, time, stability, reliability, scalability, and resource usage of the software under a particular workload. The main purpose of this test is also actually to identify and eliminate the performance bottleneck in the software application. It is a subset of performance engineering which is known as “Perf Testing”.

Speed testing – To determine the response of the application

Scalability testing – To find out the maximum user load available for the software to handle.

Stability testing – To determine if the application is stable under varying load.

4.2 Type of performance testing

* Load testing – The purpose of this test is to check the software ability while performing under anticipated user loads. The objective is to identify performance bottleneck before the software could goes live in future.
* Stress testing – The purpose of this application involve testing the software under extreme workload to see how the software handless with high volume of traffic or while the data is processing.
* Endurance testing – The purpose of this test is to ensure that the software is able to handle the expected load over a long period of time.
* Spike testing – The purpose of this test if to check if the software could react to a sudden large spike in the load that is generated by the users of the software
* Volume testing – The purpose of this test is to test large number of data is populated in a database and the overall software systems behaviour Is monitored. The objective is to check the software application performance under varying database volumes.
* Scalability testing – the purpose for this testing is the check the software application effectiveness while “scaling up” to support the increase in user load. It helps to plan the capacity addition to the software system.

4.3 Common performance issue

Below is a list of the common performance issue with a web application which could cause the company to lose potential users.

* Excessive load time – The load time is usually the time taken for the application to start up. It should be kept to the minimum. While some application are impossible to make load in under a minute, load time should be kept under a few seconds if it is possible.
* Poor response time – The response time is the time the application take when a data is input into the application to the time where the application outputs a response to the input. Generally, it should be done as soon as possible, if it takes too long, users might lose interest as well.
* Poor scalability – Poor scalability affect the software product if it is unable to handle the expected number of users or when it is not able to accommodate a wide enough range of users. Load testing should be done to be certain the application have the ability to handle the anticipated number of users.
* Bottlenecking – Bottlenecking is a obstruction in a system which degrade the overall system performance. Bottlenecking is when either coding errors or hardware issues cause a decrease of throughput under certain loads. Bottlenecking is often caused by one faulty section of code. In order to fix a bottlenecking issue is to find the section of code that causes the slow process and fix it. It is generally fixed by either fixing the poor running processes or adding additional hardware. The common performance bottlenecks are:
  + CPU utilization
  + Memory utilization
  + Operating system limitation
  + Disk usage

4.4 Performance Testing Process

There is a wide variable for performance testing, but the objective for performance test remains the same. It helps to demonstrate that your software system is able to meet certain pre-defined performance criteria. Or it can help compare the performance of two different software system. It also could help identify part of the software system which degrade the performance.

The steps to performance testing process is to:

* Identify the testing environment – knowing what’s your physical test environment, production environment and what testing tools are available. Understand the details of the hardware, software and network configuration used during testing before you begin the testing process. It helps to identify possible challenges that users may encounter during the performance testing procedure.
* Identify the performance acceptance criteria – this includes goals and constraints for the throughput, response time and resource allocation. It is necessary to identify the project criteria outside of the goals and constraints. Testers should be empowered o set performance criteria and goals as the project specification might not includes a wide enough variety of performance benchmarks. It is also a good way to find a similar application to compare which helps to set a performance goal.
* Plan and design performance test – this determine the usage amongst the end user and identify key scenarios to test for all possible use cases. It is necessary t simulate a variety of end user, plan performance test data and outline what metric will be gathered.
* Configure test environment – This is to prepare the testing environment before executing the software. Also, to arrange tools and other resources.
* Implement test design – Create the performance test according to the test designs.
* Run the test – execute the software and monitor the test.
* Analyse, tune and retest – consolidate, analyse and share the test result. If needed, fine tune the software and test it again to see if there is and improvement or decrease in performance of the software. Improvements generally grow smaller with each retest, stop when bottlenecking is caused by the CPU and consider the option of increasing the CPU power.

4.5 Performance Testing Metrics

This refers to the basic parameter that will be monitored during the performance testing.

* Processor usage
* Memory use
* Disk time
* Bandwidth
* Private bytes
* Committed memory
* Memory pages/ second
* Page faults/second
* CPU interrupts per second
* Disk queue length
* Network output queue length
* Network byte total per second
* Responds time
* Throughput
* Amount of connection pooling
* Maximum active sessions
* Hit ratios
* Hits per second
* Rollback segment
* Database locks
* Top waits
* Thread counts
* Garbage collection