Sudan University of Science and Technology

Faculty of Computer Science and Information Technology

Web-Based Application Performance Monitoring Tool

October 2017

THESIS SUBMITTED AS A PARTIAL REQUIREMENTS OF B.Sc.(Honor)

DEGREE IN SOFTWARE ENGINEERING

بسم الله الرحمن الرحيم

Sudan University of Science and Technology

Faculty of Computer Science and Information Technology

Web-Based Application Performance Monitoring Tool

OCTOBER 2017

PREPARED BY:

AMEENAH KHALID ABUSINNAIN

THEGA ALAMEEN MOKHTAR

MARSA ABUTALIB YOUSEF

MAJEDA TAJELSIR MERGHANY

SUPERVISOR:

ASHWAG MOHAMED

THESIS SUMITTED AS A PARTIAL REQUIREMENTS 0F B.Sc. (HONOR) DEGREE IN SOFTWARE ENGINEERING

SIGNATURE OF SUPERVISOR: DATE:

……. …. OCTOBER 2017

الآية :

قال عز و جل :

" قُل لَّوْ كَانَ الْبَحْرُ مِدَادًا لِّكَلِمَاتِ رَبِّي لَنَفِدَ الْبَحْرُ قَبْلَ أَن تَنفَدَ كَلِمَاتُ رَبِّي وَلَوْ جِئْنَا بِمِثْلِهِ مَدَدًا "

[199]الكهف

الحمدلله

اللَّهُمَّ لك الحمْدُ حَمْدًا يمْلأُ المِيْزان، ولك الحَمْدُ عَدَدَ ما خَطَّهُ القَلَمُ وأحْصَاهُ الكِتَابُ ووَسِعَتْهُ الرَّحْمَةُ.

اللَّهُمَّ لك الحمْدُ على ما أعْطَيْتَ ومامَنَعْت، وما قَبَضْتَ وما بَسَطْـتَ.

اللَّهُمَّ لك الحَمْدُ على كُلِّ نِعْمَةٍ أنْعَمْتَ بِهَا عَلَيْنَا في قَدِيْمٍ أوْ حَدِيْثٍ، أوْ خاصَّةٍ أوْ عَامَّةٍ أوْ سِرٍ أوْ عَلانِيَةٍ او حي او ميت او شاهد او غائب .

الإهداء

إلى أبائنا الذين يتكبدون المشاق من أجل إعطائنا الحق كاملا لنهل العلم والمعرفة

....

إلى أمهاتنا اللاتي يؤمن بقدراتنا علي مجابهة الصعاب واللاتي لم يبخلن علينا بالدعاء والرجاء حتي نتمكن من إكمال هذا العمل

...

الي إخوتنا وأخواتنا الذين واظبوا علي تشجيعنا

...

إلى زملائنا وزميلاتنا الذين لم يترددوا في تقديم المساعدة عند الحوجة

...

نهدي هذا العمل المتواضع ونرجو الله عز وجل أن يعطينا القوة لنرد لهم جميل صنعهم

شكر و عرفان

أ. محمد نافع

للنّجاح أناس يُقدّرون معناه، وللإبداع أُناسٌ يحصدونه، لذا نقدّر جهودك المُضنية، فأنتَ أهلٌ للشّكر والتّقدير ووجب علينا تقديرك، لك منّا كلّ الثّناء والتّقدير

...

أ. اشواق محمد

تتسابق الكلمات وتتزاحم العبارات لتُنظّم عقد الشّكر الذي لا يستحقّه إلّا أنت، إليك يا من كان له قدم السّبق في ركب العلم والتّعليم، إليك يا من بذلت ولم تنتظر العطاء

ABSTRACT

The web application performance become worse due to many reasons such as, the application code, database, servers, and network. Any one of these reasons or all of them can cause problems in application performance.

The aim of the research is to help developers of web application by providing them the needed information related to the application performance, this will help the developer to apply root cause analysis on performance problems, and solving them.

The research found that to be able to resolve performance problems, the developer need to know some information such as: execution time of every SQL query to determine the most expensive, the memory consumption and function runtime and also be able to deal quickly with errors that occurred in real environment before the user of the application observing it.

The APM(Application Performance Monitoring) tool helping the developer to get all these information by collecting them from different resources such as: MySQL server and PHP log file, but still the developer need to analyze the causes of the problem and solving them manually, so the researcher recommends other tools desired to work in the same field to find out to solve problems automatically.

المستخلص

تطبيقات الويب يصبح أدائها سيئا مع مرور الوقت لعدة أسباب منها : شفرة التطبيق، قاعدة البيانات، الخادم و الشبكة. أي من هذه الاسباب أو كلها مجتمعة يمكن أن تتسبب في تقليل أداء التطبيقات، الهدف من هذا البحث هو مساعدة مطوري تطبيقات الويب و ذلك بتوفير المعلومات الكافية عن أداء التطبيق و بالتالي يتمكن المطور من تحليل المشكلة وحلها.

توصل البحث إلى أنه حتى يتمكن المطور من حل مشاكل الأداء لابد من توفر بعض المعلومات مثل: الزمن الذي تحتاجه كل تعليمة استعلام ليتم تنفيذه ، مقدار استهلاك التطبيق من الذاكرة و زمن تنفيذ كل دالة أو وظيفة بالإضافة إلى ذلك يجب على مطور التطبيق التعامل بسرعة مع الأخطاء التي تحدث في بيئة التشغيل قبل ان يتم ملاحظتها من قبل مستخدمي التطبيق.

الآداة التي تستخدم في مراقبة أداء تطبيقات الويب التي قام بتطويرها الباحثون تساعد المطورين في الحصول على المعلومات المطلوبة وذلك بتجميعها من عدة موارد مثل: قاعدة البيانات، الخادم، ملفات الآخطاء ولكن لا يزال على المطور تحليل هذه المعلومات و حل المشاكل يدويا، عليه يوصي الباحثون الأشخاص الذين يرغبون في العمل في نفس المجال أن يجدوا طريقة لحل هذه المشاكل بصورة آلية دون الحاجة إلى تدخل المطور .

GLOSSARY

#	TERM	DESCRIPTION
1	APM	Application Performance Monitoring
2	OS	Operating System
3	IT	Information Technology
4	EUE	End User Experience
5	RUM	Real User Monitoring
6	ITIL	Information Technology Infrastructure Library
7	JVM	Java Virtual Machine
8	SQL	Structured Query Language
9	CPU	Central Possessing Unit
10	TPS	Transaction Processing System
11	UI	User Interface
12	PHP	Hypertext Preprocessor

13	HTTP	Hypertext Transfer Protocol
14	RDBMS	Relational Database Management System
15	API	Application Programming Interface

TABLES OF FIGURES:

TABLE OF TABLES:

TABLE OF CONTENETS:

Contents

CHAPTER ONE

Introduction

1.1 Introduction:

A web application is a client-server application run by the client in a browser. The main function of a browser is to show the information received from a server and send the user's data back. The main advantage of this approach is the fact that clients do not depend on the user' specific operating system; therefore, web applications are cross-platform services. Due to this universal feature, web apps became very popular in 1990s and 2000s. Developers do not need to prepare different versions of the same app for Microsoft Window, Mac OS, Linux, etc. An app is created only once for any platform and it can work on any operating system.

The performance of a web application plays a critical role in how an application is perceived by its users. It is important to measure it, identify the causes if it changes and react swiftly to any unexpected changes.

For developers asking what is application performance monitoring, it may be said that APM provides them first and foremost with visibility into the entire application architecture or production environment. APM helps development teams find and fix potential bottlenecks in the application.

1.2 Problem Statement:

It's inevitable that web application problems are going to happen, problems can come from anywhere, and sometimes you just need to know where to look. So it's important to remember that the best way to address performance problems is to find and eliminate them before they affect users in the first place.

1.3 Research Objective:

The primary objective, is to develop web applications performance monitoring tool in order to help the web developers to detect and identify performance problems that can be occur while the application in real environment.

1.4 Importance of Research:

Developing web based tool that help developers in terms of:

- Providing various metrics about the performance of your production site, covering everything from application database queries through to the time it takes for the end-user to view a page.
- Converting collected data to simple and clean charts presented in the web interface.
- Presenting the data stored in the error log and grouping them to be simpler in monitoring errors that occurs in real-time environment.

1.5 Research Scope:

The first and the foremost, the scope of this research is limited only to PHP web application, and the targeted audience are web developers. This research aims to help the developers by giving them information about the performance of the SQL queries in real time environment and representing them in simple chart, monitoring the source code of the application, and also help in managing the PHP error log.

1.6 Research Methodology:

In this section we will talk briefly about steps, stages and selected tools and techniques that have been taken to perform the work of implementing the tool and they will be detailed later.

The steps:

1- Collecting the requirements.

2- Analyzing collected requirements.

3- Design

4- Implementation

5- Testing

1.7 Thesis Outline:

In addition to this chapter this research contains another five chapters:

- Chapter two contain the related studies in the field which the tool will be developed in.
- Chapter three contains describe how the tool analyzed
- Chapter four contains describe how the tool designed and implemented
- Chapter five contains the case study of the system
- Chapter six contains results of the research and recommendations can be done to evaluate the tool.

CHAPTER TWO

Literature Review

2.1 Introduction:

There are many performance testing tools and solutions that have been developed for measuring and testing non-functional performance of the application, and they are deployed in pre-production environment. Performance testing tools ensure the performance of an application with the forecasted transaction volumes; however continuous availability of application in production environment is not being catered by performance testing solutions. Therefore, monitoring solutions came into picture, which is deployed on production environment for ensuring application performance and availability.

2.2 Application performance monitoring:

An APM tool is usually used to monitor the performance and the availability of a monitored web-based software application. An APM tool collects several performance metrics (such as response time) from the monitored application and mines these metrics to measure the health of the application (e.g., identify potential performance problems using mining approaches). Most of the metrics that are mined by the APM tools are used in performance regression detection research as well. Hence, APM tools might be effective in detecting performance regressions using these metrics. [3]

2.3 Elements of Application performance monitoring:

The main objective for APM is to translation IT metrics into business meaning (value), since it's represent the End User Experience (EUE) measurements which is high impact indicator of demonstrating productivity.
The four foundational elements that affect APM adoption center around the EUE are [7]:

- Top Down Monitoring (Real User Monitoring RUM)
- Bottom Up Monitoring (Infrastructure)
- Incident Management Process (ITIL)
- Reporting (Metrics)

Top Down Monitoring:

Top Down Monitoring is also referred to as Real-time Application Monitoring that focuses on the End-User-Experience. A real-time monitor enables us to observe the

behavior of the production, in order to collect genuine data, usable for statistics, or to detect anomalies and illegal states [7][8].

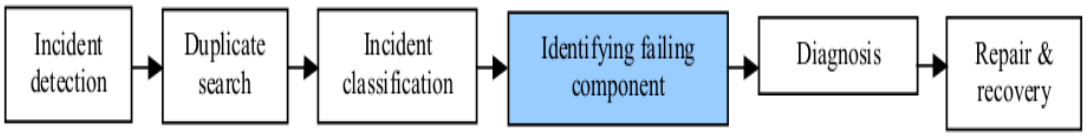
Bottom Up Monitoring:

This of monitoring should be in place for all nodes/servers within the environment. Automation the system is the key component to the timeliness and accuracy of incidents being created.

Incident Management Process:

The aim of the incident management process is quickly resolving incidents that affect the normal running of an organization's IT services. An incident is an intimation of some error or failure of some component in IT systems. [7][9]

Figure 2-1 incident management workflow



Reporting Metrics:

It is important to arrive at a common set of metrics that must be collect and then standardize on a common view on how to present the real-time performance data.

2.4 Application performance monitoring activities: [6]

- The observation of the behavior of the application, physical hardware, virtual machine, JVM, and containers(application server or web container).

- The collection, interpretation, and analysis of performance metrics across an entire application environment.

- The capture of behavior abnormalities and informing them using alerts .

- Recommendation of acting to adapt application environment to remediate performance problems.

2.5 Performance Analysis Approaches [3]:

To determine the anomalies based on performance metrics, Most APM tools use two general mining approaches, either using only one approach or using both approaches, which are:

1) baseline-based
2) threshold-based

Baseline-based Approach:

APM tool establish a baseline based on learned behavior of the application, since the workload is expected to variance during different working hours, in this approach APM tool detect metric values that deviate from this baseline.

There are different mining tools to establish the baseline:

- Uses the average value of a metric observed during a specific time range to de ne the baseline.
- Statistical techniques such as the 90th percentile and binomial distribution to calculate the baseline.

APM tools use these baselines as indicators to notify practitioners when their application performance deviates from the baselines.

Threshold-based Approach:

A threshold is Minimum or maximum value which serves as a benchmark for comparison or guidance and any breach of which may call for a complete review of the

situation or the redesign of a system. It can be calculated using simple statistical methods or it can be configured either by the software practitioners or by the APM tool itself.

A threshold can be the transaction response time, failure rate, or throughput. APM tools support the following types of thresholds:

- **Percentage deviation threshold:** When a metric value exceeds a specific percentage above the metric's average.

- **Standard deviation threshold:** When a metric value exceeds multiples of standard deviations above the metric's average.

- **Fixed threshold:** When a metric value exceeds a predefined fixed value.

2.6 Related Studies:

2.6.1 New Relic:

New Relic's software product for application performance management (APM) delivers real-time and trending data about web or non-web app's performance. New Relic uses a threshold-based approach to detect anomalies [9].

Some of New Relic’s advantages:

- New Relic offers an interface for debugging the server side of performance issues. While running on your servers it keeps track of the entire run time stack and gives a detailed report on issues. This allows you quickly to identify performance issues: which queries take too long, which query is called too many times, what function is running for way too long. All these issues become apparent for you to fix.

- Get accurate real user page load times. This is valuable because it allowed us to see exactly how our web pages were performing all across the world. Real user page performance tracking allowed us quickly to see if there were particular users that were getting a worse experience than others.

Some of New Relic's disadvantages:

- The amount of data you get about the web page loading is not very detailed. The tool does not really allow you to debug very well what on the web page was causing the page to render slow; it will just notify you that the actual web page is rendering slowly

- New Relic is all about setting acceptance thresholds and then giving you detailed reports on requests that are not in that acceptable range. If requests are in that acceptable range, but you want to look at detailed reports, you will not see any. That means you have to constantly lower your threshold to continue improving your site.

2.6.2 Pinpoint:

Pinpoint is an open-source application performance monitoring dedicated to large-scale distributed systems specifically Java-based software, Pinpoint is threshold-based approach to detect performance anomalies which depend on multi agent component that must be installed to collect and analysis performance metrics [10].

Some of Pinpoint's advantages:

- Since it directed to distributed systems, Pinpoint traces transactions flows between different component and provide clear view to identify problem areas and potential bottlenecks.

- Inspection through viewing additional details on the application such as CPU usage, Memory/Garbage Collection, TPS, and JVM arguments.

Some of Pinpoint's disadvantages:

- Pinpoint is On-premise APM which require 3 components to installed and configured, the collector, the web UI and the agent. Configuring that component is usually difficult and requires effort must be presented by IT specialist.

2.6.3 AppDynamics:

AppDynamic is a commercial application performance monitoring product, available as on-premises and cloud-based service [11].

Some of AppDynamics' advantages: [6]

- The Application Flow Map within AppDynamics displays servers and logical services, such as queues and metrics about the performance of those components. This provides a succinct representation of the relation between components and the number of calls between them. This feature is especially useful for root cause analysis.

- Provide the availability for reporting high level business transaction, which is helpful for tracking performance and impact by application, such as checking out or searching for a product.

Some of AppDynamics' disadvantages:

- AppDynamic is complex for medium and small size company, it's dedicated for complicated product environments, and there is no availability of using only the parts of the product that they would need based on the functionality needed at the time.

2.6.4 Datadog:

Datadog is a monitoring service that brings together metrics and events from servers, databases, applications, tools and services to present a unified view of the infrastructure.
These capabilities are provided on a SaaS-based data analytics platform that enables teams to work collaboratively on the infrastructure to avoid downtime, resolve performance problems, and ensure that development and deployment cycles finish on time [12].

Some of Datadog' advantages:

- Datadog offers customizable dashboards for metric correlation and flame graphs for quickly and accurately identifying the most frequently used code paths.

- Datadog consider resolving problems require examining changes in both the infrastructure and code simultaneously.

Some of Datadog' disadvantages:

- Complete lack of complex alerting of specific metrics by setting a threshold value.

- Datadog does not support log storage, it can create a parser that parses a log file and turns it into metrics or events, but it still ends up in Datadog as standard metrics.

Table 2-1 Summary of relates studies

APM tool name	advantages	disadvantages
New Relic	1. debugging the server side of performance issues 2. accurate real user page load times	1. No detailed data, just abstract problem informing. 2. No informing about acceptable range values.
Pinpoint	1. Tracing of transactions in distributed system. 2. Detailed resource monitoring (CPU, memory, etc.).	1. difficult configuration of pinpoint components
AppDynamics	1. Displays servers and logical services. 2. reporting high level business transaction	1. dedicated only for complicated product environments
Datadog	1. Customizable dashboards for metric correlation and flame graphs. 2. resolving problems in both the infrastructure and code	1. Lack of alerting of specific metrics. 2. does not support log storage

CHAPTER THREE

System Analysis

3.1 Introduction:

The Analysis Phase is where the project lifecycle begins. The Analysis Phase is where you break down the deliverables in the high-level Project Charter into the more detailed business requirements. System design is the process of defining the components, modules, interfaces, and data for a system to satisfy specified requirements.

This chapter describes the functional and nonfunctional requirements of the system and how the system will be analyzed and designed.

3.2 Collecting the requirements:

The requirements of the web applications monitoring tool are collected by dealing and observing many tools in this field, and depending on that we find out that developers need to know some information about their applications to be able to identify performance problems that can be happen to this applications in production environment.

3.3 Requirements Analysis:

After the requirements collection process, the tool's requirements collected (by observation of developed web applications monitoring tools) will be analyzed to identify the basic components of the proposed tool .and determine which components will be developed from scratch by researchers and which one will be chosen from an existing open source component and then integrated to tool, to develop the tool researchers identify the components, which they are:

- SQL query performance information collector component
- SQL query performance information visualization component
- PHP error log management component
- Code Profiling component

3.4 System Requirement

The requirement does not specify the architectural or implementation details, but specifies information at the higher level of description. The problem statement, the customer's expectations, and the criteria for success are examples of high-level descriptions.

3.4.1 Functional Requirements

3.4.1.1 SQL Monitoring:

- **Query metrics collection function:**

  The tool must provide collector component that collects information about every query that come from the monitored application to the server.
  This information can help the developers to find out what is the problems in their applications, table 4.1 include some information about query.

Table 3-1 show some query information

Metrics name	description
Start Time	Time the query began
Lock time	Total time the query was locked
Rows sent	Total time the query took to execute
Query time	Total time the query took to execute
Rows examined	Number of rows examined
SQL text	Fully query

Table 4.1 show some query information

- **Detect expensive query function:**

  This function deals with the collected information and analyzes them to detect the slow queries. The slow queries are specified based on the user threshold called Long Query Time, if a query takes longer than this many seconds, the server increments the slow queries status variable. If the slow query log is enabled, the query is logged to the slow query log file. This value is measured in real time, not CPU time.

Table 3-2 describe the long-query time

Command-Line Format	--long-query-time=#	
	Name	long-query-time
	Variable Scope	Global, Session
	Dynamic Variable	Yes
Permitted Values	**Type**	numeric
	Default	10
	Min Value	0

Table 4.2 describe the long-query time

3.4.1.2 Visualization function:

APM must provide component that receive the analyzed information and convert them into charts.

The Visualization show a comparison between the current performance statistics and previous statistics and that give the developers indicators whether their applied solution successes in improving the application performance or not

3.4.1.3 Performance prediction function:

Increasing in data size will increase the execution time of the application's SQL queries and degrade its performance, So Based on the analyzed metrics APM must compute queries cost and resources to notify the admin of performance issues.

3.4.1.4 Code Profiling:

- **memory allocation and leak:**
  APM will analyze the monitored application to identify script consumption of memory.
- **function run time:**
  Track time spent in functions self-cost or inclusive cost, inclusive cost is time inside function calls to other functions.

3.4.1.5 Error Tracking:

APM must be able to trace all errors occur in monitored application, notifying the admin, grouping them depending on their categories, calculate the ratio of every error based on specified period and give the opportunity to share that error by email to the developer or development team.

3.4.2 Non-Functional Requirement

3.4.2.1 1. Usability:

Users should have no trouble using this APM Tool.

3.4.2.2 Performance:

The system must be interactive and the delays involved must be less.

3.4.2.3 Availability:

The service is to be available to all users at any time.

3.5 System Analysis

3.5.1 System Analysis Using UML

3.5.2 Use case Diagram:

Figure 3-1 Use Case Diagram

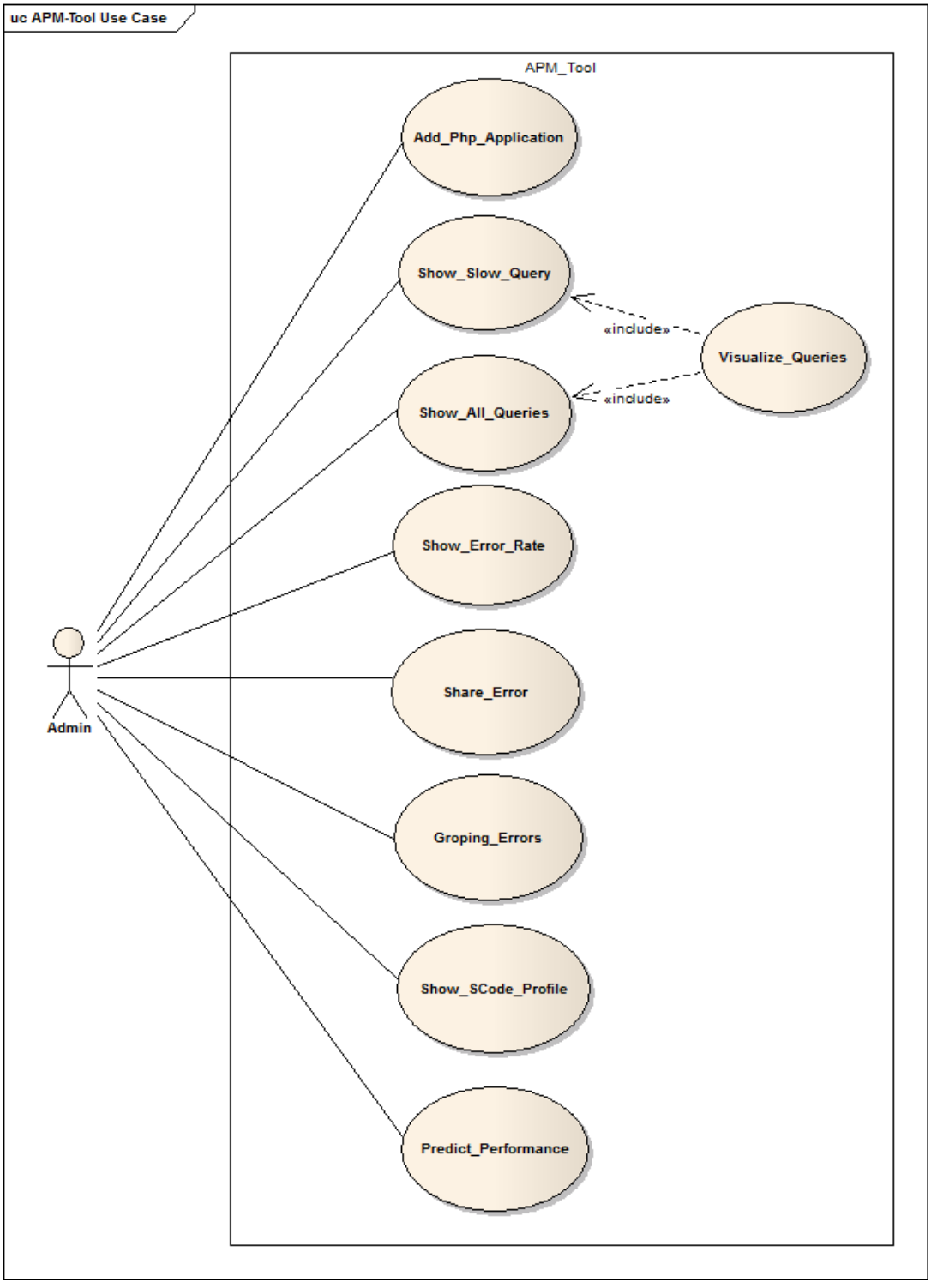


Table 3-3 use case description

Add_Php_Application	The IT Administrator will be able to add php application by entering the database name and the tool start collecting all query data related to specified database, processing and storing them in APM database to be presented later when required.
Show_Slow_Query	The IT Administrator will be able to show slow queries (queries that took more than specific time) for his/her php application database.
Show_All_Queries	The IT Administrator will be able to show all Query(query text, total number of query, average query execution) for his/her php application database.
Show_Error_Rate	This function will display the rate for each error group to the IT Administrator
Share_Error	The IT Administrator will be able to share error by email to the application developer.
Groping_Errors	The IT Administrator will be able to show all errors in group(fatal errors group, warning errors group, exception error group ,and parse error group).
Show_Code_Profile	The IT Administrator will be able to show function run time and memory consumption for his/her php application.
Predict_Performance	The IT Administrator will be able to predict query throughput.
Visualize_Queries	This function will display all queries data in simple charts.

3.5.3 Sequence Diagrams:

Figure 3-2 Add Application component sequence

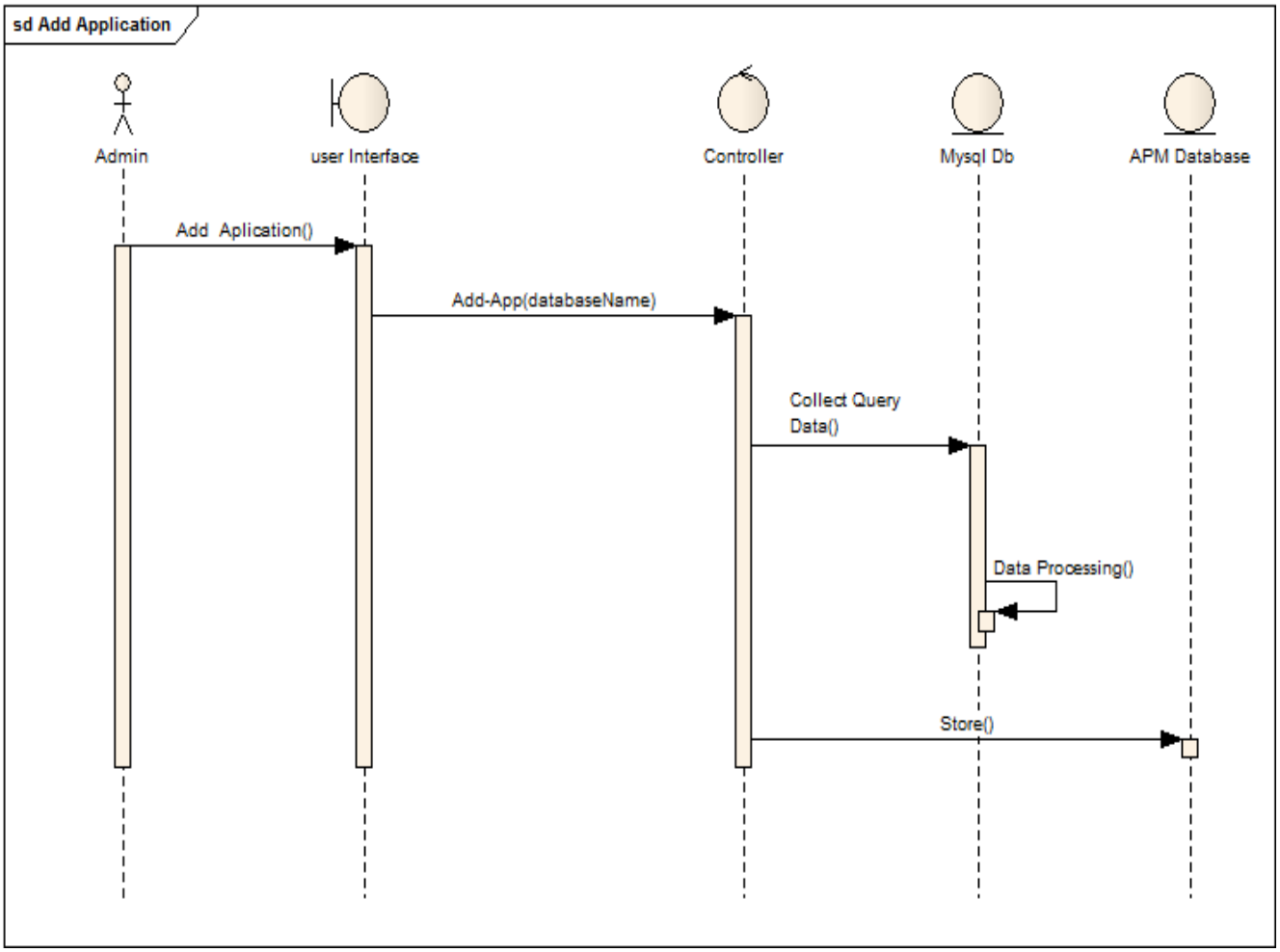


Figure 3-3 Expensive queries

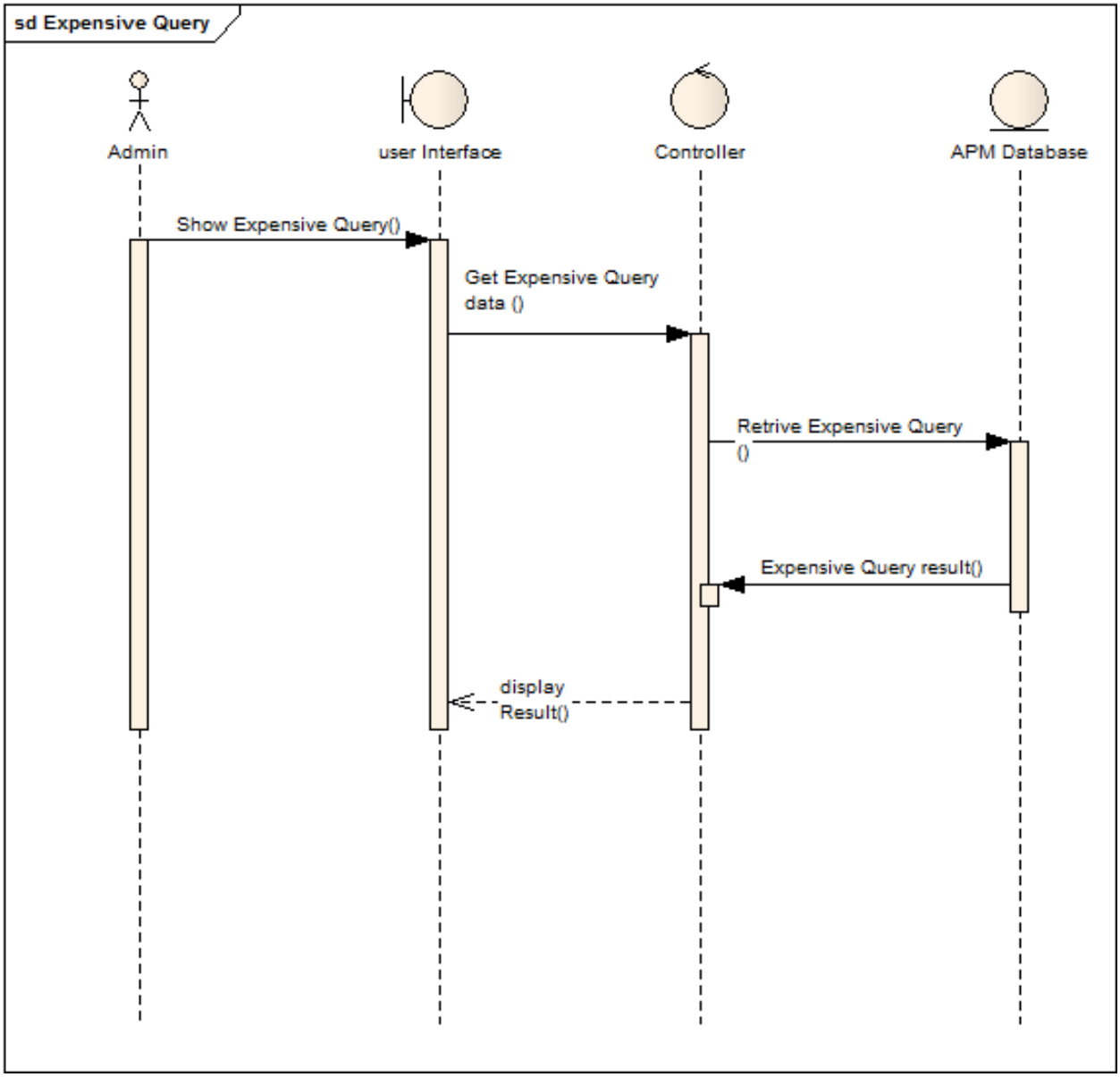


Figure 3-4 all queries

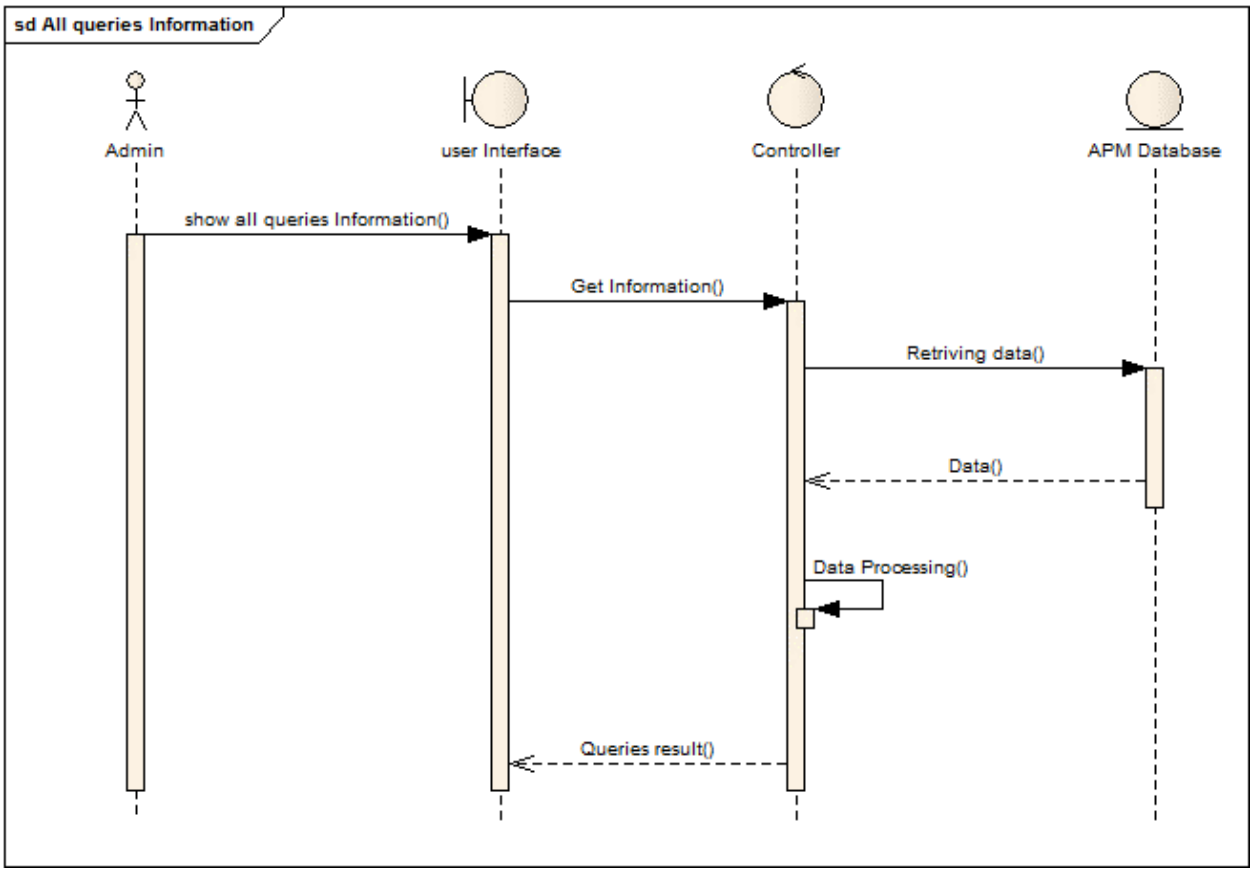


Figure 3-5 visualization metrics sequence

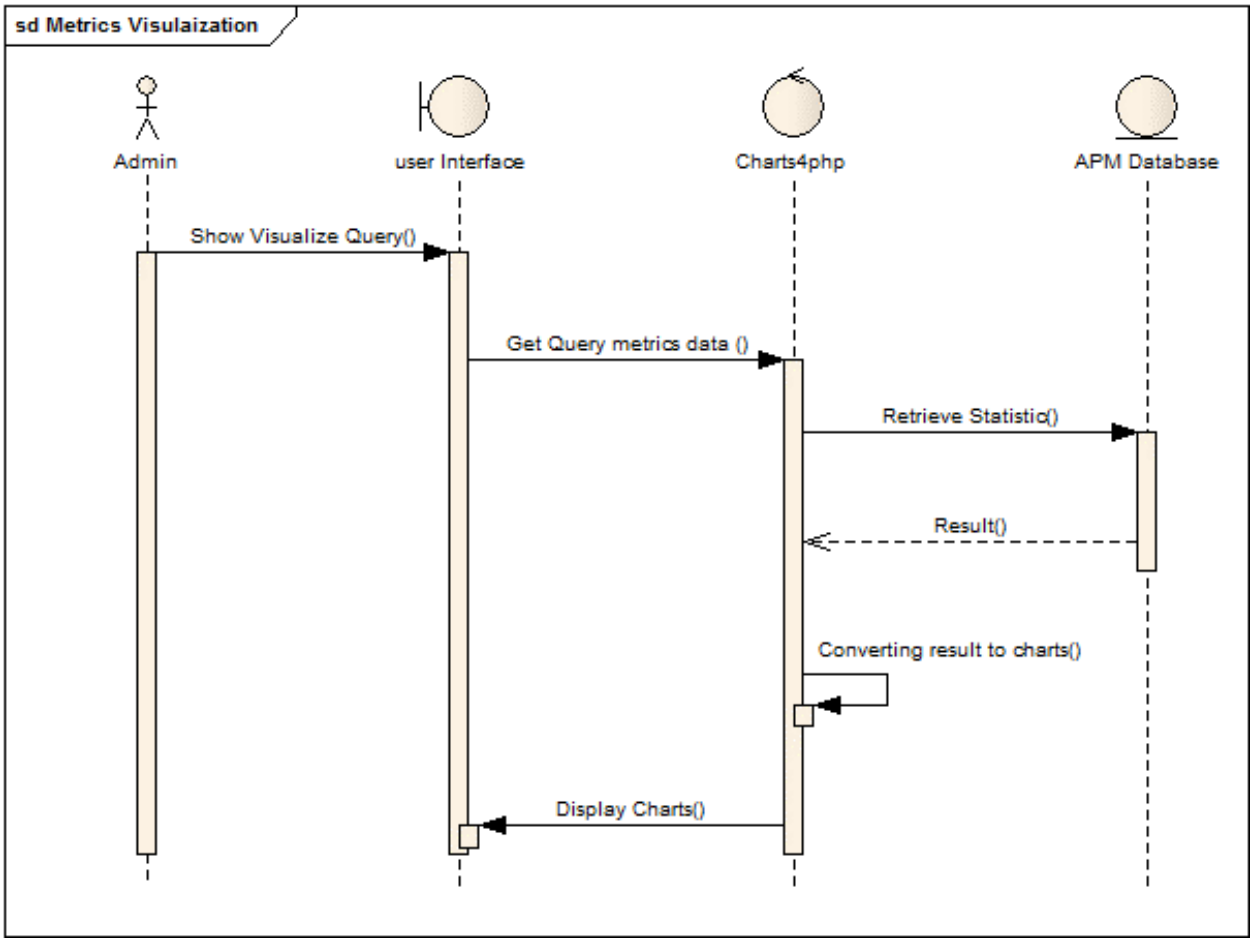


Figure 3-6 error rate sequence

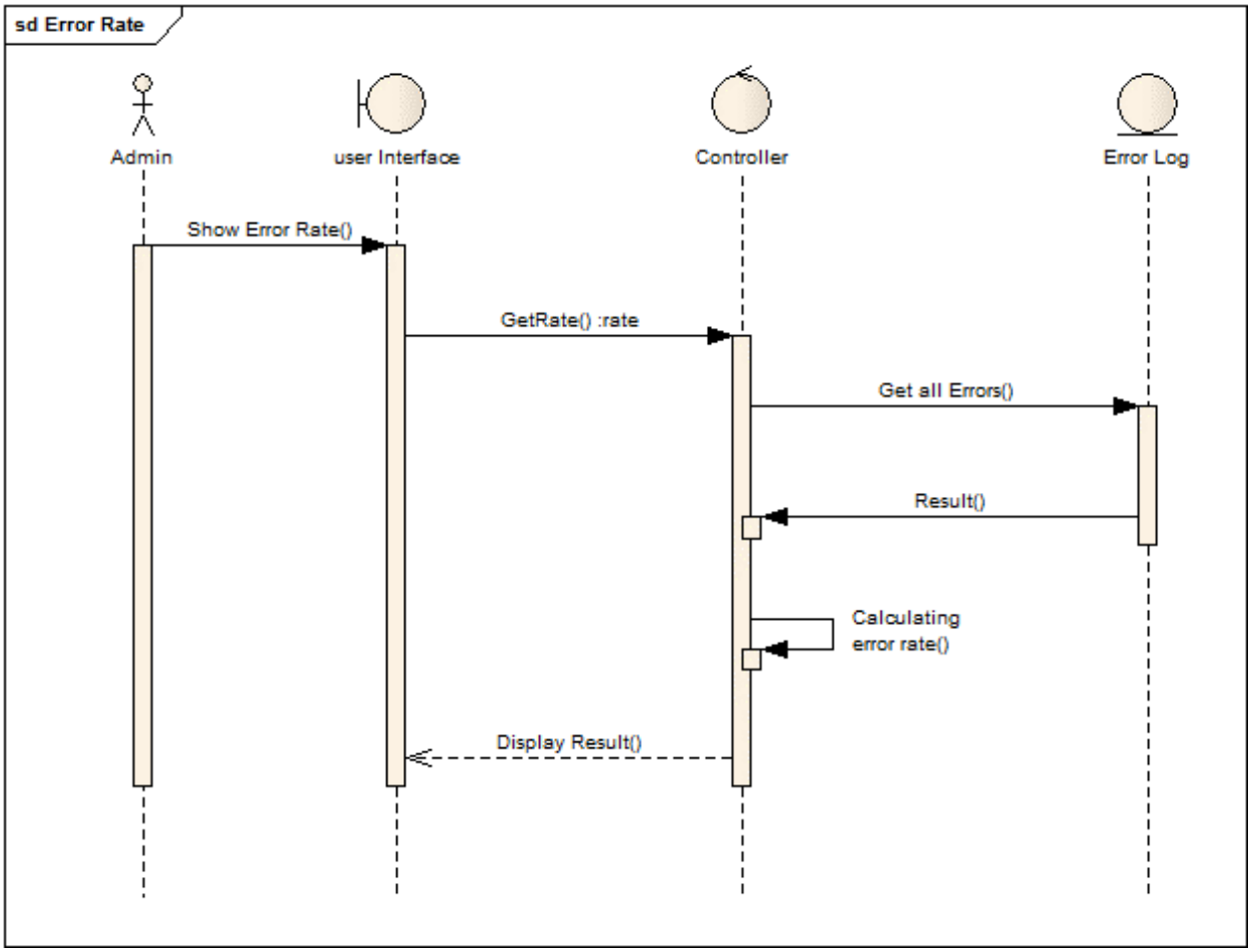


Figure 3-7 share error sequence

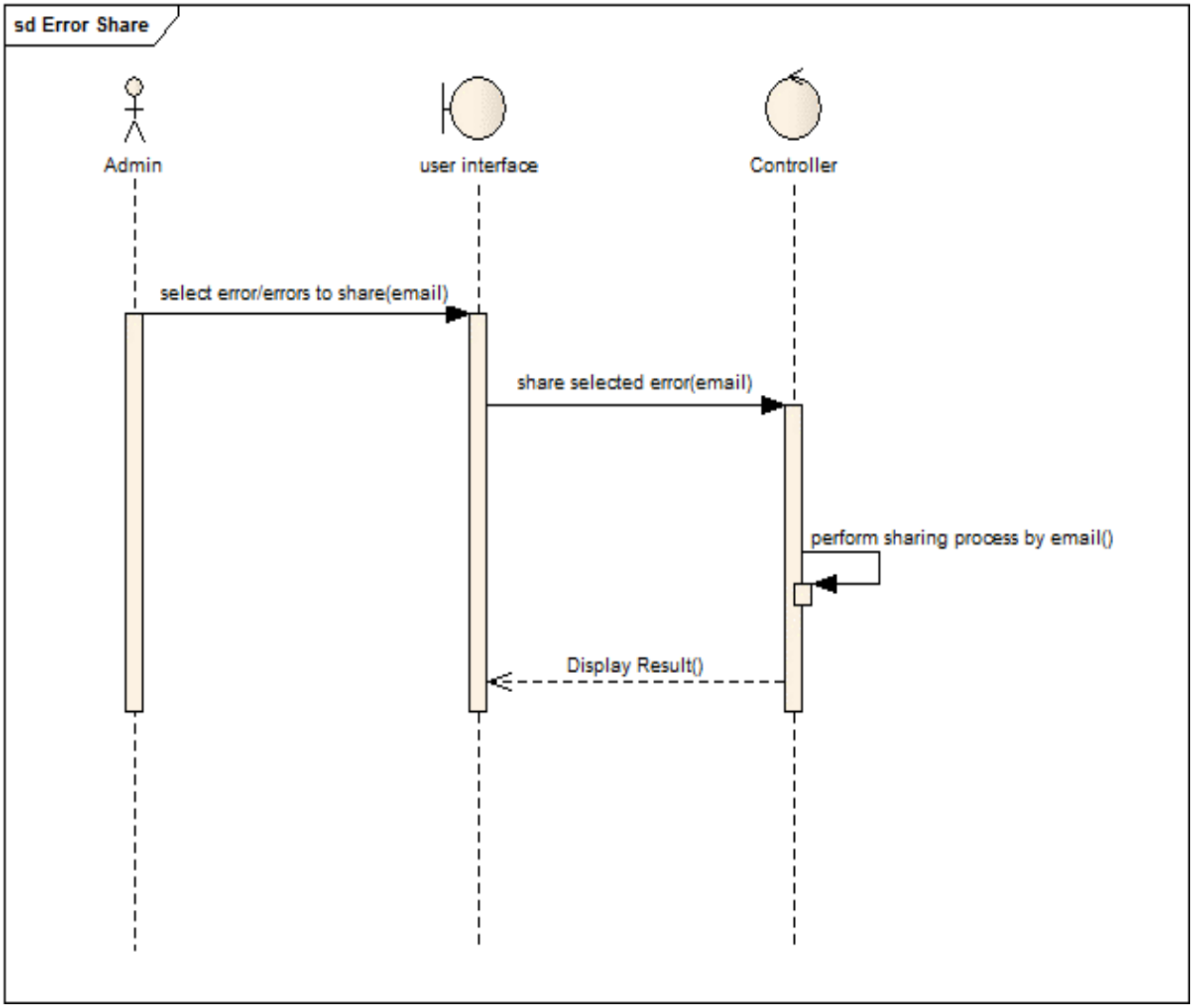


Figure 3-8 Grouping errors

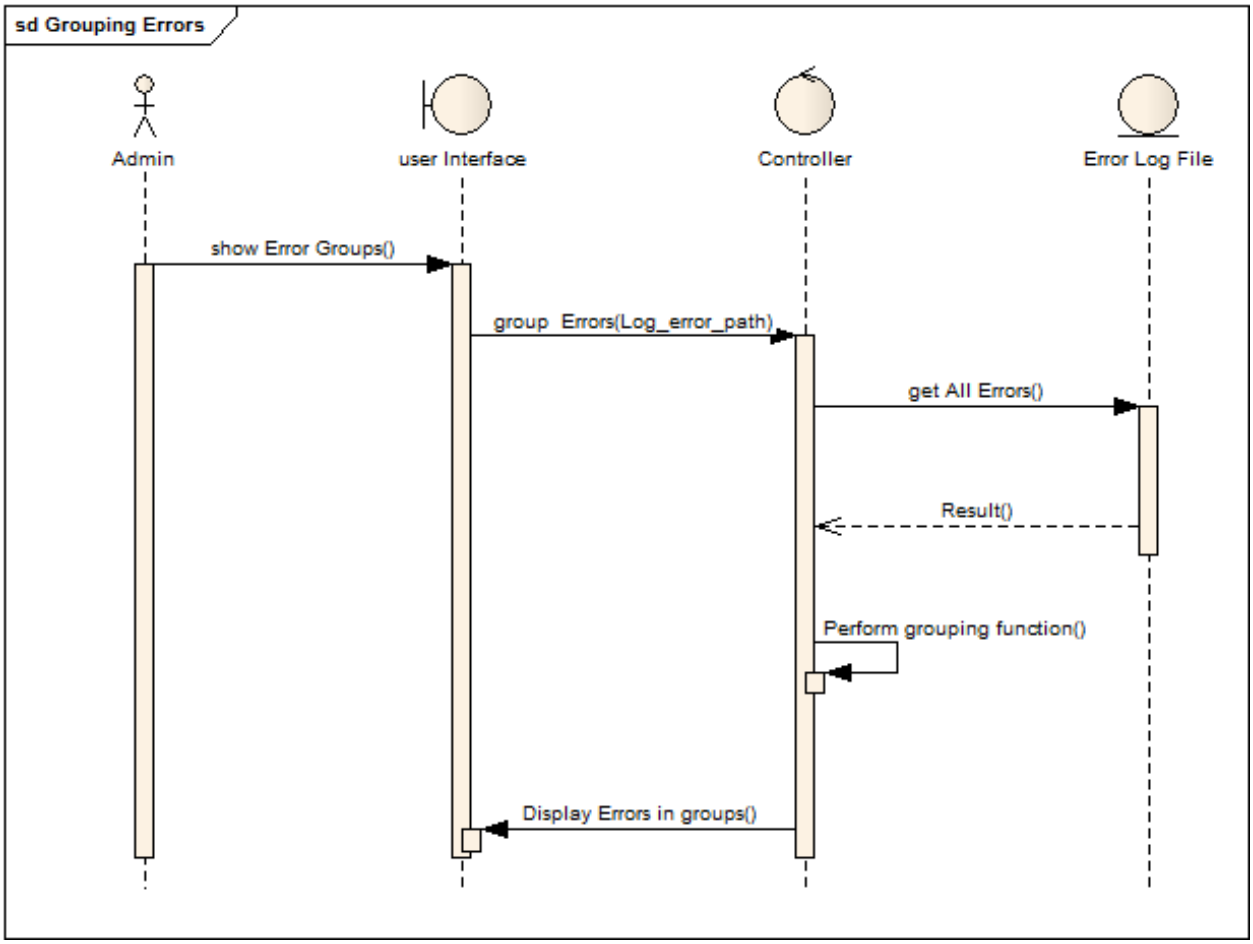


Figure 3-9Code Profiling

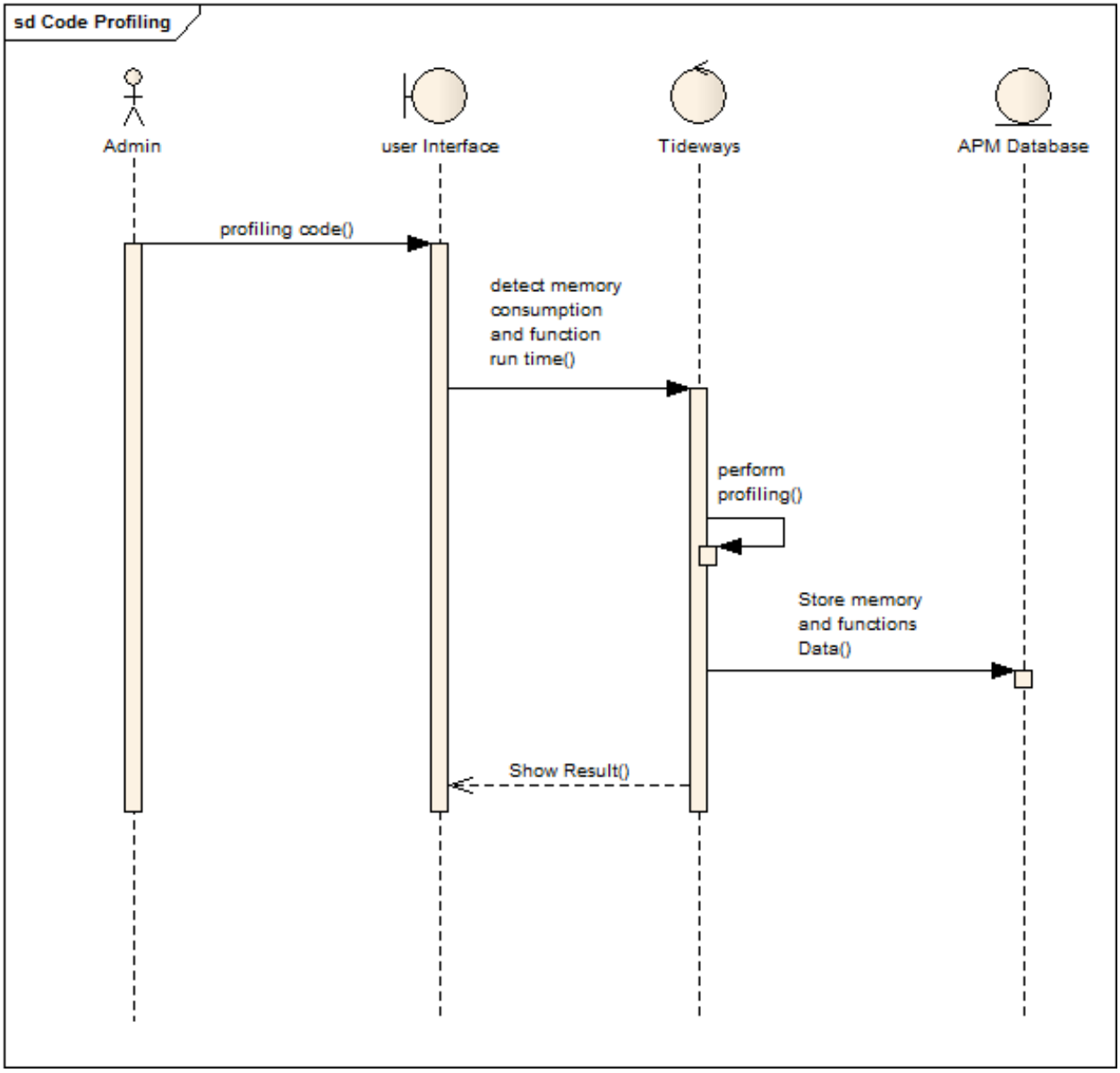
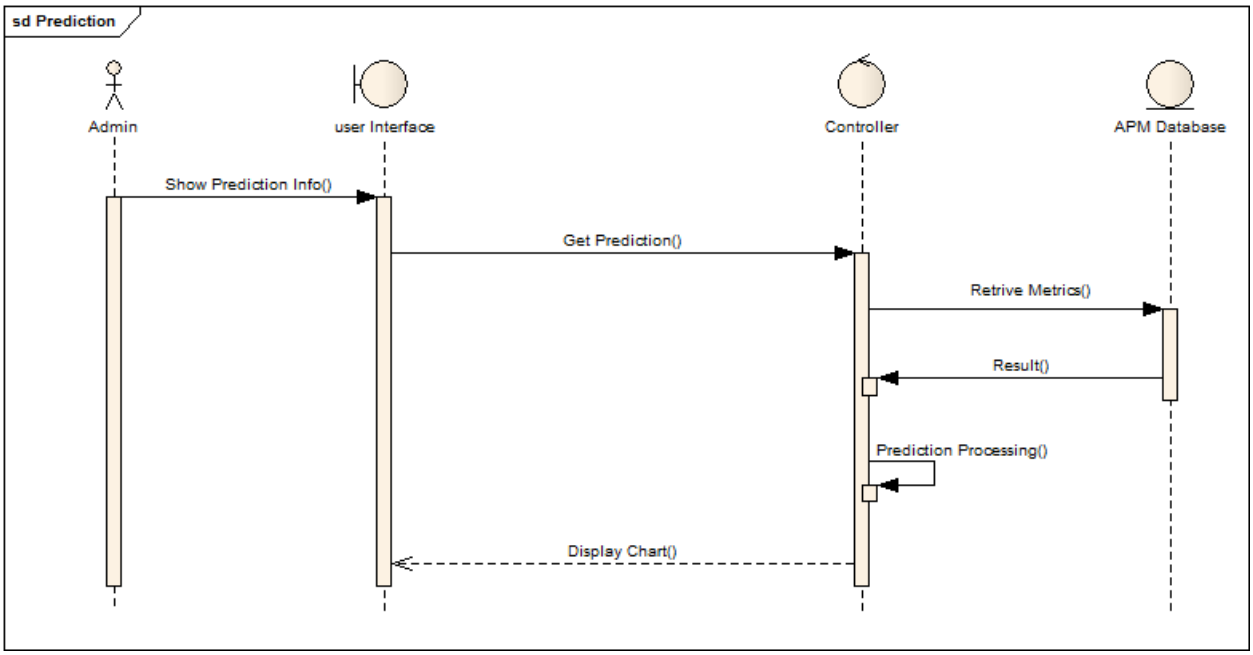


Figure 3-10 Prediction



3.5.4 Activity Diagram:

Figure 3-11 add application

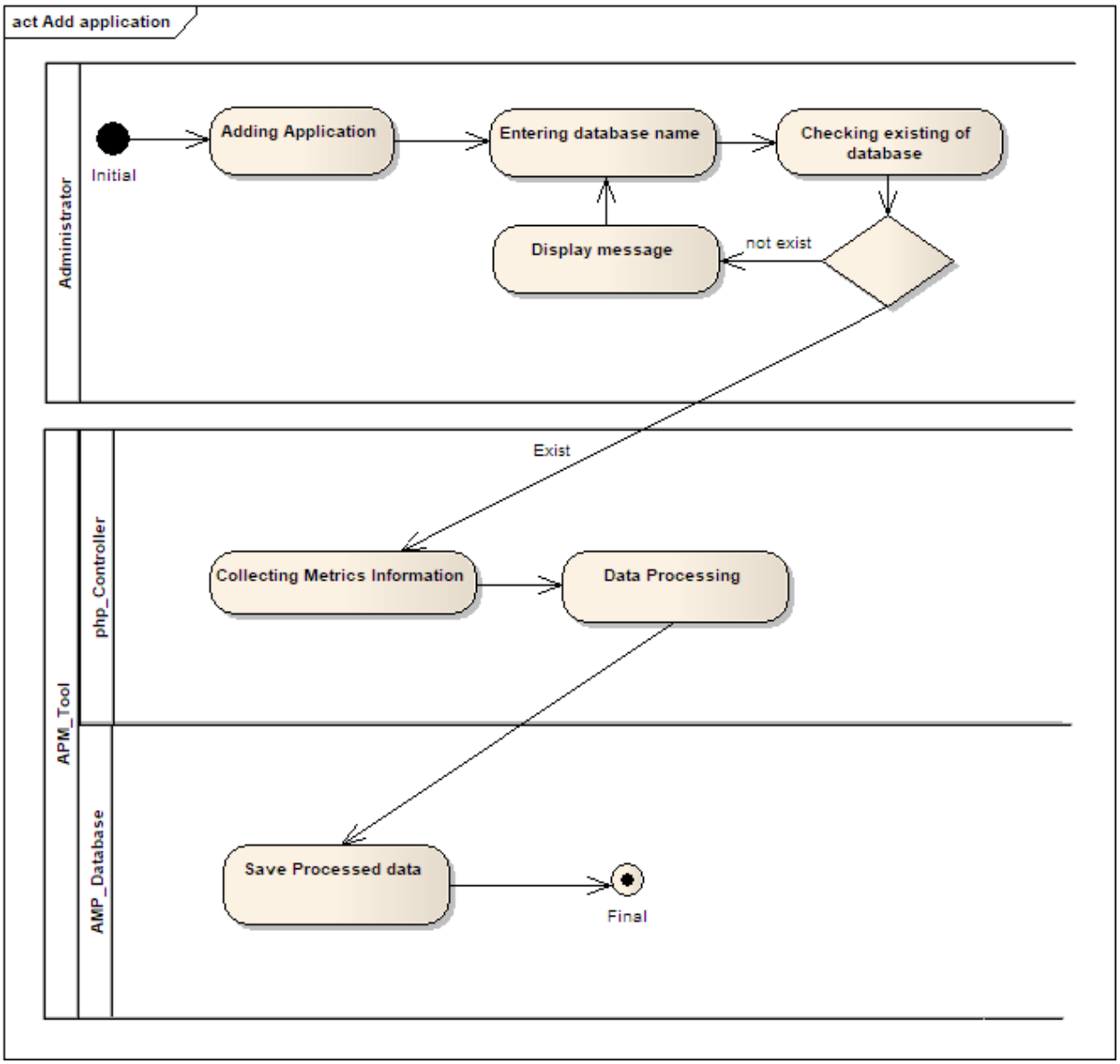


Figure 3-12 visualization

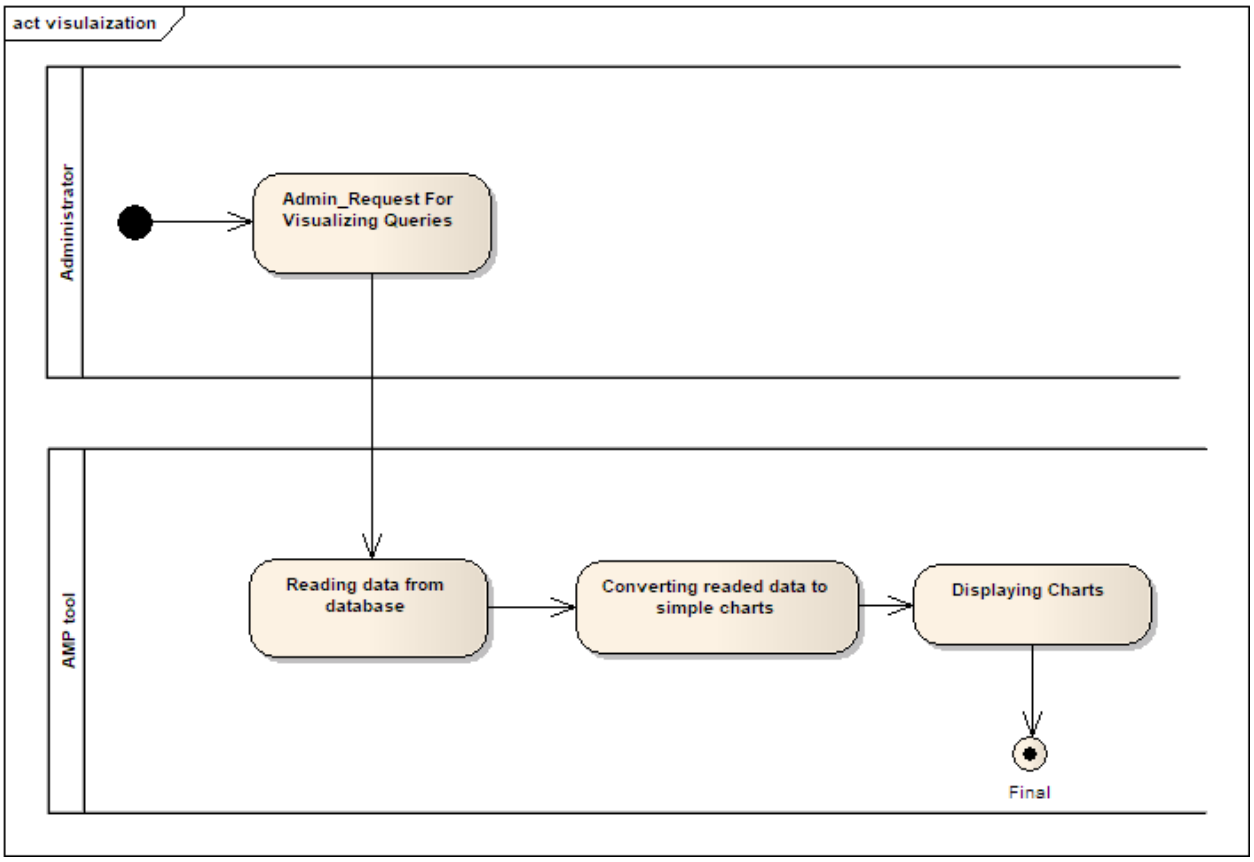


Figure 3-13 Display expensive queries

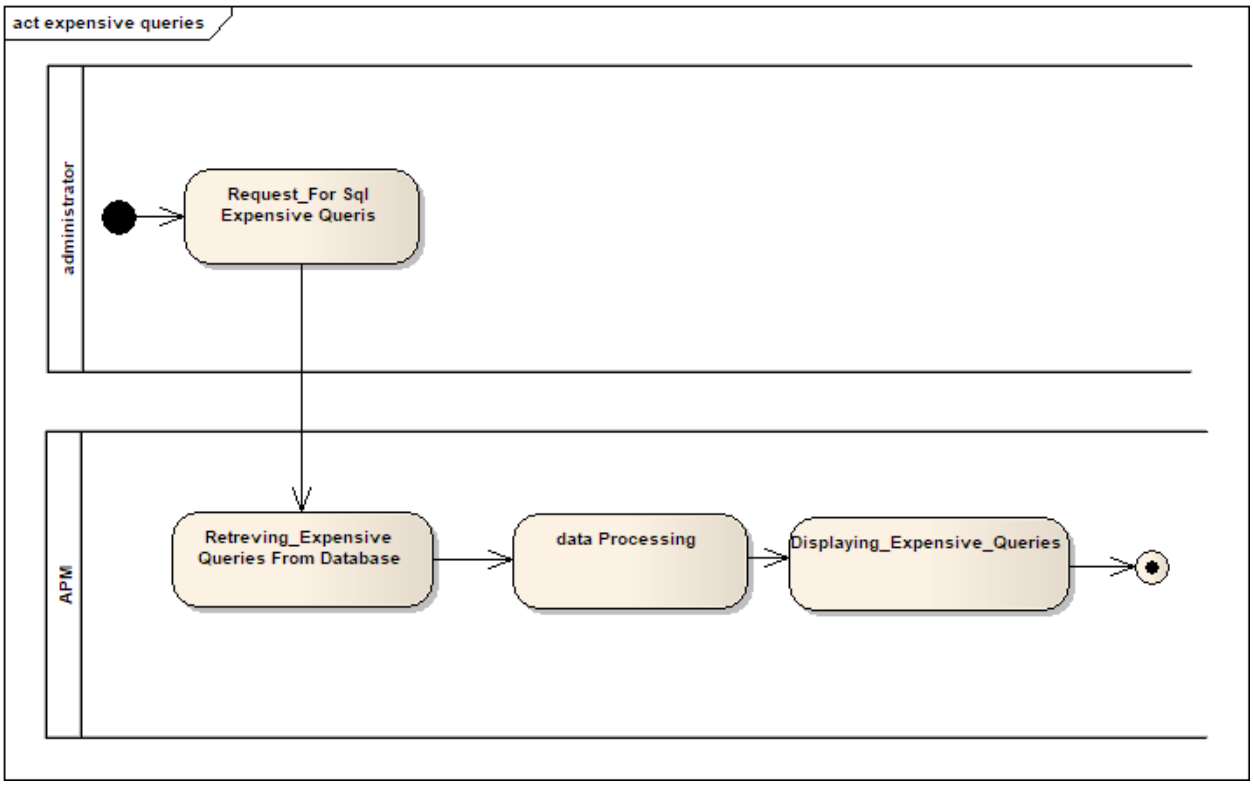


Figure 3-14 all queries activates

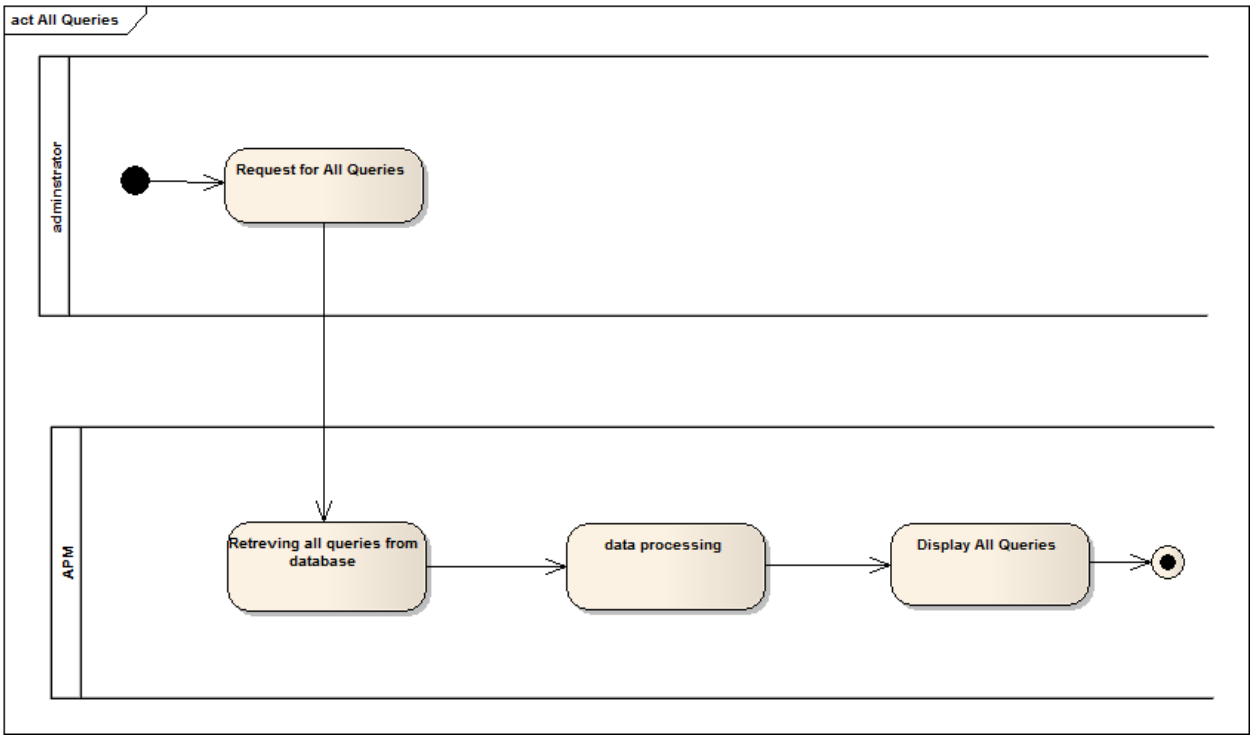


Figure 3-15 error rate

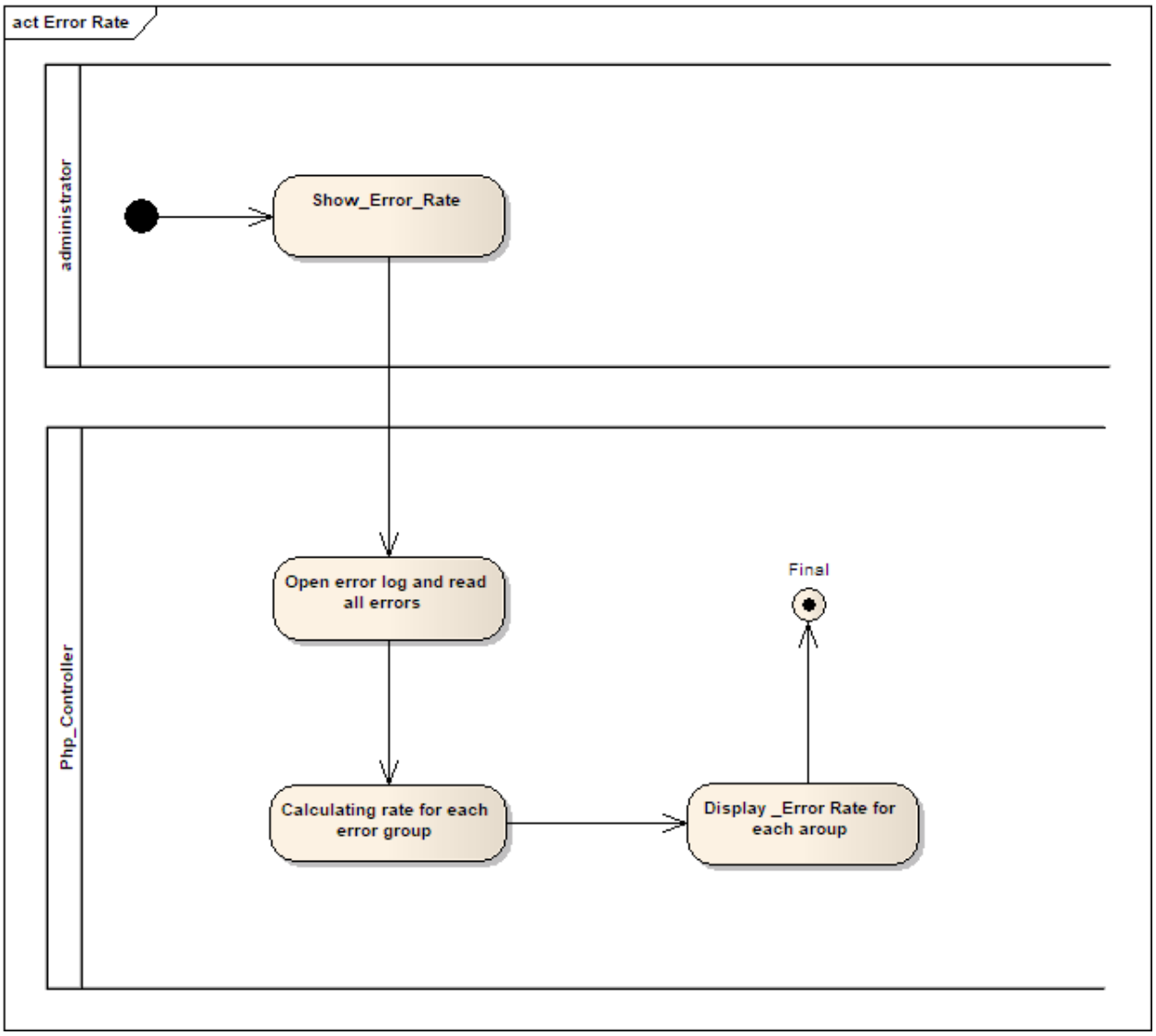


Figure 3-16 error share

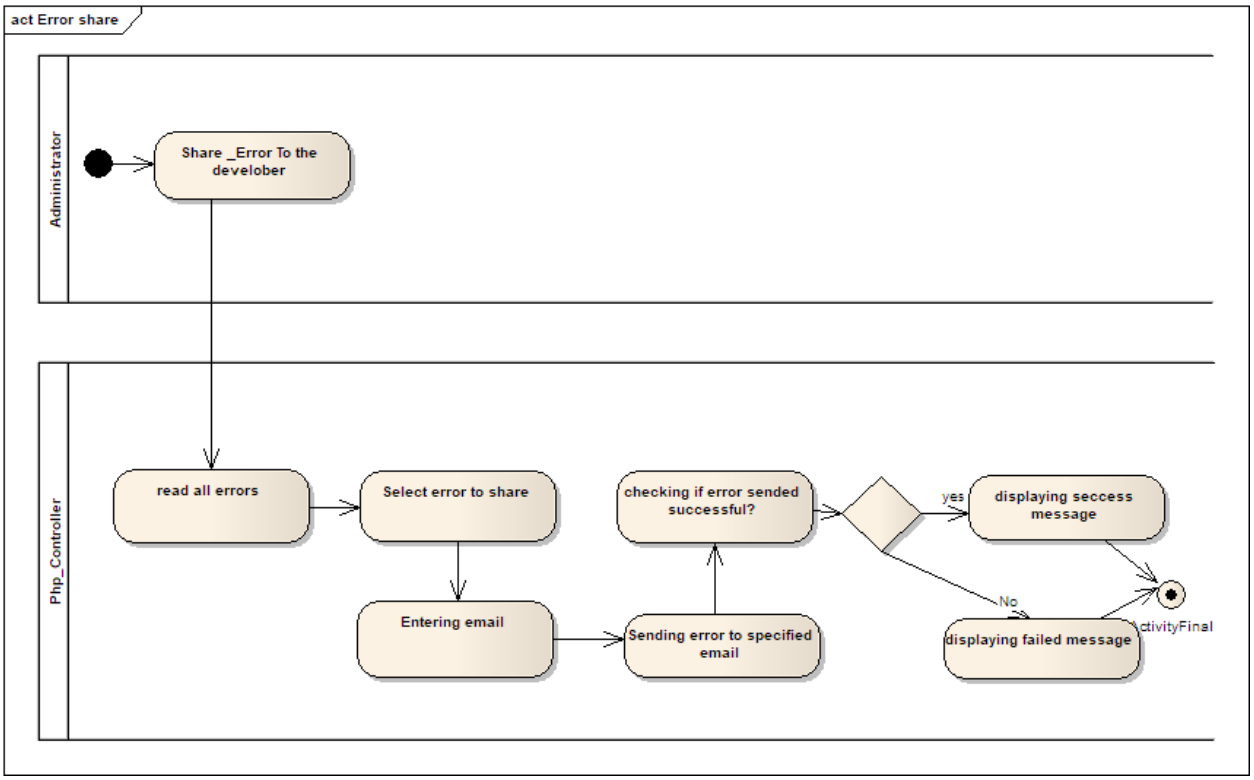


Figure 3-17 Grouping Errors activity

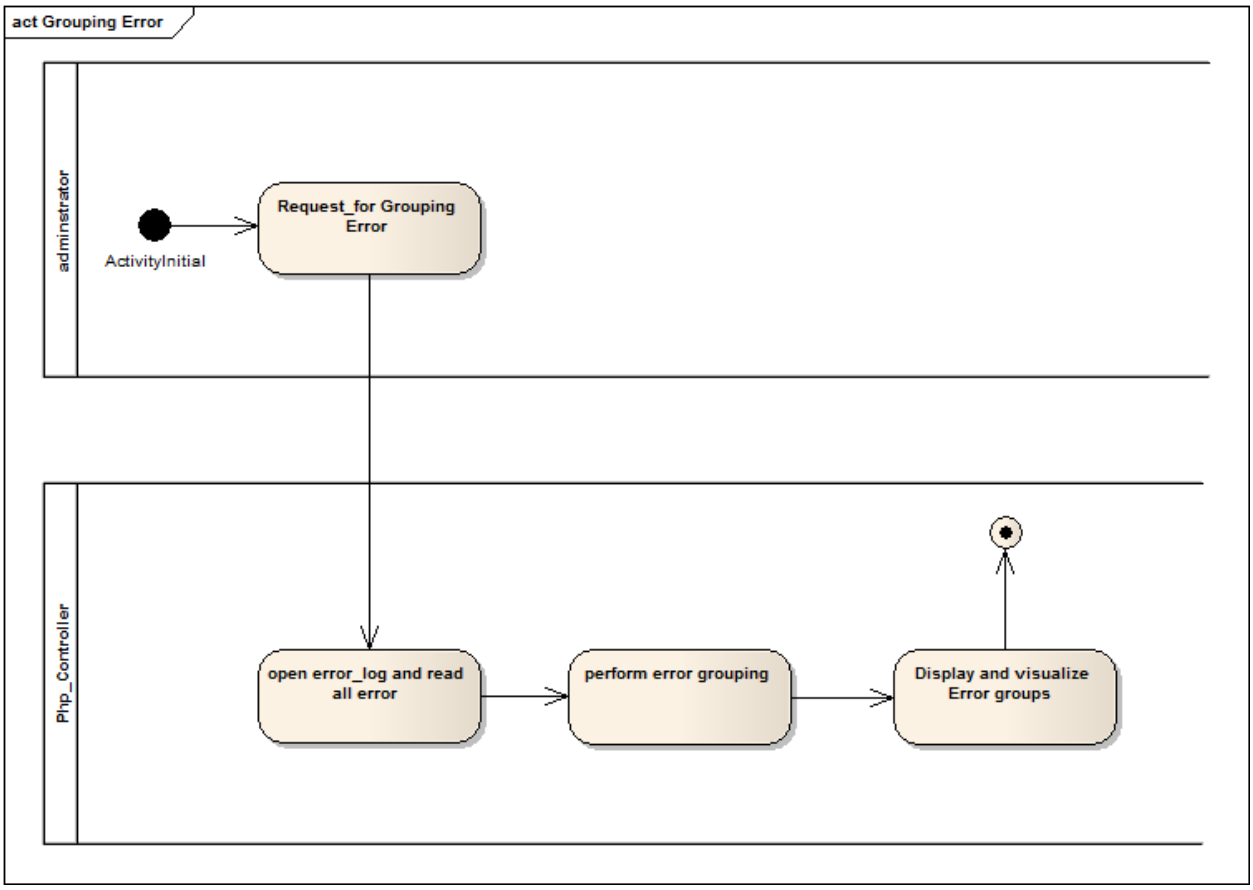


Figure 3-18 prediction

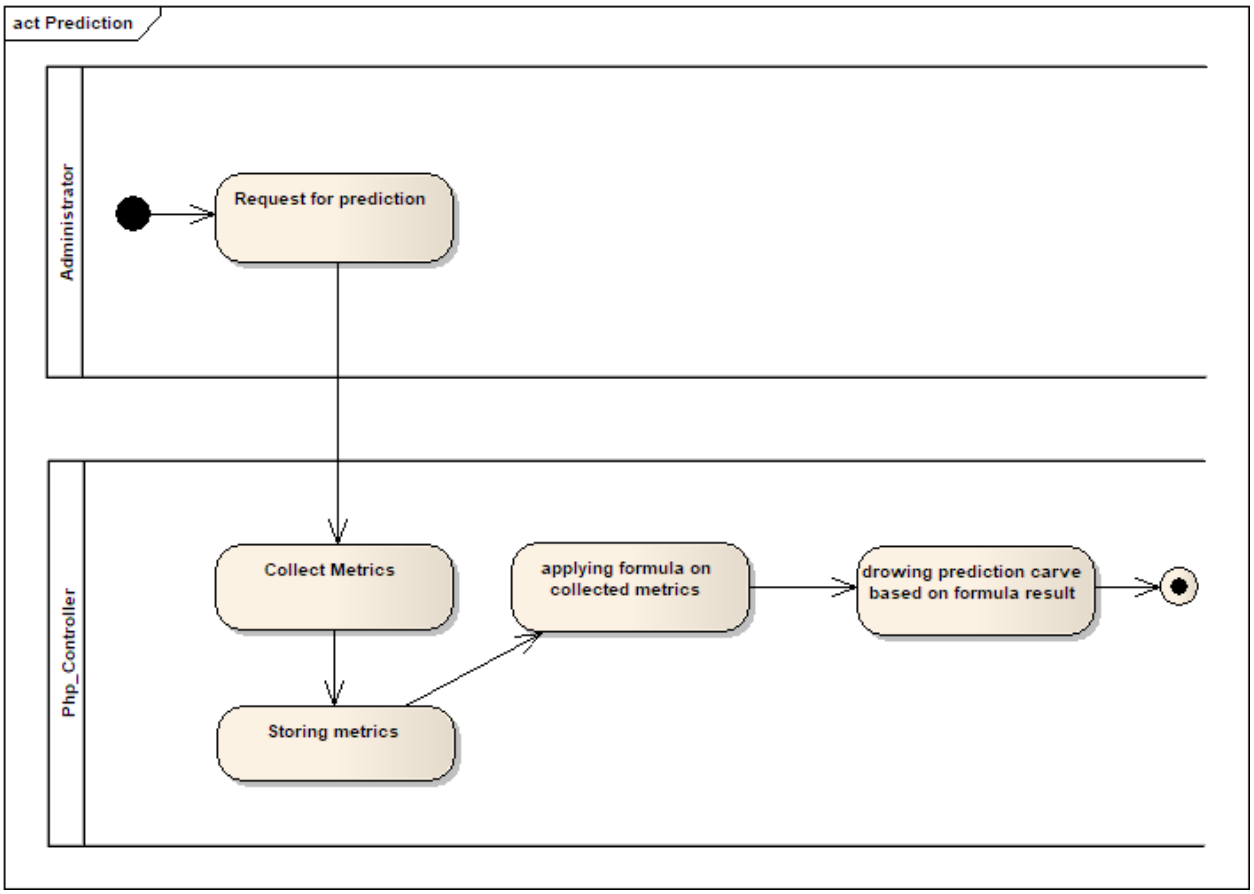
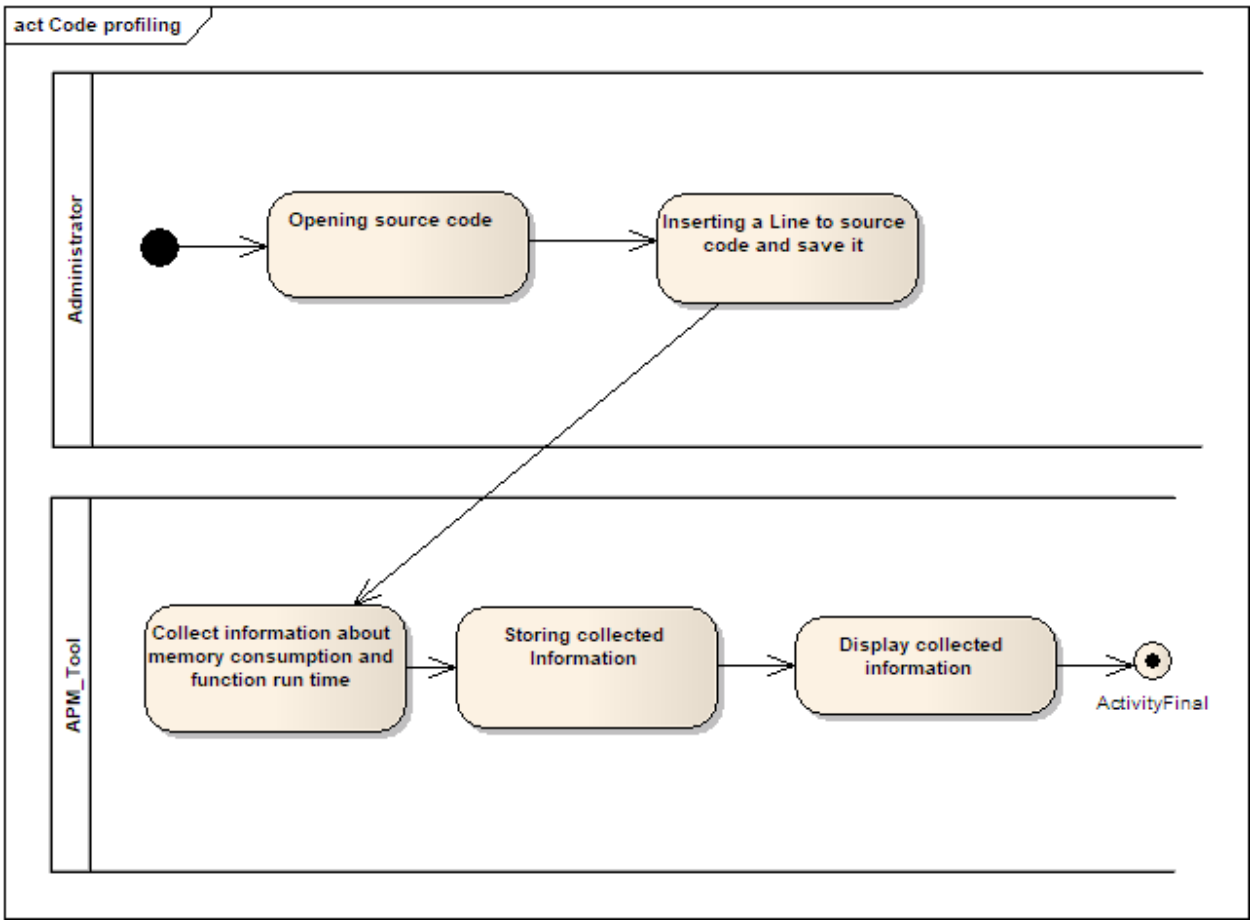


Figure 3-19 code profiling



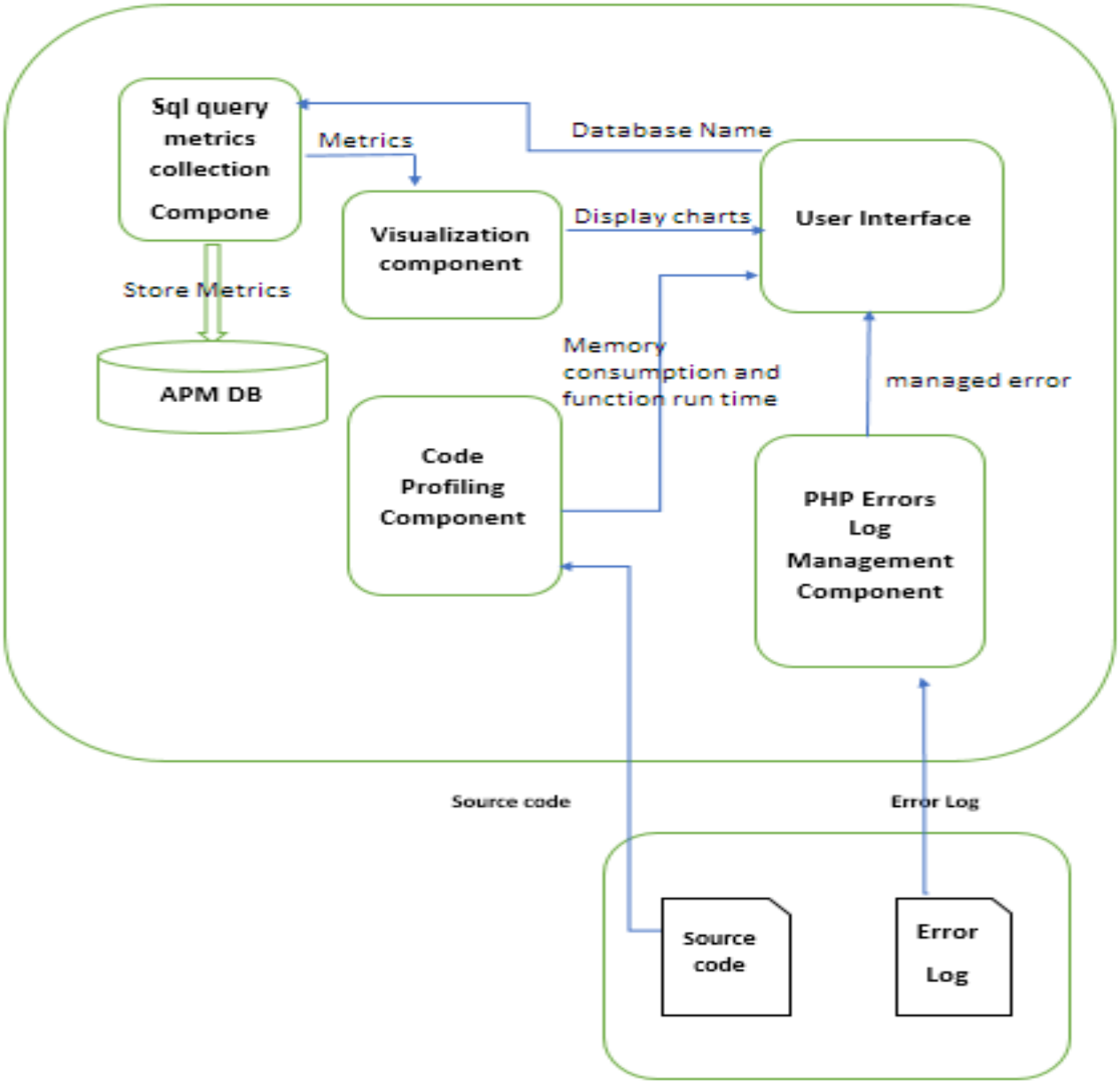
Chapter Four

System Design and Implementation

4.1 Design: -

Tool design illustrates the architecture of the tool, components, inputs and outputs for each one, as described above there are four components and now we will detailed them.

Figure 4-1 The APM Tool Architecture



Components Inputs and Outputs:

1- User interface:

Inputs:

In this module there are two inputs must be entered by the user:

1- Name of the database which the monitored application deal with.
2- Long query time to specify slow queries.
3- Path of error log.

Output:

-display all output mentioned below

2- SQL query metrics collection component:

Input:

Database name that received by user interface used by this component to collects information about every query that come from the monitored application

Output:

SQL queries information and passing this information to be stored in the database.

3- Visualization component:

Inputs:

1- Average execution time for all queries
2- Total number of queries
3- Total execution time

Output:

Displaying above data in simple charts, (for normal queries and slow queries).

4- Code Profiling Component:

Input:

Run for source code

Output:

1- CPU time in seconds for any page and function.
2- Memory usage in byte for any page and function.
3- Whole time in seconds for any page and function.

5- PHP error log management component:

Input:

The input is a Path of the error log that has been received by the user interface.

Output:

Managed error log (error information are stored in the database).

4.2 Implementation steps:

In this section we will talk about the steps that is took to implement the components of the tool.

1- SQL query metrics collection component :

To implement this component we used:

1.1- Apache http sever

Apache HTTP Server, colloquially called Apache is free and open-source cross-platform web server software, released under the terms of Apache License 2.0. [13]

Apache http sever was configured to help in collecting information about SQL queries.

1.2- MYSQL :

is an open-source relational database management system(RDBMS)[19]

We use MySQL database to store information about all query.

1.3- PHP:

PHP (Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML [15].

PHP has been used to connect with the database and send required data to be stored.

2- Code Profiling component:

Following tool has been installed and configured to complete the function of code profiling

Tideways:

This PHP extension is a complete, modernized open-source rewrite of the original XHProf extension, with a new core data structure and specifically optimized for PHP 7. The result is an XHProf data-format compatible extension with a much reduced overhead in the critical path that you are profiling.[ref]

3- Visualization component:

A tool has been used to complete this component and it is:

charts4php:

Charts 4 PHP API allows you to quickly retrieve data from your databases, and visualize your data through variety of Charts types.[18]

We use charts4php to create charts of [prediction curve, all query and slow query].

4- PHP error log management component:

A code has been written in PHP language to read the error log periodically and filtering it.

5- User Interface:

Flowing tools used to implement the user interface:

1- HTML5:

We use HTML5 for strutting and describe shape of web page.

HTML5 advantages: [16]

- Cleaner markup/ Improved code
- Elegant forms
- Consistency
- Supports rich media elements
- Offline Application Cache

2- JavaScript:

We use JavaScript programing language to make website more interactive.

3- jQuery:

JQuery is JavaScript library used to simplify some task .

4- CSS (Cascade Style Sheet):

We use this stylesheet language to describe style of website.

CHAPTER FIVE

Testing

5.1 Introduction

This chapter shows the case study steps of this software and some of the testing that has been done.

5.2 Case Study:

5.2.1 Initial setup

This is the login screen which is used to login to software

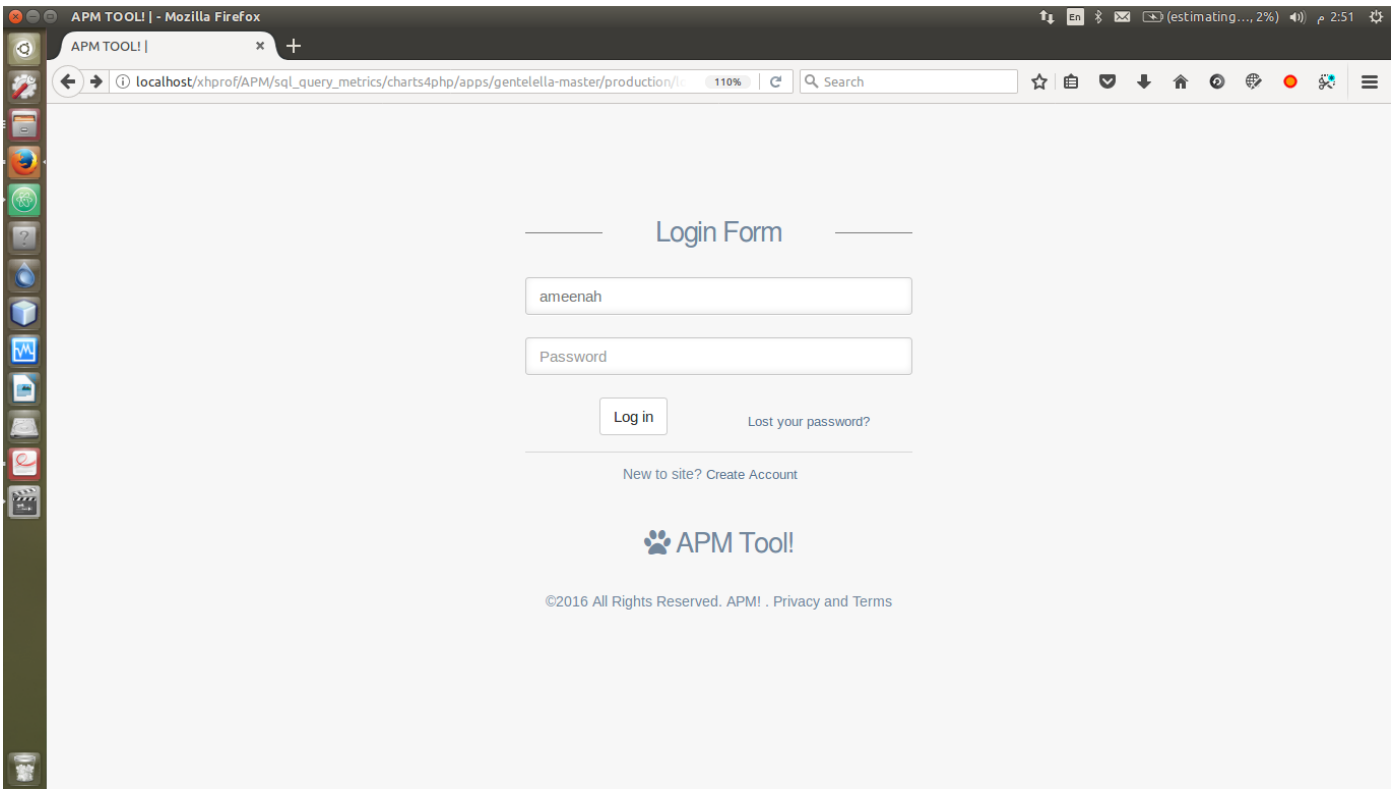


Figure 5-1 Login Screen

The sign up Page where the users registers his account

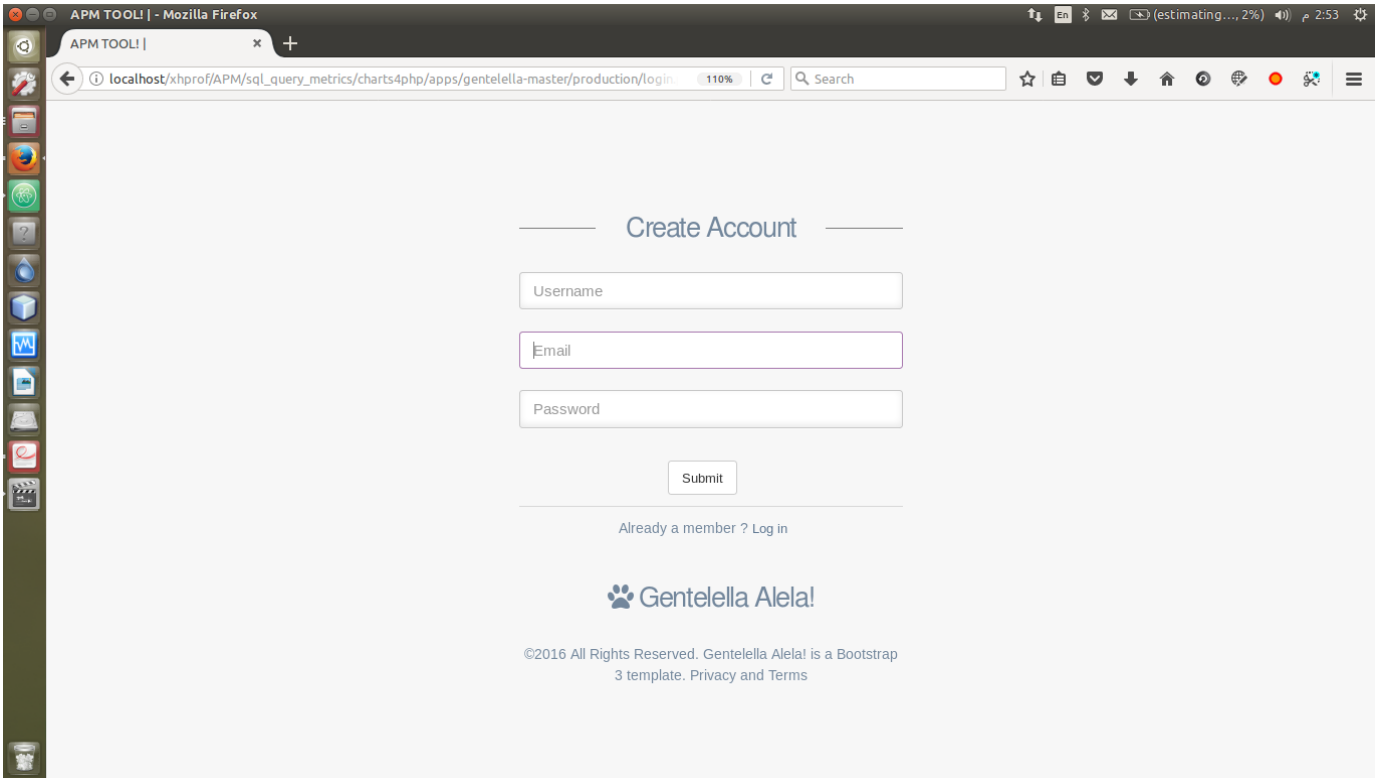


Figure 5-2 sign up Screen

After sign up, setting page will show up. User must enter the application database name, the most acceptable query duration, and the error log path in the server

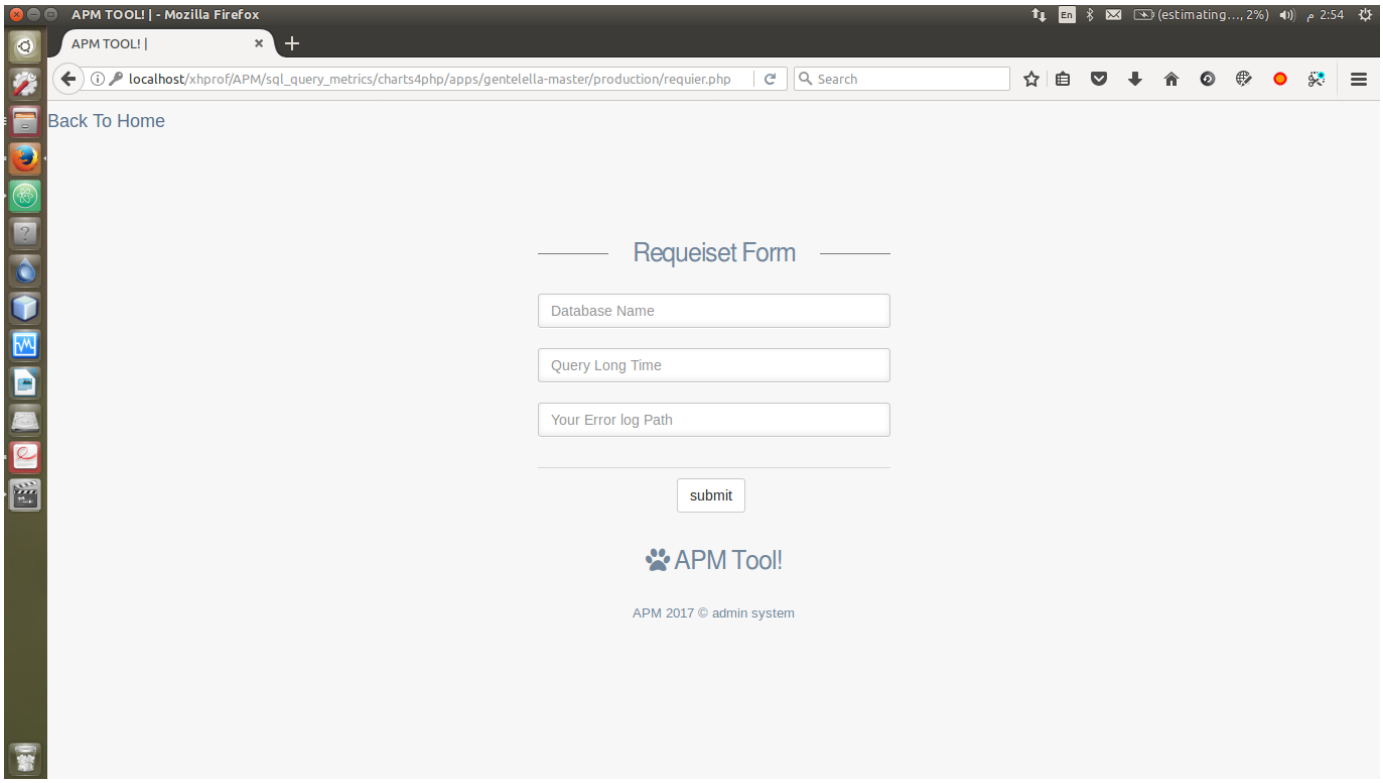


Figure 5-3Setting screen

After saving the setting, user will headed to the home page where on the side bar can go to the different section of the project

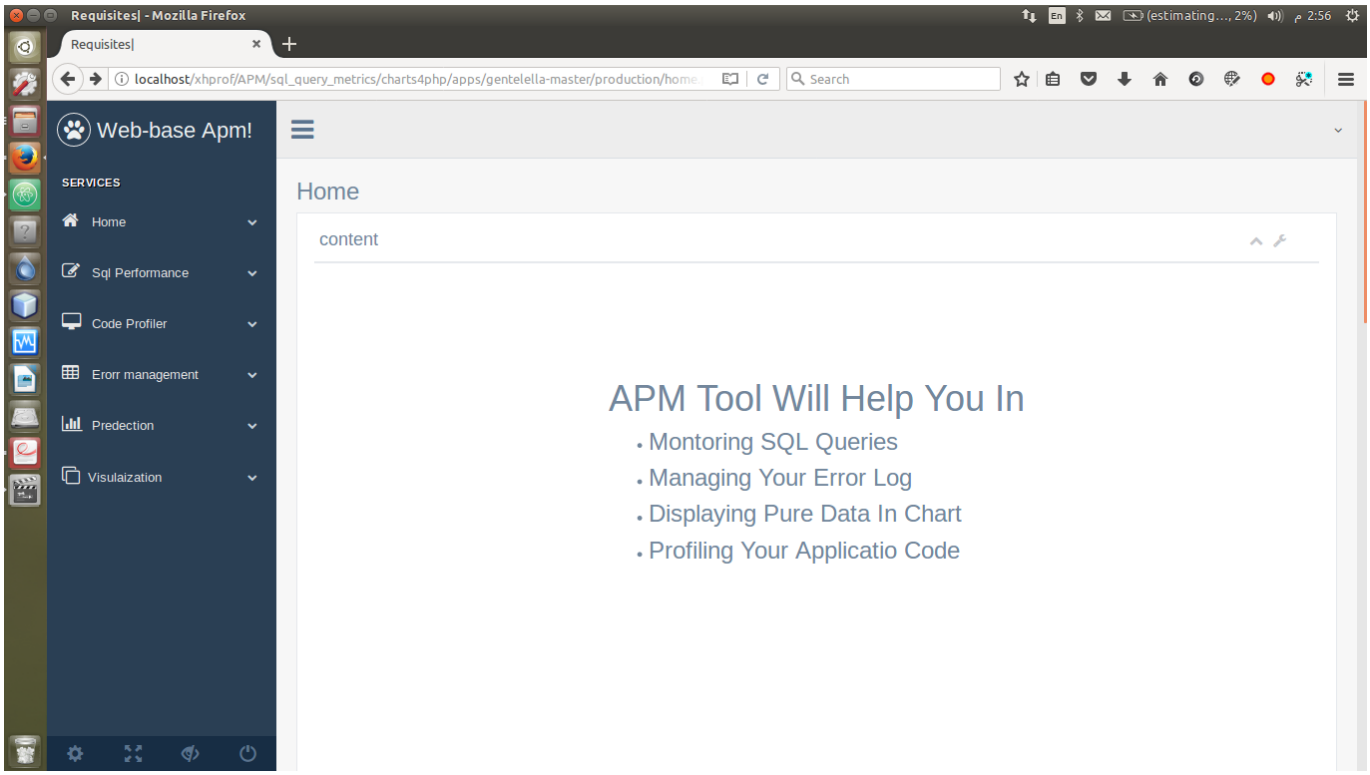


Figure 5-4Home Screen

5.2.2 SQL performance section:

- **Show normal application queries information**

when the administrator request for queries the tool Will display all normal queries and full information about it, also calculate and display the total number of all queries, average execution time and total execution time.

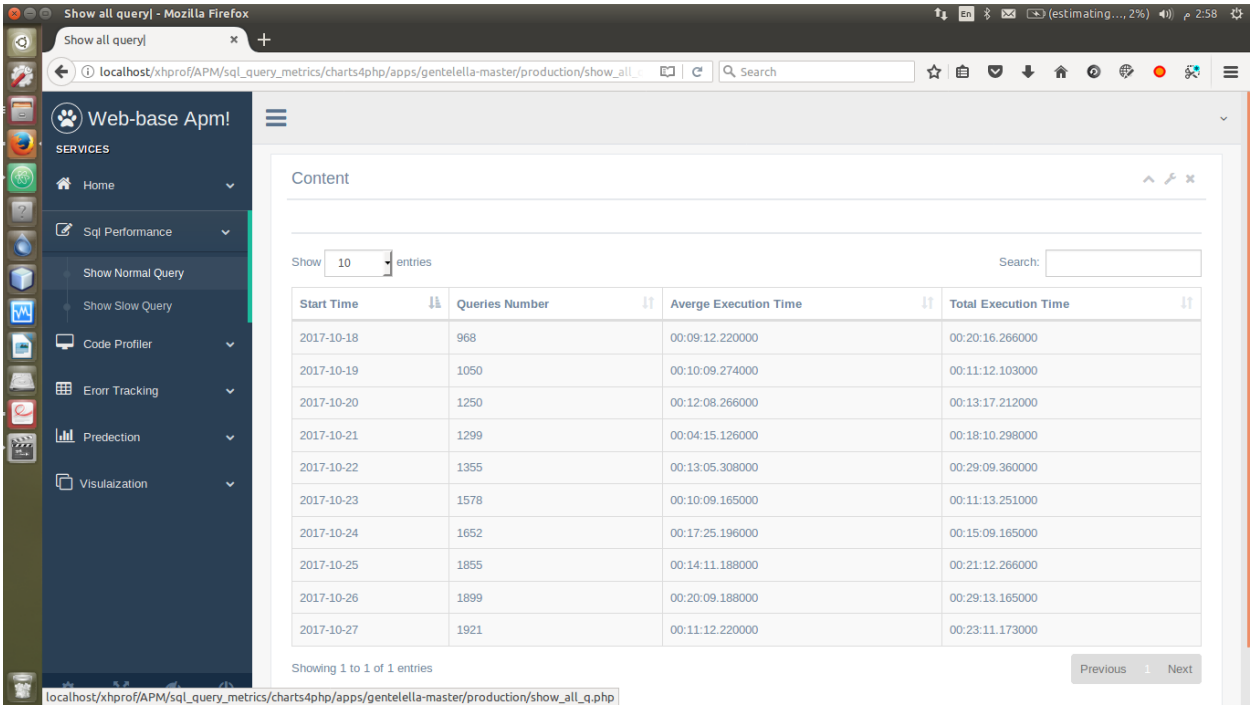


Figure 5-5 Normal Queries Screen

- **Show slow queries**

When the administrator request for showing expensive queries the tool will retrieve them based on specific value of long time to took that specified by him.

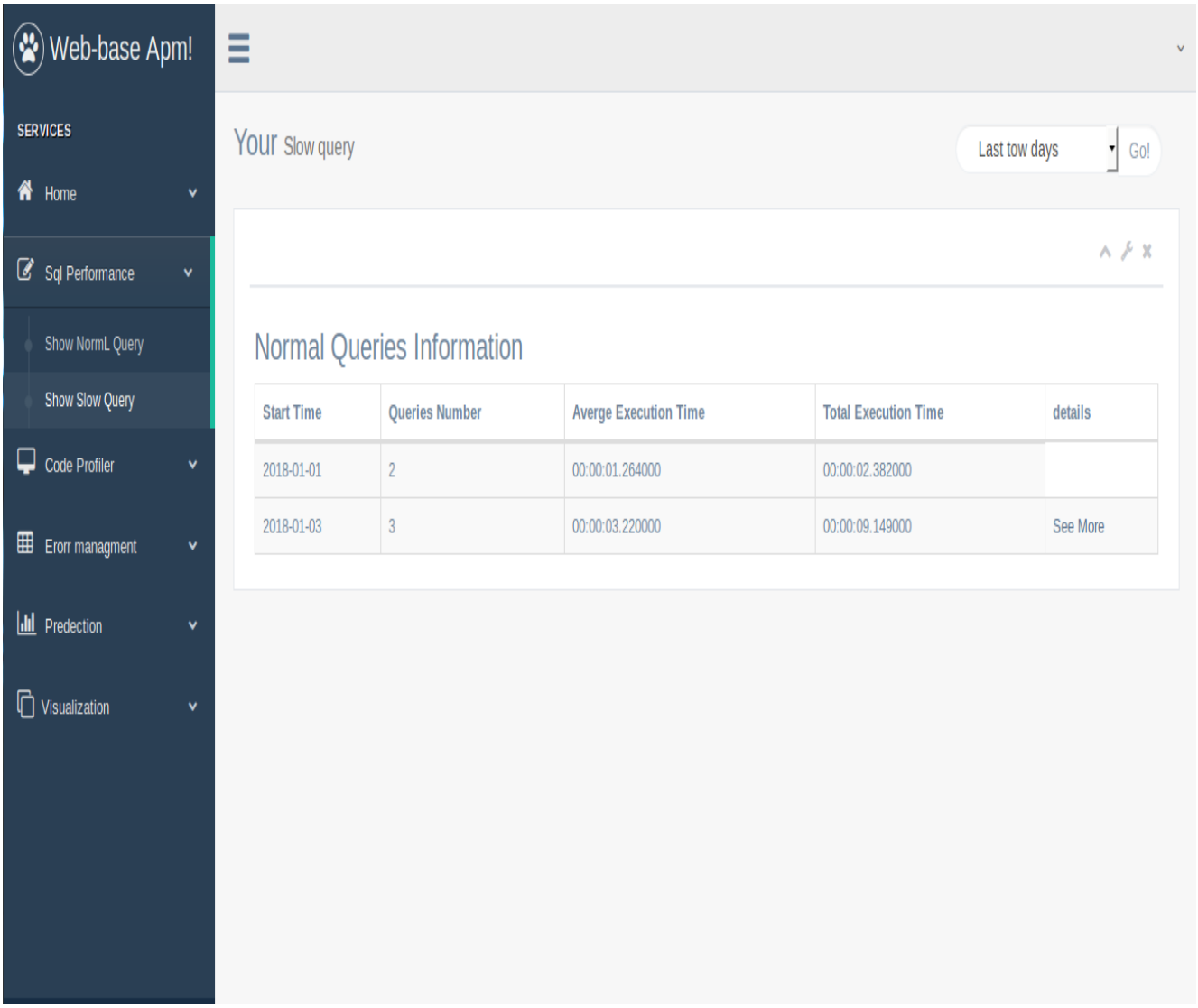


Figure 5-6 Slow Queries Screen

Tool will retrieve details about queries when requested by the user.

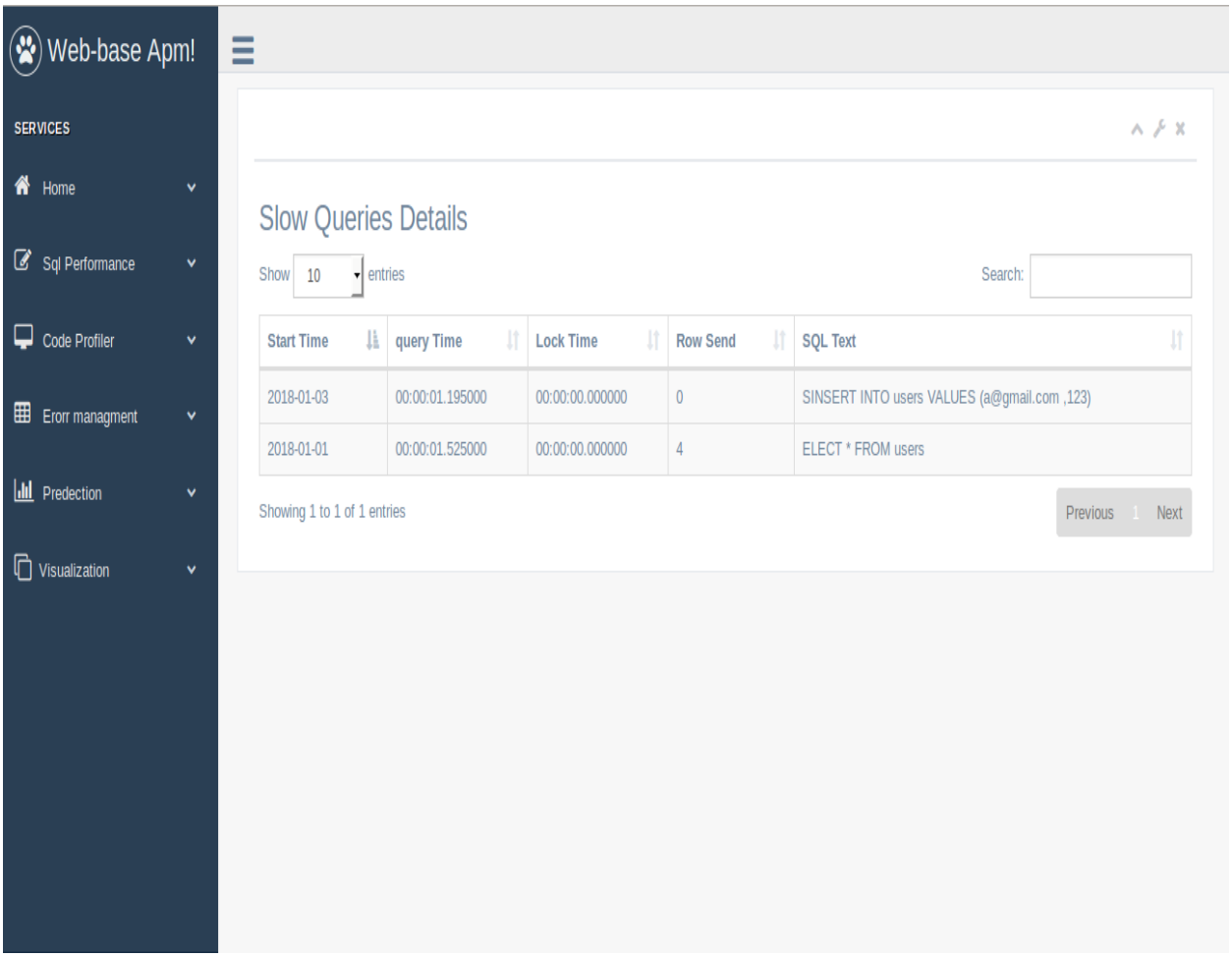


Figure 5-7 Slow Queries details Screen

5.2.3 Code Profiler section:

On the code profiler section, PHP pages profiling run metrics will show up, each row represent a run which include the date and time of execution , the CPU time, the wall time, peak memory usage, the URL, and the simplified URL.

Runs can be sorted by the hardest hit, the most expensive (in CPU term), the most ram usage, and the longest running.

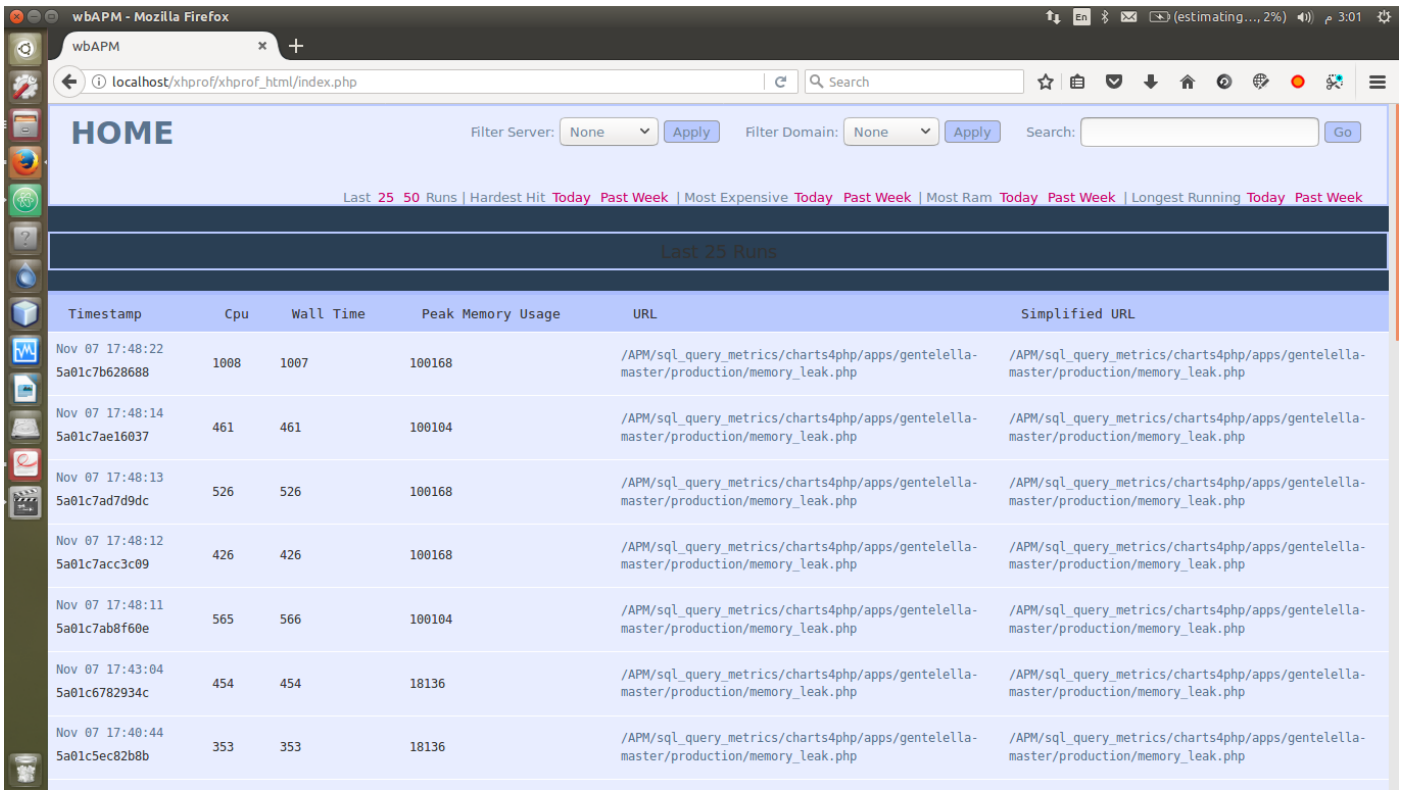


Figure 5-8 Code Profiler Screen

Runs metrics can be shown in statistical chart to facilitate detecting page runs that have issues.

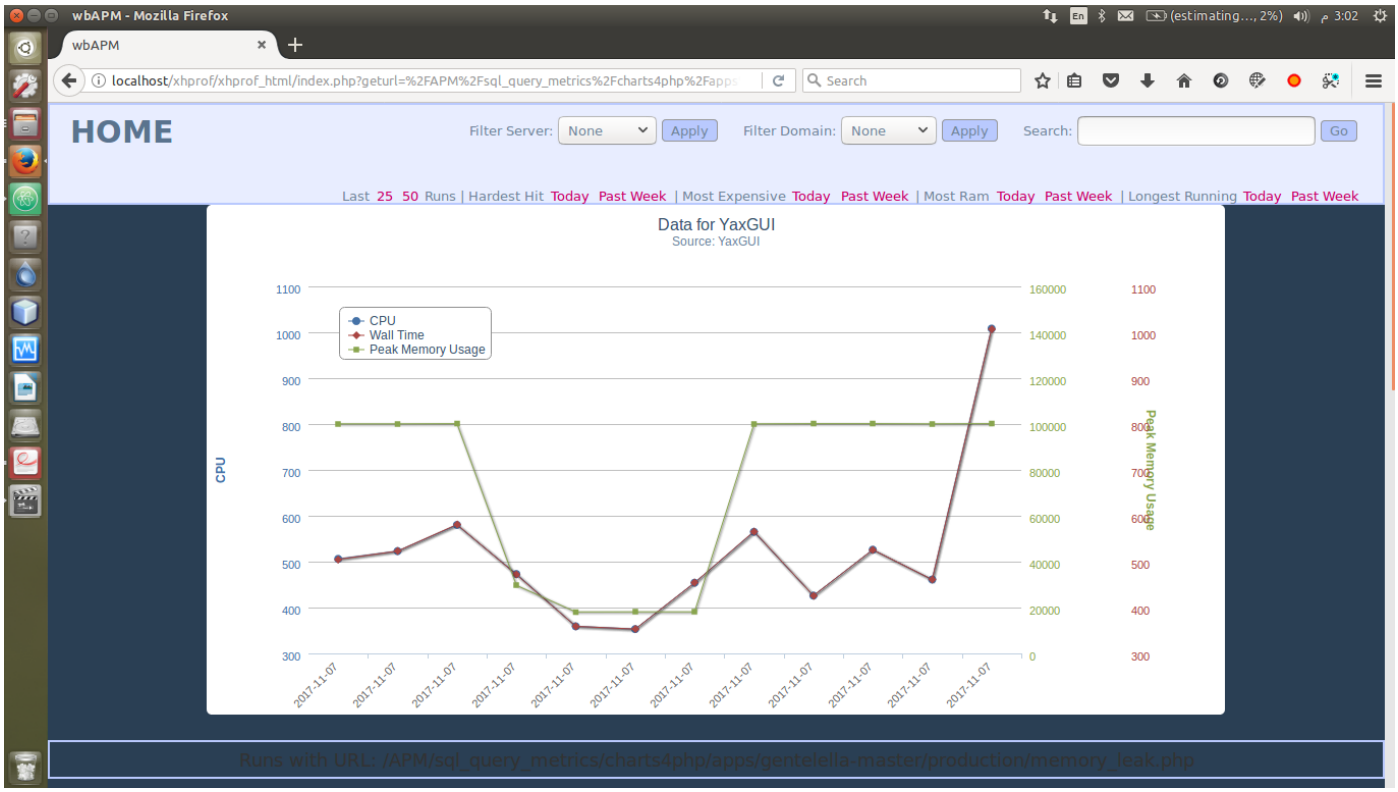


Figure 5-9 Profiling Chart Screen

5.2.4 Error Manager Section:

Error manger is an approach to replace the error log, you can search specific errors by entering date or a string to look up

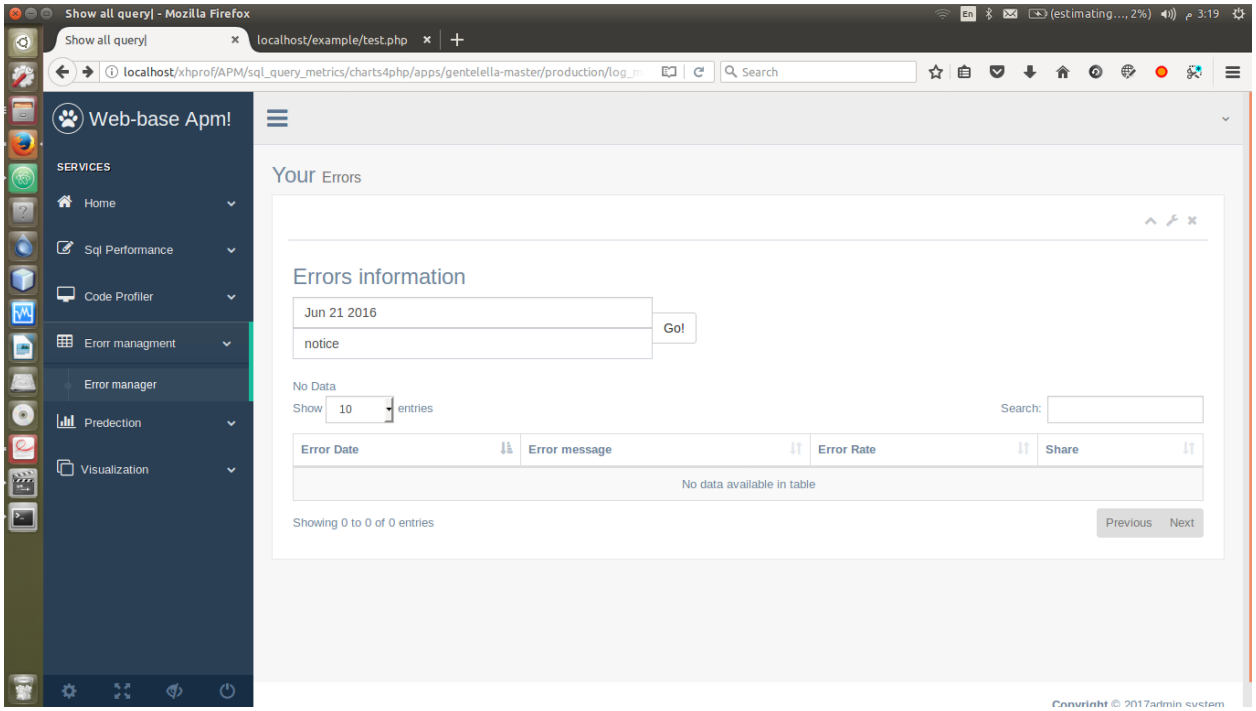


Figure 5-10 Search Error

Search result will show up, each row represent an error, which include the date, the error message, error rate.

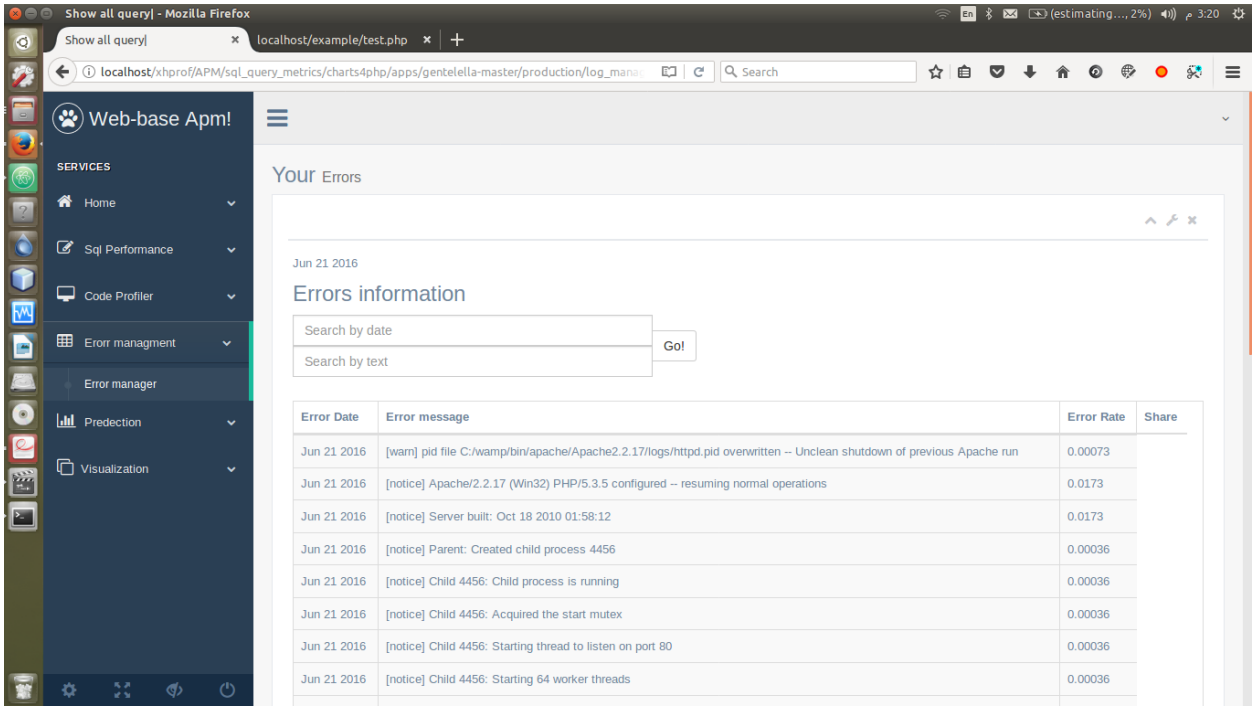


Figure 5-11 Search Result screen

Here the user can share the error by the share button, the error manager connected th thunderbird (mailing agent) so he can send it by his email.

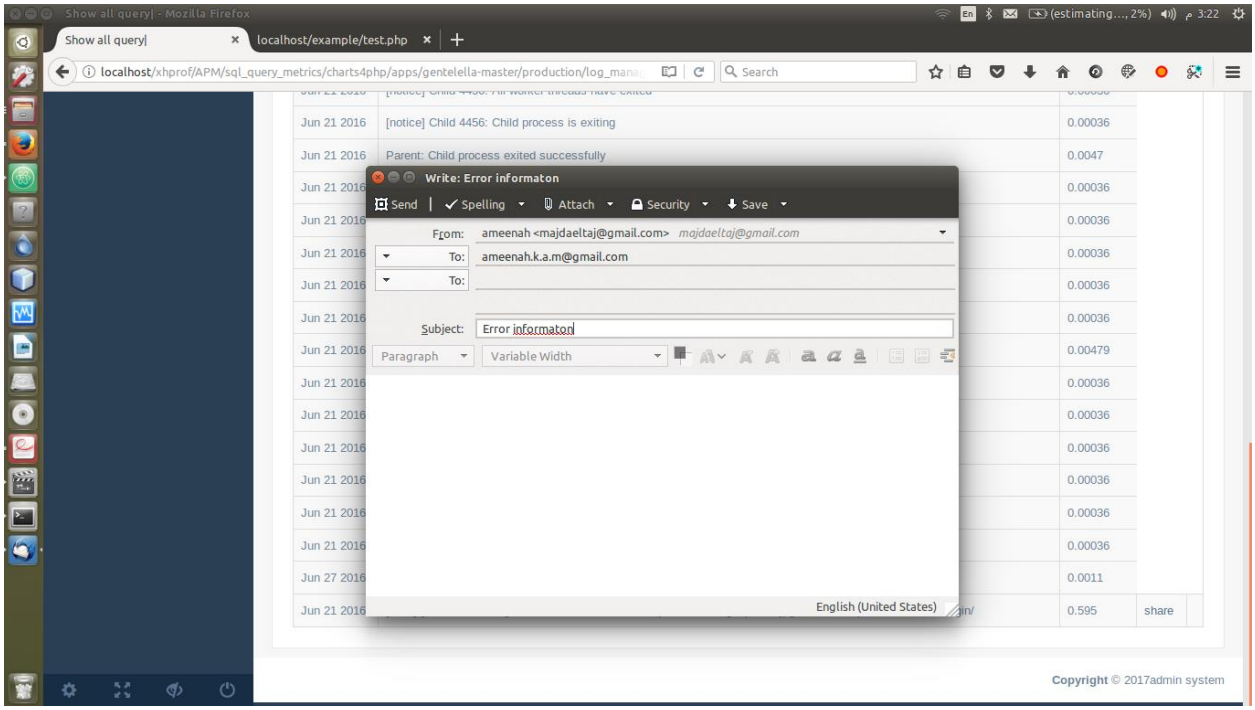


Figure 5-12 Share Error Screen

5.2.5 Visualization Section:

- **Visualizing normal queries:**

The output is a chart presenting the number of normal queries in different periods (months).

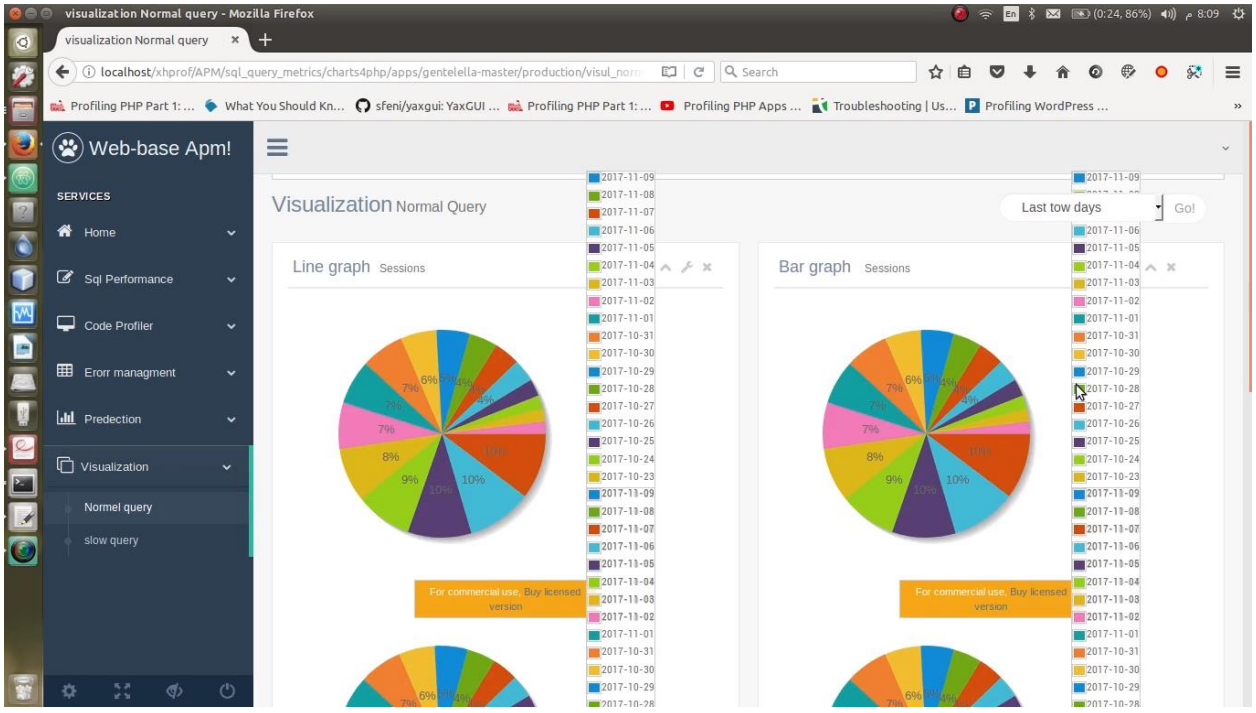


Figure 5-13 Visualize Normal Queries

- **Visualizing slow queries**

The output is a chart presenting the number of expensive queries in different periods (months).

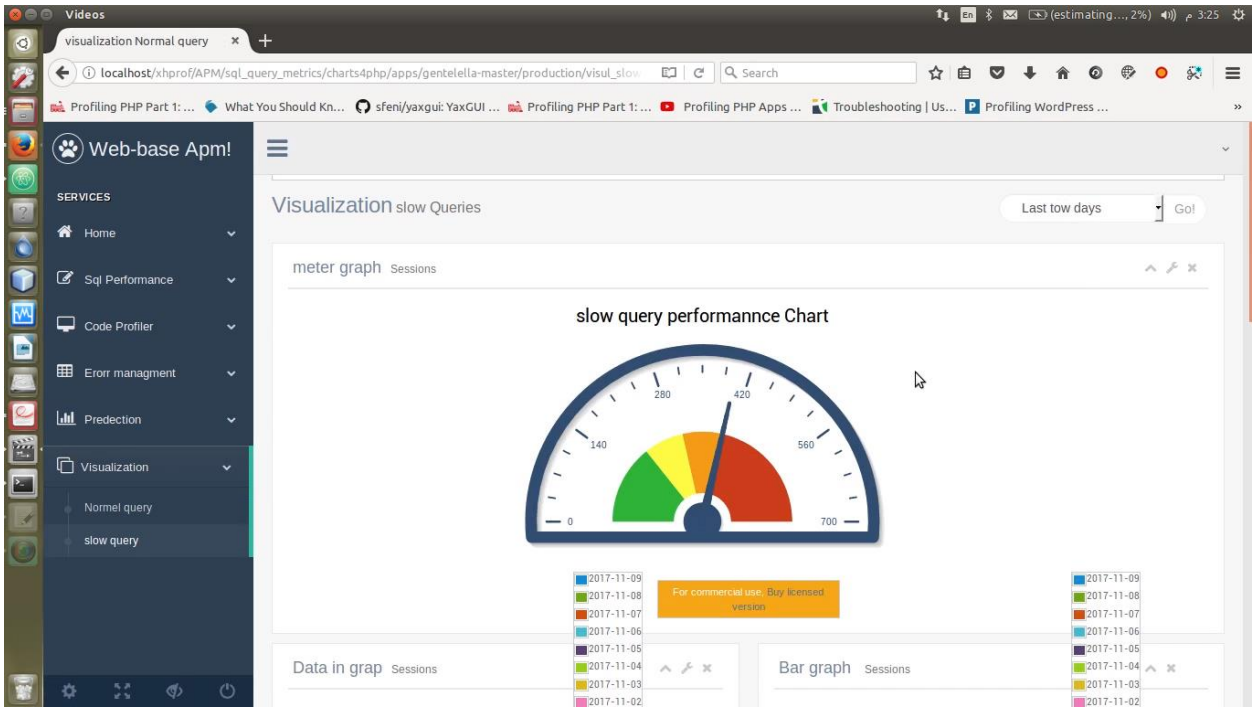


Figure 5-14 Visualize Slow Queries

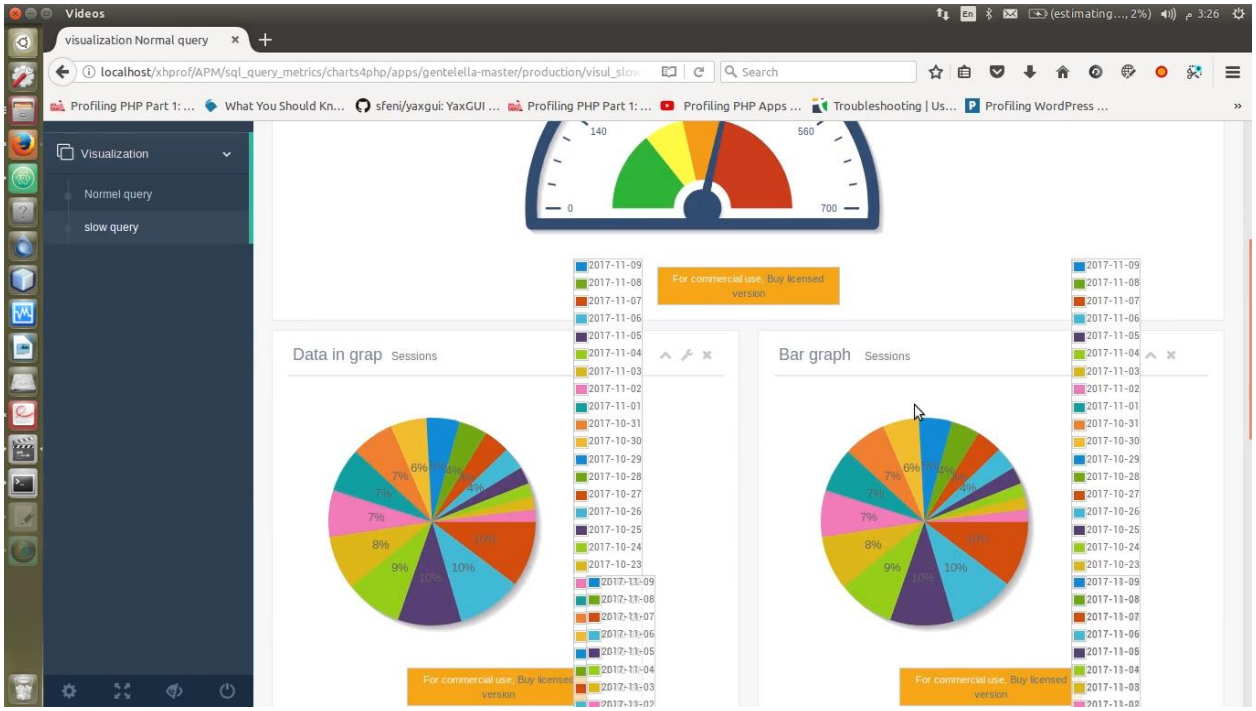


Figure 5-15 Visualize Slow Queries pt2

Chapter Six

RESULTS &RECOMMENDATION

6.1 Introduction:

This chapter discusses the result of using application performance monitoring, the recommendation for functionality extension, and the conclusion.

6.2 Result:

- Understanding system behavior.
- Enhancement of problem detection.
- Forecasting of performance behavior.
- Availability of informing about problems.

6.3 Obstacles:

- Lack of academic literature papers.
- Difficulty in tools configuration
- Gaining metrics approaches.

6.4 Recommendation:

- Apply advanced techniques in performance prediction.
- Heterogeneous languages and database support.
- Automating of fixing problems.
- Network monitoring.

6.5 Conclusion:

This tool doesn’t rise up to the level of the other available tools such as NewRelic and Daynatrce, because these tools developed by expert developers, but we as developers did our best to provide all the features We could that is required in a tool of this scale and surly there has been a lot of effort to correctly complete and enhance this project.

By using this tool troubleshooting of application performance problem become easier, faster and more efficient. The tool provide information about SQL queries performance and displaying the information in chart, monitoring the application source code and also managing the error log file.

References

[1] Lakshmi S. Iyer, Babita Gupta, Nakul Johri, "Performance, scalability and reliability issues in web applications" Greensboro, North Carolina, USA, Industrial Management & Data systems, Vol. 105 lss 5 pp 561 – 576

[2] Mandar Sahasrabudhe, Meenakshi Panwar, Sagar chaudhari, "Application Performance Monitoring and Prediction" Pune, Maharashtra, India,

[3] Tarek M. Ahmed1, Cor-Paul Bezemer1, Tse-Hsun Chen1, Ahmed E. Hassan1, Weiyi Shang2 Software Analysis and Intelligence Lab (SAIL), Queen's University, Kingston, Ontario, Canada, "Studying the Effectiveness of Application Performance Management (APM) Tools for Detecting Performance Regressions for Web Applications: An Experience Report"

[4] the pros and cons of using NewRilic [online]. Avilable: https://www.experts-exchange.com/articles/16459/Pros-and-Cons-to-using-New-Relic-for-performance-monitoring.html. [Accessed 9 2017].

[5]SAAS application monitoring, [online]. Avilable: http://www.tomsitpro.com/articles/saas-application-monitoring-tools,2-711.html [Accessed 9 2017].

[6] "An introduction to application performance monitoring" "Appdynamic white paper"

[7] The anatomy of APM: Four foundational elements to a successful strategy [online]. Avilable: http://www.apmdigest.com/the-anatomy-of-apm-4-foundational-elements-to-a-successful-strategy [Accessed 9 2017].

[8] BERNHARD PLATTNER: " real time execution monitoring" IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. SE-10, NO. 6, NOVEMBER 1984

[10] Automating ITSM Incident Management Process paper

[9]"New Relic APM " [online]. Avilable: https://newrelic.com/ [Accessed 10 2017].

[10] ”Pinpoint APM ” [online]. Avilable: https://github.com/naver/pinpoint [Accessed 10 2017].

[11] ”AppDynamic APM ” [online]. Avilable: https://www.appdynamics.com/ [Accessed 10 2017].

[12] ”Datadog APM ” [online]. Avilable: https://www.datadoghq.com/apm/ [Accessed 10 2017].

[13] “(Apache HTTP Server) Wikipedia “[online]. Avilable: https://en.wikipedia.org/wiki/Apache_HTTP_Server [Accessed 10 2017].

[14] “(Unified Modeling Language) Wikipedia “[online]. Avilable: https://en.wikipedia.org/wiki/Unified_Modeling_Language [Accessed 10 2017].

[15]”PHP manual ” [online]. Avilable: http://php.net/manual/en/intro-whatis.php [Accessed 10 2017].

[16]https://gotechark.com/blog/advantages-html5/ [Accessed 10 2017].

[17] “(GIT) Wikipedia “[online]. Avilable: https://en.wikipedia.org/wiki/Git [Accessed 10 2017].

[18]]”chart PHP ” [online]. Avilable: http://www.chartphp.com/ [Accessed 10 2017].

[19] “(mySQL) Wikipedia “[online]. Avilable: https://en.wikipedia.org/wiki/MySQL [Accessed 10 2017].

Appendix

Appendix (A)

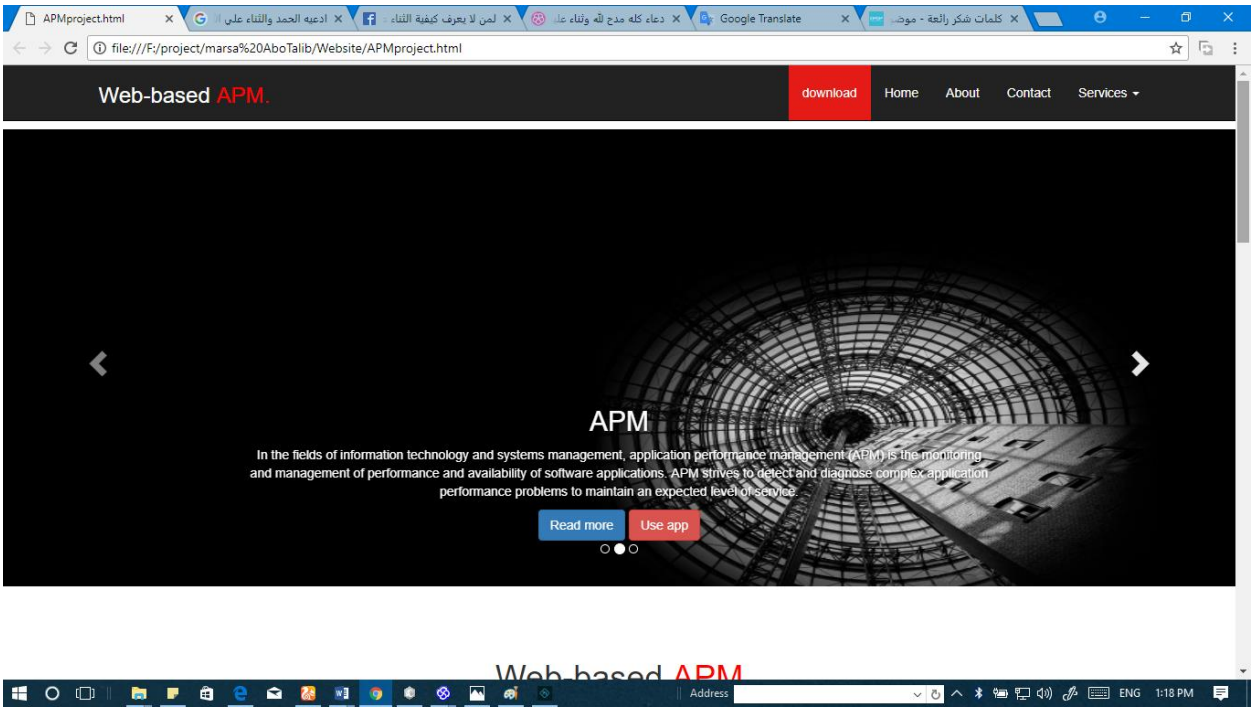
How to use this APM Tool

All people can visit our website to see and download this tool but it's more useful for administrator to show performance for specific application and his databases. If IT-Administrator want to use this APM tool,

First:

IT-Administrator must visit our website to download package contain all configuration to this tool.

Figure 8-1download APM



Second:

After this step the administrator will be able to setup this downloaded package to enjoy all APM services.

Figure 8-2 APM features