2019 1st International Conference on Cybernetics and Intelligent System (ICORIS)
22nd-23rd August 2019 at Institut Teknologi dan Bisnis (ITB) STIKOM Bali, Indonesia

A New PHP Web Application Development Framework Based on MVC Architectural Pattern and Ajax Technology

Stenly Ibrahim Adam
Department of Computer Science
Universitas Klabat
Airmadidi, Indonesia
stenly.adam@unklab.ac.id

Stevani Andolo
Department of Computer Science
Universitas Klabat
Airmadidi, Indonesia
s11610034@student.unklab.ac.id

***Abstract*—Hypertext Preprocessor (PHP) web application frameworks have become significantly popular in web development, by providing built-in libraries which could help developers to write codes without writing them from scratch. Because of their popularities, most developers are required to have solid knowledge on using the available frameworks. To create a dynamic and real-time web application, most of the developers use Ajax technology due to what it is capable of. However, most of the existing PHP frameworks do not have built-in Ajax libraries where users need to implement their own Ajax request. Therefore, in this paper, we develop a new PHP web application development framework based on the Model View Controller (MVC) architectural pattern and Ajax technology. The framework itself implements Ajax technology with its built-in library. After conducting this research, we concluded that the new PHP web application framework could be used as a web application development tool to help users to create a dynamic and real-time web application.**

Keywords—PHP application framework, Ajax technology, MVC architectural pattern

I. Introduction

A. Background of Study

Web technology has been increasing and improving in the last few years and it becomes one of the major improvements in Information Technology (IT) world [1]. Because of these changes, impacts can be seen in the business world, where most of the business people tend to use and are using technology like Web Services to increase their productivity in generating profits, maximum outcomes, fulfilling customers' needs and managing products or services. In addition, this kind of technology is a tool used to make jobs easier than using a manual approach [2]. Therefore, in responding to these changes, there have been various Hypertext Preprocessor (PHP) based Frameworks that have been built to help and empower developers in creating a well-managed and more structured Web application.

According to [3], there are some of the most popular PHP Frameworks that are available for Web developers to use, which are Laravel, Symfony, Codeigniter, Yii, Zend, CakePHP, and Inne Framework. Based on this study, a good Framework is the one that has tools such as Models, Create Read Update Delete (CRUD) tools, Controller, Views, Forms, and Modules. Not only that, but they also stated that technology and programming techniques such as Model View Controller (MVC), Database Object and Ajax are very important in Web Application [3].

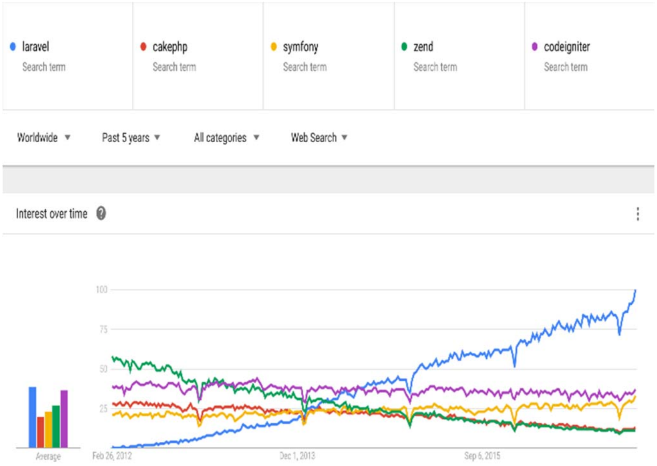


Fig. 1. The Most Popular PHP Frameworks in 2018.

Fig. 1 shows the chart of all the most popular Frameworks with its popularity in 2018. From Fig. 1, we also can see that PHP frameworks became more popular from time to time. However, most of the popular PHP frameworks are not contain any Ajax built-in libraries. In other words, users will still need to create their own Ajax request with a configuration that is required. Therefore, in this paper, we develop a new PHP Web Application Development Framework based on MVC architectural pattern and Ajax technology. The framework is called "MadeLine". After conducting this research, we concluded that the new PHP web application framework could be used as a web application development tool to help users to create a dynamic and real-time web application.

II. Literature Review

A. Hypertext Preprocessor (PHP)

PHP is considered and known as one of the most popular scripting languages used in Web development. PHP became popular from time to time because of its flexibility and the ease of using and learning [4]. Moreover, because of its flexibility, PHP is and can be used for client and server site and has been used by the servers around the world as part of the Cross Platforms Apache, MySQL, PHP (XAMPP) [5].

According to [6], PHP is a "powerful" language used in developing dynamic and interactive web applications; this is because of one of the defining features PHP offers to developers, which is the ease of connecting and manipulating the Database due to the built-in Database functions provided

978-1-7281-1474-3/19/$31.00 ©2019 IEEE

by the PHP itself. In addition, PHP is a powerful language because it offers some key advantages, such as performance, scalability, open-source, and portability.

B. PHP Framework

PHP Framework is a tool used for Web programming, which makes Web development better and well-structured. Not only that, Framework has impacts on increasing the productivity of development as there is no need to write from scratch for commonly used tasks [7]. As the Web technology era changes, PHP Web Application Framework continues to evolve from the pure or native full-stack until the implementation of MVC that is now popular and has become the base of most of the Frameworks [8].

C. MVC Architectural Pattern

Model View Controller (MVC) Architectural pattern is a pattern used to build a web site, which has become the based architecture for most of the PHP Frameworks. Developing any types of applications can be a pain every developer faces due to managing or separating User Interface (UI) from the application logic [9]. Therefore, to avoid those difficulties and have an efficient, reusable, stable, and scalability application, adopting an MVC pattern is an effective solution [10].

Implementing the use of MVC pattern in building a Framework makes the application development process becomes well-structured. The pattern is abstracted into three distinct parts as model, view, and controller that [11].

- Model behaves as a container used to encapsulate core data, logical relationship, and business rules of the application. It responds to any request from controller and or view.
- The view is the application interface shown or seen by the clients and interacts with the clients. A view will be loaded in the browser by the controller according to the client’s request.
- The controller is the processor of client's request. It requests the data from the model and loads the view accordingly [12].

D. JavaScript and Ajax Technology

JavaScript is part of front-end in web development. According to [13], JavaScript is a scripting language that is mainly used for client-side interactivity. In addition, [14] languages like PHP and JavaScript have become evident of popular scripting languages used today.

Asynchronous JavaScript XML or a term that is already known as Ajax is a technique used in creating dynamic Web pages [15]. The phrase of Ajax was coined by Jesse James Garrett and Jesse also went on to say that Ajax is a powerful approach to building a Web Site and learning it is not as hard or complicated as learning new language [16]. With this technique, pages and data exchange activity become more efficient and stable. The core role of Ajax technology is to update data of the page without page refresh. This is done by "the use of a web page server XMLHttpRequest object data through the Document Object Model (DOM) object tree to insert data into page rendering". In addition, this technology enables user to do multiple operations because Ajax communicates asynchronously with the server; Google Maps is one of the Web Applications that use Ajax technology [17]. Another feature of what Ajax can do is that it can update one or multiple parts of a Web Site's data. MySQLi Extension and SQL

MySQLi is a relational database driver used in PHP scripting language to provide and interface with MySQL database. In most cases, MySQLi extension is implemented in Framework, which will expose its API to PHP developers in order for the developers to be able to use its facilities.

Structured Query Language (SQL) [18] is a special language used to retrieve, edit, update, and insert data into database. In addition, when it comes to communicate or interact with database, SQL is used to do the job [19]. According to [20], there is another SQL module implemented in apace, which is called Spark SQL. This module has the ability that can be used by the developers.

E. Extreme Programming

Using Extreme Programming model is extremely suitable for developers whose aim to get the product done in parts with its documentation because this methodology gives the ability to create the documentation while developing the actual application or software [21]. Below is the picture and description of Extreme Programming model.

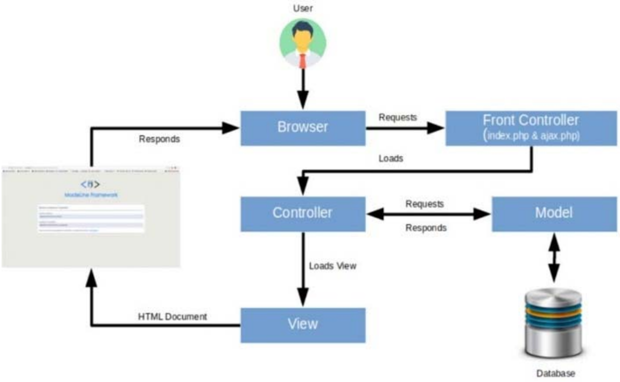


Fig. 2. Theoretical Framework.

1. The planning phase is where the strategy development that covers the release plan, completion plan for accepted stories and developers' meeting and communication among the project members.
2. The design phase is where the user's stories or requirements will be visualized accordingly.
3. The coding phase is done in two sections which are:
   - Pair programming is where the team members will be paired up to finish the assigned tasks.
   - Refactoring is where the code will be refactored by removing duplicated code, makes the code understandable and kept in good form.
4. The testing phase is done to test whether the product works or not and fix any encountered errors while performing a test.
5. The last phase is handing in the product to the customer.

III. Analysis And Design

A. Theoretical Conceptual Framework

Based on the chosen model, the conceptual framework for this research was structured as follows:

1. Planning the completion of the Framework based on the collected stories or requirements. Designing the process of how the Framework works based on the stories or requirements, using Unified Modeling Language (UML).
2. Doing the implementation of the framework based on the design.
3. Doing a test on the Framework and fix any encountered errors.

B. Application Conceptual Framework

Based on the chosen MVC architectural pattern, the application conceptual framework looked as follows:

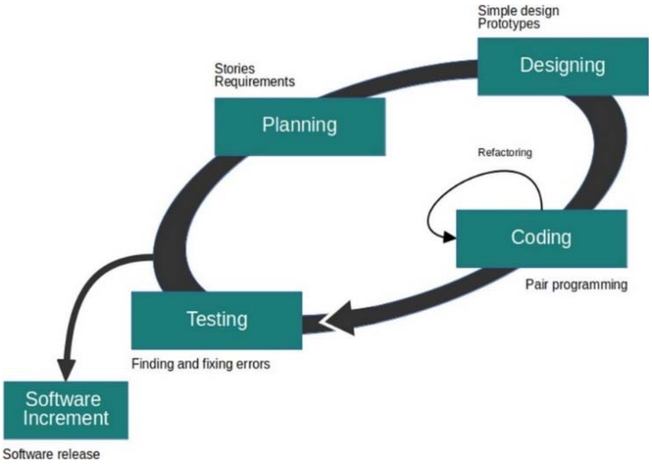


Fig. 3. Application Conceptual Framework.

1. User sends a request to the controller through a browser, which will be sent to the front controller.
   1. If it is an Ajax request, the front controller will use the *ajax.php*, otherwise, the default one will be chosen and that is *index.php*.
   2. Loads the requested controller.
2. Controller will process the request.
   1. Sends request to model.
   2. Model sends query to database.
   3. Model responds back to controller.
3. Controller loads requested view.
4. View loads the HTML document.
5. View responds back to user through browser.

C. Application Design

In order to analyze how the framework works, we used Component Diagram to gain the overview of the framework functionality specifically the interaction of the framework towards the users who are going to use the framework. Below is the component diagram of the framework.

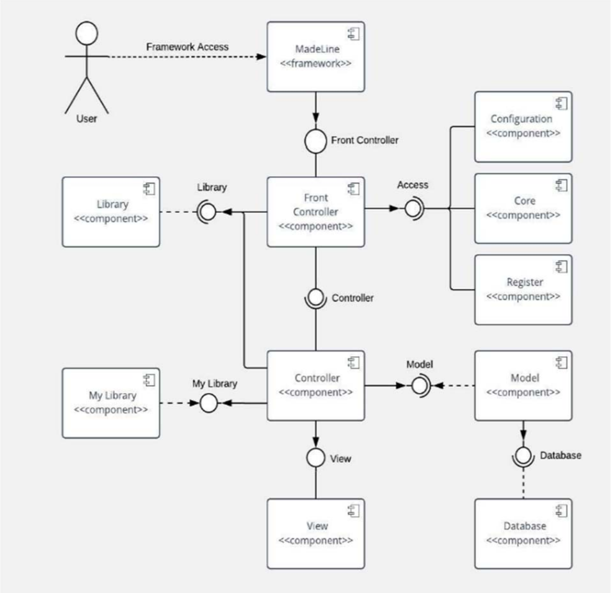


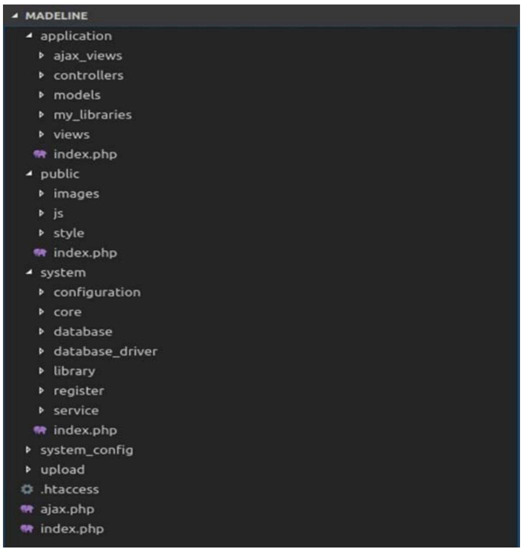
Fig. 4. Framework Component Diagram.

1. User accesses the "*MadeLine*" Framework.
2. "*MadeLine*" loads the requested front controller: default or Ajax controller.
3. Front controller includes required libraries, core and register files.
4. The front controller will get the requested controller.
5. Controller can contain required libraries, my libraries and models.
6. The controller loads the requested view through the invoked method.
7. The view is returned back and displayed to the user.

IV. Implementation

A. Framework Directory Structure

Based on the analysis and design for this framework, the directory structure looks as follow:



```
MADELINE
  application
    ajax_views
    controllers
    models
    my_libraries
    views
    index.php
  public
    images
    js
    style
    index.php
  system
    configuration
    core
    database
    database_driver
    library
    register
    service
    index.php
  system_config
  upload
  .htaccess
  ajax.php
  index.php
```

Fig. 5. Framework Directory Structure.

1. Application Directory will be used to store external directories for user ajax views (*ajax_views*), models, views, controllers and user's libraries (*my_library*).
2. Public Directory will be used to store application resources. For instance, media, JavaScript, and CSS.
3. System Directory is where all the "*MadeLine*" configuration, built-in libraries, core engines, database and database driver, register and service are stored.
4. System Config Directory (*system_config*) is where the configuration events based on Graphical User Interface (GUI) are stored.
5. Upload Directory has the same behavior as Public Directory where user can put any media, but it is created with the purpose of storing any uploaded files.
6. *.htaccess* is used for Universal Resource Identifier (URI).
7. *ajax.php* is used as a controller for Ajax request.
8. *index.php* is used as a default controller.

B. *Libraries.*

In this PHP framework, there are nine the built-in libraries for the framework. However, only Form Validation Class, File Class, and Ajax Technology will be illustrated below, just to have a visualization of how the libraries are used.

I. *Basic Form Validation Class.*

This class is and will be used for handling a form submission within a requested method in a controller, as follows:

1. Load the *Form Validation* class file by inserting the code:

```
$this->library(‘form’);
```

2. Set input rule by inserting the following code : If the input is required:

```
$this->form->input(‘input_name1’);
```

If the input is optional:

```
$this->form->input(‘input_name2’, 0);
```

or put them in one line as follows:

```
$this->form->input
(‘input_name1, input_name2:0’);
```

3. Validate the input using *Form Validation* by inserting the following code:

```
if($this->form->validate())
   { It returns true.
} else {
   It returns false.
}
```

II. *File Class.*

This class is and will be used for handling file upload within a requested method in a controller, as follows:

1. Set the configuration:

```
$config[’file_name’]   = ’input name’;
$config[’path]         = ’input name’;
```

To delete file (optional):

```
$config[’delete_file’] = ’input name’;
```

To resize file (image file only and optional):

```
$config[’resize’]      = ’input name’;
```

Optional:

```
$config[’type’]        = ’file type’;
$config[’max_size’]    = ’maximum file
size’;
```

2. Load the *File* class file by inserting the code:

```
$this->library(‘file’);
```

3. Validate the input using *Form Validation* by inserting the following code:

```
if($this->file-
   >upload()) { It
   returns true.
} else {
   It returns false.
}
```

III. *Ajax Technology*

When using *Ajax*, the users are only required to have basic knowledge of JavaScript and bit of knowledge of creating a controller that will respond to the request. As an example, an onclick event will be used. Below is how to create one *Ajax* function:

1. Create a button in a view.
2. Create an onclick event of a button that responds to the button based on its id name or class name created in a view.
3. Create one *file_name.js*, save it in public/js/ directory and call the file in a view by: `js(’file_name’);`
4. Insert the following codes and save:

```
$(function() {
    action.click(‘id_name’, ‘new_call’);
});


Array data is an array sent from the top
click function.

function
    new_call(array_data) {
    var controller =

        ajax.set_controller(’test’)
    ;  var method =
    ajax.set_method(’try’);

    var data   =
    ajax.set_method(’MadeLine’); Var
    ajax_data = (controller, method,

        data);

    array_data[’data’] = (ajax_data);
    ajax.request(array_array,
    'new_return');
}
```

```
Respond is a response from a requested
controller.

function
    new_return(response) {
    alert(response);
}
```

5. Create requested controller and save it in application/controller/.

```
<?php

    DEFINED('AJAXPATH') OR

    exit(header('location: 403'));


    Class Test extends ML_Controller {

      $data is sent from the
      new_call function in.

      Public function
        try($data) { echo
        'Hallo '.$data[0];

      }

    }

?>
```

IV. *HTML Generator.*

HTML generator was designed for users who like to write HTML tags using PHP. In other hands, it allows users to write a view file using only PHP. Below are some of the syntaxes.

- Open html tag:

  ```
  to('html_tag');
  ```

  Adding attributes within the HTML tag by using (::) for = and separating multiple attribute values with (*).

  ```
  to('div', 'class::class*class1 id::id');
  ```

- Close html tag:

  ```
  tc('html_tag');
  ```

V. *User friendly URL Segment.*

URL segment is best used for identifying what controller, method, and other data within the URL. "*MadeLine"* provides a method of getting the URL segment through an array-based URL segment. The way of getting the URL segment in this framework is as follows:

- To get a single segment, use `URI`[index] constant.
- To get the last segment, use `LAST_URL` constant.
- To get number of URL segments, use `URIS` constant.

VI. *MadeLine Query Liner*

"MadeLine" Query Liner is built-in query builder or active records that will be used to construct a query against the database. Below are only some of them.

- To get one row use (1), to get all rows (), to get total rows ('*c*') and to preview the query ('*show*'). Below is an example of getting one row that produces:

  "SELECT * FROM `table_name` WHERE cn = 'cv' AND cn != 'cv' OR cn = 'cv'".

  ```
  return $this->all('table_name')
              ->where("cn::cv")
              ->and("cn::cv", '!=')
              ->or("cn::cv")
              ->get(1);
  ```

- To update a data, look at the following example that produces:

  "UPDATE `table_name` SET cn = 'cv' WHERE cn = 'cv' AND cn != 'cv' OR cn = 'cv'".

  ```
  $this->set('cn::cv')
       ->where("cn::cv")
       ->and("cn::cv", '!=')
       ->or("cn::cv")
       ->update('table_name');
  ```

- To insert a data, look at the following example that produces:

  "INSERT INTO `table_name` (`cn1`, `cn2`) VALUES ('cv1', 'cv2')".

  ```
  $this->data('cn1::cv1')
       ->data("cn2::cv2")
        ->insert('table_name');
  ```

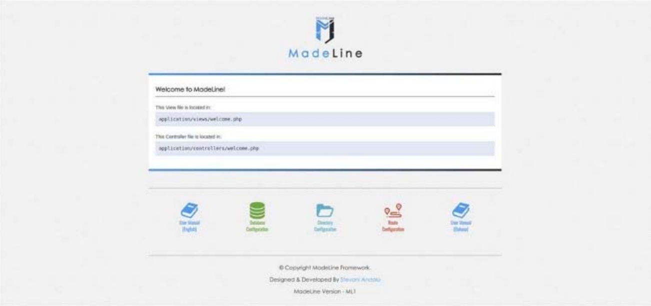
- To delete a data, look at the following example that produces:

  "DELETE FROM `table_name` WHERE cn = 'cv' AND cn != 'cv' OR cn = 'cv'".

  ```
  $this->where('cn1::cv1')
       ->and("cn::cv", '!=')
       ->or("cn::cv")
       ->delete('table_name');
  ```

C. *Displaying Framework Welcome Page*

Welcome page is the default page for the framework that shows the current default controller name and location and default method that has been invoked, and the view file and its location. The welcome page also displays the manual and configuration.



Welcome to MadeLine!

This View file is located in:

application/views/welcome.php

This Controller file is located in:

application/controllers/welcome.php

© Copyright MadeLine Framework.
Designed & Developed By Stevani Andolo
MadeLine Version - ML1

Fig. 6. Framework Welcome Page

The process of getting the welcome page shown, involves the following steps.

1. Default loader (*index.php*) will be invoked and it will check the default controller and method in *route.php* configuration file.
2. If the default controller is found and so is the method, the default controller and method will be invoked.
3. The controller will look for a file name "*welcome"* in views directory within the application directory and load it up if it is found.
4. The controller file should look as follow:

```
<?php
    DEFINED('AJAXPATH') OR
    exit(header('location: 403'));


    Class Welcome extends
        ML_Controller { Public
        function index() {
          $this->view('welcome');
        }
    }
?>
```

V. CONCLUSIONS

In conclusion, a new PHP Web Application Development Framework based on MVC architectural pattern and Ajax technology has been completed. Based on the implementation section above, we can see that, the framework can be installed and run across platforms, provides human friendly configuration with the use of GUI, provides one configuration for multiple use purposes, provides the manual in English and Bahasa, provides a library for multiple-use purposes, provides various styles in "MadeLine" Query Liner, let users easily get URL segments and does not require users to set the base path. Therefore, we can see that the new PHP web application framework could be used as a web application development tool to help users to create a dynamic and real-time web application.

REFERENCES

[1] S. Farrukh, "Modern and Responsive Mobile-enabled Web Applications," *Procedia Computer Science,* vol. 110, pp. 410-415, 2017.

[2] A. Afef, G. S. Yachi, "Approach based on web services for business process adaption", *Elsevier B.V,* 2015.

[3] A. Zurkiewicz, M Milosz, "Selecting a PHP Framework for a web application project – the method and case study", 2013.

[4] L. Xiaosong and C. Jahanzaib A, "An Empirical Study of Three PHP Frameworks," *International Conference on Systems and Informatics*, 2017.

[5] A. Theodoros and C. Alexander, "Studying the evolution of PHP web applications," *Elsevier*, 2016.

[6] S. Chanchai, "PHP framework for database management based on MVC pattern," *International Journal of Computer Science & Information Technology*, 2011.

[7] P. Natalya and B. Victoria, "Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems," *Procedia Computer Science*, 2017.

[8] S'A. Umi, NH. J Akhmad and H. Masfu, "Implementing Singleton method in Design of MVC-Based PHP Framework," *International Electronics Symposium (IES)*, 2015.

[9] Z. Hui, Z Hoa and C. Qijun, "The Research of Dynamic Statistics Chart Based on MVC Design Pattern," *International Conference on System Science ang Engineering (ICSSE),* Shanghai, China, 2014.

[10] N. Wang, L. Li, Y. Wang, Y. B. Wang and J. Wang, "Research on the Web Information System Development Platform Based on MVC Design Pattern," *International Conference on Web Intelligence and Intelligent Agent Technology*, 2008.

[11] H. Shu-qiang and Z. Huan-ming, "Research on Improved MVC Design Pattern Based on Struts and XSL," *International Symposium on Information Science and Engieering*, 2008.T. Wang and L. Jiang, "Research on IETM Publishing System Based on MVC Pattern," *WASE International Conference on Information Engineering*, 2010.

[12] A. Chlipala, "Ur/Web: A Simple Model for Programming the Web,"*Association for Computing Machinery* (ACM), 2015.

[13] A. Theodoros and C. Alexander, "Studying the evolution of PHP web applications," *Information and Software Technology*, no. 72, pp. 48-67, 2016.

[14] H. Wu, H. Chen and T. Qin, "Research on Manufacturing SME Information Management Systems Based on Ajax and MVC," *International Conference on Measuring Technology and Mechatronics Automation*, 2015.

[15] L. Zhijie, W. Jiyi, Z. Qifei and Z. Hong, "Research on Web Applications Using Ajax New Technologies," *2008 International Conference on MultiMedia and Information Technology*, 2018.

[16] S. Takeo, T. Takashi and N. Hideo, "GIS Crisis-management Systems Using Ajax Technology," *SICE Annual Conference*, 2008.

[17] A. B. M. Ali, A. Y. I. Shakhatreh, M. S. Abdullah and J. Alostad, "SQL-injection vulnerability scanning tool for automatic creation of SQL-injection attacks," *Procedia Computer Science*, vol. 3, pp. 453-458, 2011.

[18] S. Z. Barbhuiya, B. K. Ray, Z. Azim and Y. J. Singh, "Suggestive Local Engine for SQL Developer: SLED," *ADBU-Journal of Engineering Technology,* vol. 4, pp. 23-27, 2016.

[19] M. Armbrust, R. S. Xin, C. Lian, Y. Huai, D. Liu, J. K. Bradley, X. Meng, T. Kaftan, M. J. Franklin, A. Ghodsi and M. Zaharia, "Spark SQL: Relational Data Processing in Spark," *ACM*, 2015.

[20] C. Jitender and Dr. S. Ugrasen, "Iterative Maintenance Life Cycle using eXtreme Programming," *International Conference on Advances in Recent Technologies in Communication and Computing*, 2010

[21] C. Wei, H. Lin, L. Lijing and L. Jing, "The Research of PHP Development Framework Based on MVC Pattern," *International Conference on Computer Sciences and Convergence Information Technology*, 2009