

Design an MVC Model using Python for Flask Framework Development

Mohammad Robihul Mufid, Arif Basofi, M. Udin
Harun Al Rasyid

Politeknik Elektronika Negeri Surabaya, Indonesia
mufid@pasca.student.pens.ac.id, {ariv,
udinharun}@pens.ac.id

Indhi Farhandika Rochimansyah, Abdul rokhim
Akademi Komunitas Negeri Lamongan, Indonesia
indhifarhandika@gmail.com, abd.rokhim@stmik-
yadika.ac.id

Abstract— The Model-View-Controller (MVC) framework has become the standard in modern software development, with the model layer, display layer, and controller layer making it easier and faster. The Flask is a framework that uses Python language with easy to understand code writing. But the Flask framework still doesn't use the MVC method, so files and codes are not regular. The purpose of this study was to design a MVC for a framework that uses the Python programming language. This system has a generator that can make MVC folder structure easily and quickly, this system is also equipped with the Bootstrap framework, and this system is open source. The results showed that the presence of MVC on the flask framework could make users easier in creating new projects and have faster fully load time.

Keywords— Framework, Model-View-Controller (MVC), Python, generator

I. INTRODUCTION

Today, the development of websites has been going a long way since the beginning of the World Wide Web (WWW), many programming technologies and languages are now being used to build web applications. One of the technologies for building web applications is MVC. MVC is a method for building web by separating the model layer, controller layer, and display layer so that it is easier and faster. MVC has shown its benefits for interactive web applications that allow multiple representations of the same information, promote code reuse, and help developers focus on certain application features. The MVC framework has broadly become a standard in modern software development [1].

Framework Flask is a web framework from Python language. Flask provides a library and a collection of codes that can be used to build websites, without the need to do everything from scratch. Because of its simple features, the flask will be lighter and not dependent on many external libraries that need attention. In general, the flask provides 'Werkzeug' which is useful for receiving requests (url) and responding [2]. But Framework flask still doesn't use the Model View Controller (MVC) method. So there are weaknesses including the difficulty in making repetitive codes and source code management. Therefore, to make it easier for

web developers who use framework flask, MVC is needed in the framework flask.

There have been several previous studies on the use of MVC, including Sarker et al. [1] proposed the MVC design to be implemented in the Java Framework in developing desktop application research. In the proposed MVC it can separate data, display, control software, and can also achieve the separation of the business logic layer and presentation layer. The results of the research conducted show that software can be developed effectively using this MVC framework, and this framework is also stable, efficient, and able to develop high-quality applications.

Paul et al. [3] proposes a model for developing fast web applications based on the Model-View-Controller (MVC) architecture which has several useful components such as security, form validation and creation, as well as database access and routing. The proposed MVC model uses the PHP programming language, but can be implemented in other languages and development environments using the same concept. The results show that this model can reduce web application development time, and allow developers to focus on application specific tasks, rather than wasting time trying to apply well-known patterns and practices.

Xu et al. [4] proposed a study of the application of the MVC-based Struts framework to be applied in the online rural teacher training system in China. This system is based on the J2EE platform, which provides types of components and containers for program development in carrying out MVC roles, and makes development easier and more efficient.

Wang et al. [5] proposed the implementation of the application of code reuse technology based on the MVC concept applied to a comprehensive fishing vessel inspection service system project. The results of the study indicate that a combination of code reuse and MVC framework can improve the efficiency of code development and maintenance.

Zhang et al. [6] proposed research on the design of a lightweight MVC framework based on Spring MVC and Mybatis. Spring MVC can do method-level interceptors for requests to improve operating efficiency, and completely separate business logic and display layers. Whereas by using Mybatis as a persistence layer framework, it can increase the

speed of development, database access, and bring more space to improve system security, performance, and maintenance.

Pan et al. [7] proposed a web application with the MVC framework for IEEE floating-point standards using java language. This system also uses several web optimization techniques to improve performance, such as RWD, AJAX, HikariCP, and there are two software design patterns such as DAO and DTO. The test results are determined by a simple algorithm.

Singh et al. [8] proposed designing a lightweight MVC framework for developing small to medium-size web applications, using java language. Java has many MVC frameworks, but two of them stand out in popularity like Spring and Struts 2. Spring and Struts 2 still lack support for the latest technology and are too complicated with steep learning curves, making Spring and Struts inadequate for developing small web projects to medium. The results of this study the researchers created a framework such that it is flexible, easy to learn and efficient at the same time.

Wijaya et al. [9] implement the MVC method on the Bakti Prabumulih Vocational School website. This application is made with the PHP programming language and uses MySQL as its database. The results of this study make it easy to control and process data information about schools.

The aim of this study was to design an MVC on the Flask framework using Python language, to help developers improve the speed and quality of work, and provide a Flask framework that is better for both novice or experienced users. This system has a project generator that will make folders and files structured and automatic, so users don't need to create manual folders and files.

II. OVERVIEW OF MODEL-VIEW-CONTROLLER (MVC)

In this section, the model-view-controller path will be used in the framework flask. The model-view-controller method was first introduced by Trygve Reenskaug in the 1970s. This method is divided into three interconnected parts, namely the model, view, and controller [1]. Figure 1 shows the flow of the model-view-controller.

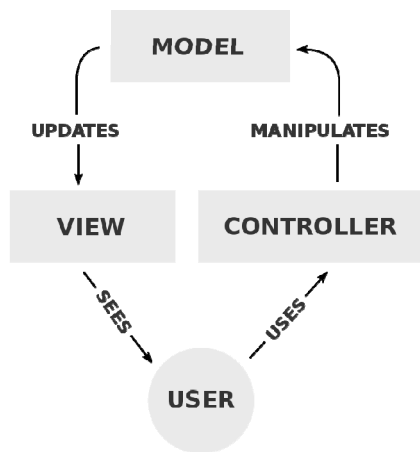


Fig. 1. The MVC pattern

- Model : dealing directly with databases to manipulate data (insert, update, delete, search), handle validation from controller parts, but cannot deal directly with the view section.
- View : the part that handles the presentation logic. In a web application this section is usually an HTML template file, which is set by the controller. View functions to receive and represent data to the user. This section does not have direct access to the model section.
- Controller : the part that regulates the relationship between the part of the model and the part of view, the controller functions to receive requests and data from the user then determine what the application will process.

III. OVERVIEW OF FRAMEWORK FLASK

Flask is a micro framework web written in the python programming language, flask was created by Armin Ronacher in 2004. Flask is licensed under a three clause BSD License, flask is designed to make web applications quickly and easily, with the ability to improve complex applications. It started as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks. Figure 2 shows an example of a flask script [2].

```

from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'
  
```

Fig. 2. A simple example of a flask script

IV. SYSTEM DESIGN

In this section we will explain the system design to implement the MVC method on the framework flask. This system design will create a folder that has an MVC structure and there are important files. Figure 3 shows the system process diagram.

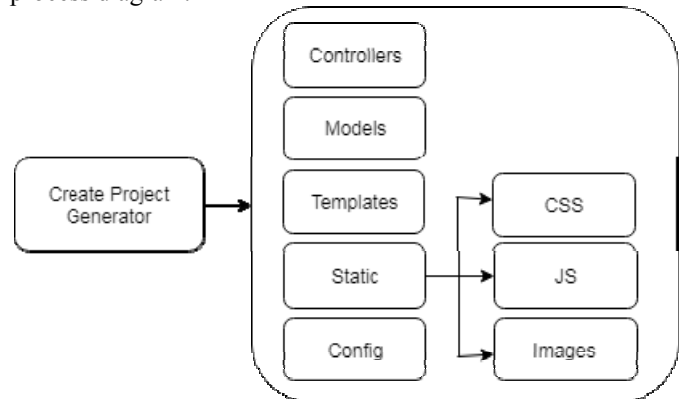


Fig. 3. System Process Diagram

In this study to implement MVC on the framework flask, it will be divided into the following stages:

a) Create Project Generator

At this stage, this system has a project generator that will make folders and files structured and automatic, so users don't need to create manual folders and files. Figure 4 explains how to create a controller folder and create a `__init__.py` constructor file, to create a model folder, view, static, config is also the same as creating a controller folder, but the difference in naming the variable.

```
# Membuat Folder controllers dan isinya
print(f"Make {ctr}")
os.makedirs(f"{base}/{pr}/{ctr}")
f = open(f'{base}/{pr}/{ctr}/__init__.py', 'w+')
f.write("""import os, glob

__all__ = [os.path.basename(f)[:3] for f in glob.glob\
(os.path.dirname(__file__) + "/*.py")]""")
f.close()
```

Fig. 4. Summary of Source Code Generator

b) Produce the MVC Structure

In this section we will explain our system design to implement the MVC method on framework flask. Main architectural concept in this system, has three components such as Model, View and Controller. Figure 5 shows the MVC pattern on the system to be created. First, the user sends a data request to the controller, then the controller receives the data request from the user. If the data request from the user requires data access from the database, the controller will send to the model, then the model will retrieve the data needed from the database and will be sent to the controller after the controller receives data from the model, the data will be sent to view.

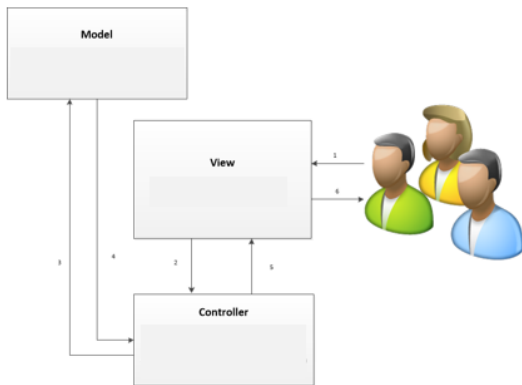


Fig. 5. The MVC Pattern in this system

The following is the folder structure that has been generated by the MVC generator. Figure 6 explain about the MVC folder structure that is in the python flask framework :

- Templates or views, is the part that handles the presentation logic. In a web application, this part is usually an HTML template file, which is set by the

controller. The function of view block is to the user. Section views does not have direct access to the model section.

- Model usually dealing directly with databases to manipulate data (insert, update, delete, search), handle validation from controller parts, but it can't deal directly with the view section.
- Controller is a part that regulates the relationship between the part of the model and the part of view, the functions controller of is to receive requests and data from the user then determine what the application will process.
- Static is for saving files css, javascript, and also to save images.
- Config is part of managing database configuration and other system configurations.

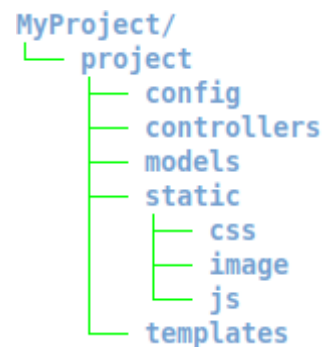


Fig. 6. Project Folder Structure

c) Generate Files CSS, Javascript, Python, and HTML

The project generator also creates css, javascript, python, and html files that are already in the folder.

- In the basic project folder there is 1 python file, namely: `artisan.py` which functions to run the server, create a controller and make a model.
- In the project folder there is 1 python file, namely: `__init__.py` is a file constructor that will automatically run.
- In the CSS folder there are 2 css files, namely: `cover.css` to design html files and `bootstrap.min.css` are files from framework version 4.3
- In the js folder there are 2 javascript files, namely: `popper.min.js` and `bootstrap.min.js` are javascript files from the bootstrap framework version 4.3
- In the templates folder there are 3 html files, namely: `index.html`, `about.html`, and `contact.html` are the web start page when the generator creates a new project.
- In the model folder there are 2 python files, namely: `__init__.py` is a file constructor that will read the python file in the model folder. And `hello.py` is the initial python file to set the model.
- In the controllers folder there are 4 python files, namely: `__init__.py` is a constructor file that will read files in the controller folder. Then there are `index.py`, `contact.py`, and `about.py` are the initial python files

created by the generator.

- In the config folder there are 2 python files, namely: `__init__.py` is a constructor file that will read python files that are in the config folder, and `Database.py` for mysql database settings.

V. PERFORMANCE EVALUATION

In this section a testing process will be carried out to determine the performance of the MVC model design on the Flask framework using the Python language. In this test, it will be divided into 2 parts, namely the implementation of website creation and analysis of the performance of Flask MVC.

a) Implementation of website creation

To create a new website, using Flask MVC we just make it from a project generator. the project generator will create folders and files in a structured and automatic manner, so users do not need to manually create folders and files. Figure 7 is a display of a generator project from Flask MVC. There is one input to the project generator that is entering the name of the project that we will create. After the project is created, then the project will automatically have an MVC folder structure that is also equipped with a web server named `Artisan.py`. Figure 8 is the view to run the server, the user must enter the project folder then open terminal / cmd, then type "python artisan.py runserv", the system will run the server and output the output to inform the user that the server is running, then open the browser and enter the url `http://localhost:8000/`. The system will look for a controller named `index.py` and will run it. Figure 9 is the start page view of a website that is created automatically from the use of the MVC Flask generator. The home view is connected to the `index.py` controller.



Fig. 7. Display Generator Project

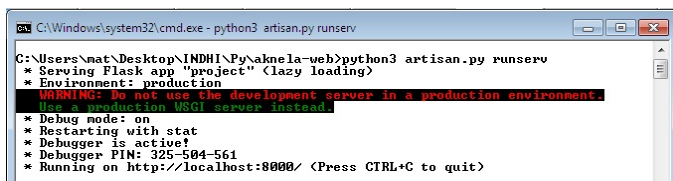


Fig. 8. Display when running the server

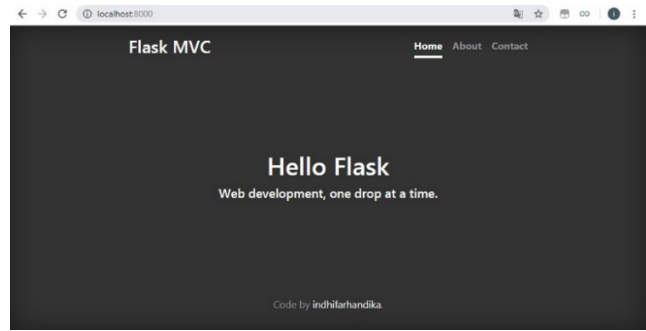


Fig. 9. Display Home

Figure 10 is the display about connecting to controller `about.py`.

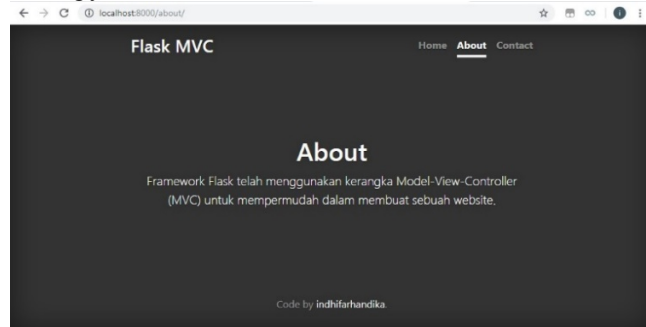


Fig. 10. Display About

Figure 11 is the display contact connecting to controller `contact.py`.

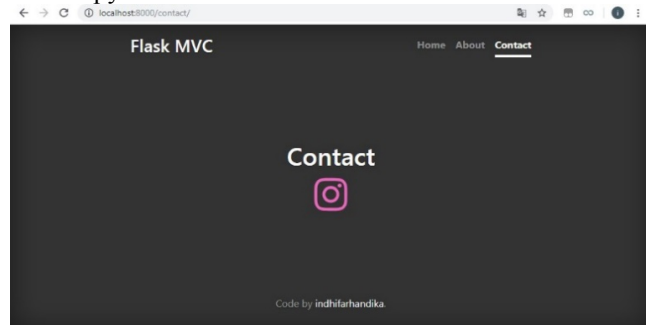


Fig. 11. Display Contact

b) Analysis of the performance of Flask MVC

System analysis is intended to analyze the performance of the system that has been made if it is as expected, the following are the results of the analysis of the performance of the website using the GTmetrix tool. The website to be tested is a website that uses the Flask MVC framework with a website that uses the Flask framework without MVC, with the same file and server size. Metrics measured include PageSpeed, YSlow Score, Full Load Time, Total Page Size, and Relative Request. Figure 12 shows the PageSpeed and YSlow scores, the results of the PageSpeed score and the YSlow score on the MVC Flask with the same Flask. The PageSpeed score is the result of analyzing web pages based on

Google's rules. YSlow's score is the result of analyzing web pages based on Yahoo's rules.

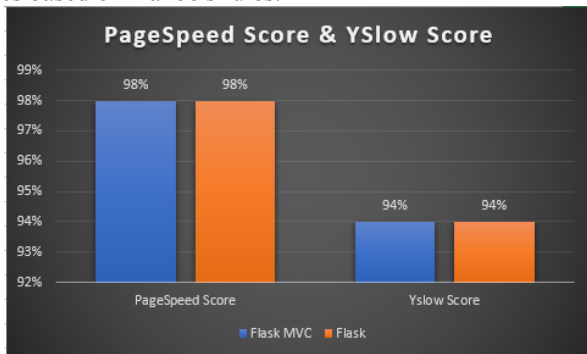


Fig. 12. Result PageSpeed and YSlow Score

Figure 13 shows the results of Fully Loaded Time on Flask MVC is 0.473 seconds and the results of Flask are 0.5 seconds. This result show that Flask MVC have time faster than Flask in load total page. Figure 14 shown the result of total page size on Flask MVC and Flask. Total Page Size is the total of files on the website. Figure 15 shown the result of the score request that is in the MVC Flask and Flask. From the experiment above it can be concluded that the use of MVC in the python flask framework greatly facilitates and speeds up performance when creating websites, website speed formatting when tested faster using MVC.

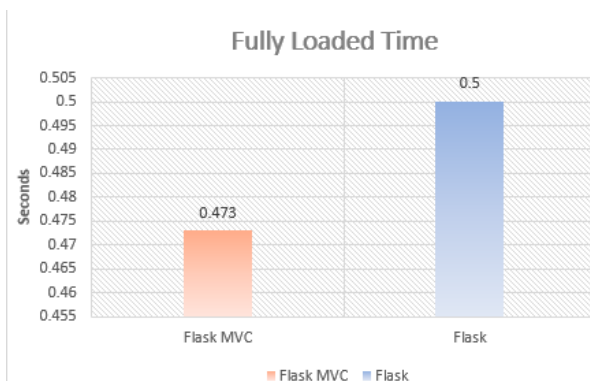


Fig. 13. Result Fully Loaded Time

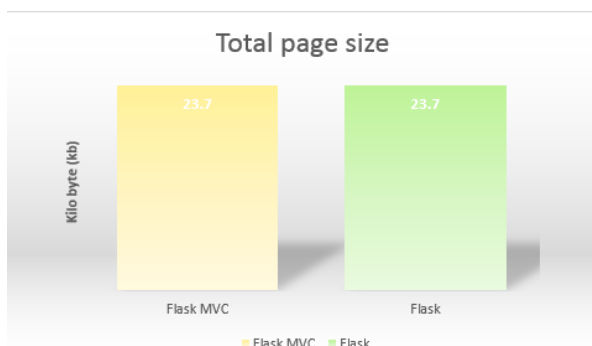


Fig. 14. Result Total Page Size

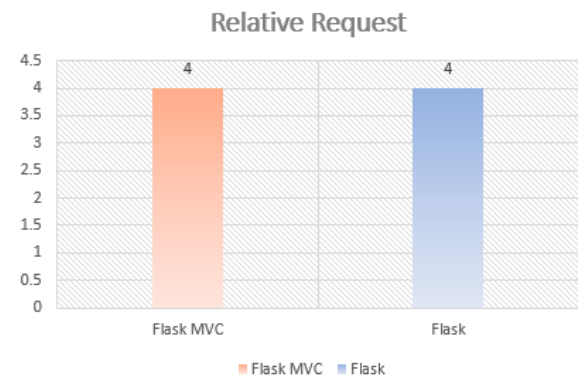


Fig. 15. Result Relative Request

VI. CONCLUSION

MVC is a method for making applications concise and fast, but unfortunately, in framework Flask it still doesn't use the MVC method. In this study, we aim to provide easy website creation using the MVC method in framework flask with python language. Our system has been equipped with a project generator and is also equipped with a bootstrap framework which will make it easier to design the appearance of the website. Our system helps developers improve the speed and quality of work, and provides a framework and work platform that is good for novice or experienced users. For further research, it is expected to add features that are still not available on our system, like add JQuery Framework and connect to the Oracle database.

REFERENCES

- [1]. I. H. Sarker, K. Apu, "MVC Architecture Driven Design and Implementation of Java Framework for Developing Desktop Application," International Journal of Hybrid Information Technology, Vol.7, No.5, pp. 317-322, 2014.
- [2]. G. Miguel, "Flask Web Development," O'Reilly Media, Inc. 2014.
- [3]. D. P. Pop, A. Altar, "Designing an MVC Model for Rapid Web Application Development," Elsevier Ltd, Vol. 69, pp. 1172-1179, 2013.
- [4]. S. Xu, T. Yang, "Application of Struts framework based on MVC in Online Countryside Teachers' Training System in China," International Conference on Multimedia Technology, IEEE, pp. 6252-6255, 2011.
- [5]. X. Wang, B. Xu, R. Gu, "The Application of Code Reuse Technology Based on the MVC Framework," International Conference on Computer Sciences and Applications, IEEE, pp. 534-537, 2013.
- [6]. D. Zhang, Z. Wei, Y. Yang, "Research on Lightweight MVC Framework Based on Spring MVC and Mybatis,"

International Symposium on Computational Intelligence and Design on IEEE, Vol.1, pp. 350-353, 2013.

- [7]. C. C. Pan, C. C. Lin, "Designing and Implementing a Computerized Adaptive Testing System with an MVC Framework: A Case Study of the IEEE Floating-Point Standard," International Conference on Applied System Invention (ICASI), pp. 609-612, 2018.
- [8]. A. Singh, P. Chawla, K. Singh, A. K. Singh, "Formulating an MVC Framework for Web Development in JAVA," International Conference on Trends in Electronics and Informatics (ICOEI), pp. 926-929, 2018.
- [9]. K. Wijaya, A. Christian, "Implementasi Metode Model View Controller (MVC) Dalam Rancang Bangun Website SMK Yayasan Bakti Prabumulih," Paradigma-Jurnal Komputer dan Informatika, Vol. 21, No. 1, pp. 95-102, 2019.