**PROJECT PROPOSAL**

# **Advancements in Computer Software Engineering**

**TITLE**

**Creation of a PHP MVC Framework with OOP**

**BY**

**ONI ADESOLA AYOMIDE**

**LUC-NGA-002-ADM-1000398**

**&**

**MAMA MESHAC EBERENNA**

**LUC-NGA-002-ADM-1000281**

1. **Introduction**

The Model-View-Controller (MVC) architectural pattern is a fundamental approach in modern web development, enabling the creation of organized, scalable, and maintainable applications. While there are several PHP frameworks available, many do not adequately balance simplicity, flexibility, and the needs of contemporary developers.

This project proposes the creation of a new PHP MVC framework tailored to address these challenges. The framework will incorporate essential features such as built-in command line tools, migration support, and a simple MVC structure, all while leveraging Object-Oriented Programming (OOP) principles and supporting PHP 8 and upwards.

1. **Problem Statement**

Existing PHP frameworks often come with complexities that can overwhelm developers, particularly those working on smaller projects or those new to the MVC pattern. These frameworks may include unnecessary overhead or lack crucial features such as built-in functions and validations, simple routing, and session management. Additionally, ensuring proper security for the application folder from public users can be cumbersome with some frameworks.

The proposed project aims to create a PHP MVC framework that is lightweight, easy to use, and provides the necessary tools and features to streamline development without sacrificing security or performance.

1. **Proposed Solution**

The proposed solution is a PHP MVC framework that offers a simple yet powerful structure. Key features include:

* **Available Command Line Tools**: Streamlines development tasks such as project setup and migration.
* **NodeJs Installation**: The Framework can be installed with npm CLI.
* **Migration Support**: Facilitates easy database schema management.
* **Simple MVC Structure**: Provides an intuitive and organized approach to application development.
* **In-Built Functions and Validations**: Reduces the need for external libraries by offering essential utilities out-of-the-box.
* **Simple Controller-Based Routing**: Ensures easy mapping of URLs to controllers with minimal configuration.
* **Support for PHP Version 8 and Upwards**: Leverages the latest features and performance improvements in PHP.
* **Important Pager, Request, and Session Models**: Simplifies common tasks such as pagination, request handling, and session management.
* **Pre-Coded Controllers and Views**: Includes ready-to-use templates for common pages like Welcome, Login, Signup, Logout, and 404 errors.
* **Core Features**: Provides essential components like Config, Initialization, Database Functions, and Model Functions to streamline the development process.
* **Proper Session Management**: Ensures secure handling of user sessions.
* **Security for the App Folder**: Protects sensitive files and data from unauthorized access by public users.
* **Namespaces**: Organizes code and prevents naming conflicts, making the framework scalable and maintainable.
* **Error Display**: Helps a developer to debug by allowing Errors display in the browser.

*…..And Many More Features…..*

1. **Novelty and Originality**

The framework’s originality lies in its combination of simplicity and completeness. By integrating built-in tools and features that are typically found in more complex frameworks, this solution provides a unique balance that caters to both beginners and experienced developers. Its emphasis on security, proper session management, and the use of namespaces further distinguishes it from other PHP frameworks.

1. **How Does Our Solution Improve Upon Existing Solutions?**

While other frameworks may offer similar functionalities, they often do so with added complexity or lack key features such as migration, built-in validations, or proper session management. Our framework simplifies the development process by offering these features in a lightweight package, making it easier to set up and maintain web applications. Additionally, the focus on PHP 8 and upward compatibility ensures that developers can take advantage of the latest advancements in the PHP language.

1. **Technology Stack**

The project will utilize a technology stack designed to maximize performance and ease of use:

* Programming Language: PHP 8.x
* Framework Structure: Custom MVC with OOP
* Database System: MySQL
* Development Tools: TrueHost, Git, Node Js, Git Bash, npm CLI.
* Testing Frameworks: PHPUnit

1. **Project Objectives**

* Develop a fully functional prototype of the PHP MVC framework.
* Implement and test the core features, including routing, migrations, and session management.
* Ensure compatibility with modern development workflows and tools.
* Provide comprehensive documentation for developers.

1. **Expected Outcomes**

Upon completion, the project will deliver:

* A functional PHP MVC framework with all the aforementioned features.
* A set of pre-coded controllers and views ready for use in various applications.
* Detailed documentation and guides to facilitate quick adoption by developers.
* Validation of the framework's performance and ease of use through extensive testing.

1. **Project Timeline**
   * Research and Literature Review: 1 month
   * Design and Development: 3 months
   * Testing and Evaluation: 1 month
   * Documentation and Presentation: 1 month

## 10. Conclusion

This project will introduce a new PHP MVC framework that is both simple and powerful, addressing common challenges faced by developers today. By offering essential tools and features in an intuitive package, the framework aims to improve the efficiency and effectiveness of web development in PHP.