Journal of Electrical Engineering, Electronics, Control and Computer Science – JEEECCS, Volume 8, Issue 28, pages 7-14, 2022

An Approach Towards the Development of Init Framework

Fidelis Chete and Ebuka Odini
Department of Computer Science
University of Benin
Benin City, Nigeria
ebukaodini@gmail.com, odichet@yahoo.com

Abstract – **Presently, most web developers are faced with the challenges of choosing the best file structure and programming paradigm to implement, when developing a project. There is need to develop web applications faster and also to impact a better way to write and maintain cleaner PHP codes, so that database related codes are not seen in the same text range as a HTML view code This work introduces the use of web frameworks that provides developers with a semi-complete application to develop their own application on top of an already built one. The proposed web framework, called Init Framework solves the problem of Uniform Resource Locator (URL) maintenance, URL flexibility, decoupling etc. with the help of the Route Class developed to link URLs to Controllers. Init Framework supports Object Oriented Programming and also follows the Model-View-Controller (MVC) design pattern. The proposed framework also generates the semi-completed application for the developer to build upon with the help of the Init Generator Class. The Init framework was developed using PHP, a web scripting language; HTML, CSS and JavaScript to display the web page of the framework.**

Keywords-Init framework; Model-View-Controller; web frameworks; Object Oriented programming

I. Introduction

The rapid development of the internet for web-based application has resulted in the increased demand for a more reliable, scalable, efficient, secured and maintainable coding methodology [1]. A software framework in computer programming is an abstraction in which software providing universal features can be selectively modified by additional user-written code [2]. It is a standardized, modular software environment that offers basic functionality as part of a larger software platform to enable software applications, products and solutions creation. Support programs, compilers, code libraries, tool sets, and application programming interfaces (APIs) may include software frameworks that combine all the different components to allow project or system development [2]. [3] describes frameworks as an integrated collection of library codes that simplifies programming in any given language, and language as the actual syntax and grammar of software writing. Although programming languages will never be completely obsolete, for a number of reasons, a growing number of programmers prefer to work with frameworks and see them as the most modern and state-of-the-art alternative [3].

Most traditional web developers build applications with PHP, which included the combination of all segments of codes such as data access, business logic, and data representation. This, consequently, created development problems for large projects in particular, which can now be solved with the aid of frameworks [2]. Before web frameworks became very apt for the development of web applications, developers had to start developing every application from scratch. Web frameworks abstracted this so the developer can fully concentrate on the development of the application logic.

PHP has become an important and one of the most widely used web development language in recent times. Several programmers on the web choose to build applications based on PHP, thereby combining issues such as data access, business logic, and data representation layer together. This was the traditional way of programming applications for the web and has been observed to create development problems especially for big projects. The proposed Init framework attempts to solve these problems for developers to be able to create highly-scalable web applications, using a single programming language, PHP.

II. Brief Review Of Related Literature

Frameworks, by using a single programming language, help developers to create highly-scalable web applications. [4] understands how difficult it is to choose a web framework for a project and further explains what to consider when choosing a web framework. There are two types of web frameworks which are front-end frameworks and back-end frameworks. Examples of front-end frameworks are explained as follows:

Angular is considered as one of the best frameworks for developing rich single-page applications. It has the ability to develop complete client-side applications. The first version of Angular was developed by Google using Javascript but later releases adapted Typescript.

React, even though regarded as a framework is a Javascript frontend library. React is considered the most popular web development framework for Javascript and is known for its component-based architecture. React was developed and maintained by Facebook.

One of the most trusted web development frameworks is Ember. This framework, growing rapidly in the field of professional web development is mostly used for complex web application development. Ember is popular for its 2-way user data migration utility.

Flutter, written in dart, is a server-side web app for building desktop applications. It is mostly used to develop mobile iOS and Android applications. It is also used to create network applications for the Google Fuschia OS. Flutter was developed by Google.

Vue.js is a framework that is commonly known for its lightweight, minimalistic, efficient approach to building frontend applications. One of the reasons for its success is that you can integrate and use Vue.js on your product easily without any trouble and lags.

Examples of back-end frameworks are explained as follows:

Ruby on Rails is a model view controller (MVC) written in Ruby language. It reduces the development time and offers many libraries and tools thus making it easy for developers to solve complicated development problems easily.

Though one of the oldest and most popular web frameworks, Django, which was written in Python offers one of the best architecture designs for web frameworks which is the model view template (MVT) pattern. Django follows the Convention over Configuration philosophy and DRY concepts.

Spring is a web framework that is based on Java. The purpose of this framework is the capabilities to make the formation of J2EE applications. It follows the MVC pattern and was developed by Pivotal Software.

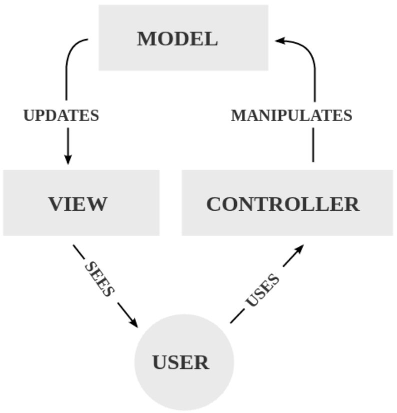
Laravel is a web framework that was built on PHP language. It is open-sourced and it comprises several PHP components and packages. Some of its features includes implicit model binding, simplified Eloquent global scopes, and middleware groups.

Express.js is an easy to use and maintain web framework that is based on Node.js. Its ideologies are simplicity and speed. Express is very flexible and can be used to develop web applications, complete websites and REST API.

A. Methodology

The study used the Model-View-Controller (MVC) design approach. MVC is a popular Architectural Pattern for building robust applications. It essentially defines a strategy for robust code design. It breaks down the application to different parts, in order to achieve higher separation of concerns between the app's modules [5]. MVC design pattern defines an effective way to separate the codes into layers so that each layer carries out a distinct activity. The major purpose of the MVC pattern is to split an application into separate layer that can work separately and yield same result. Fig. 1 depicts the concept of the MVC design pattern.

Figure 1. Concept of MVC design pattern [6]



The MVC is a design pattern that splits all the work to be done on the server side before returning response to the client side.

With the MVC design pattern comes one more technology called Routing. Like routing in networking, this routing is a program that uses the HTTP request URL and the HTTP request method as a key to map the functionality tied to the URL. For example, if the URL https://www.myapp.com/users is meant to fetch all the users in the application, the routing program uses the route (usually the section after the domain of the application; which in this case is ‘/users’) to call the functionality that is tied to it. Now it is at this functionality layer that the MVC design pattern comes into use. The first of the layer in the MVC to be used is the Controller; it handles requests from the client and sends back response to the client after carrying out all its functionality. The controller makes use of the Model to Create, Read, Update and Delete records from the database. The Model is where all the interactions with the application database are done. After interaction with the Model, the Controller then prepares the View to be returned back as HTTP response. The View most times is the HTML document that is used to represent the data that is gotten from the database.

[1] listed the advantages of using this approach as follows:

- The application is divided into components.
- Because it is component-driven, it supports code reuse.
- It is a model or standard architecture that allows easy visualization of how the entire system works.
- Reusable and thoroughly tested code in the libraries, classes and functions.
- Applications built using MVC pattern are well structured and consistent.
- It supports abstraction, interoperability and code maintenance.

B. Design of Init Framework

Before the development of web frameworks, web applications were written traditionally from scratch using programming languages like PHP, JavaScript, Python, etc. but in recent times, many developers have preferred to use web frameworks like Laravel, Django, Flask, React etc. to develop web applications. These web frameworks are built on top of these programming languages.

1) *Design Concepts*

Here, we analyse different features of web programming that make building web applications with web frameworks better than building web applications traditionally.

a) *File Structure*

When starting a new PHP project, it can be discouraging trying to figure out the best way to organize the project. Knowing where to put the images or external libraries, as well as keeping the application logic separate from the layouts could get you confused. This is because there is no defined way to organize the file structure of an application. The proposed system – Init Framework would have a simple and standard file structure so that the resources such as images, CSS files, JavaScript files and the view templates are separated from the application logic files which are the controllers. It would further separate the database model files and the service provider files. The framework would have a storage folder to hold all the files that the users of the application can upload or download files from. A core folder would also hold the core classes of the framework. The proposed file structure would look like this:

- Resources
    - CSS
    - Images
    - JavaScript
    - Views
- Providers
    - Provider files
- Models
    - Model files
- Controllers
    - Controller files
- Core

Organizing the framework like this would help structure the web application around three physical tiers – client, application, and the database. The database manipulation would be handled by the models, the application contains business logic running on the controllers, and the client is the web browser that runs HTML generated by the application layer.

b) *Clean Uniform Resource Locators (Pretty URLs)*

Clean URLs, also known as RESTful URLs, user-friendly URLs, or search engine-friendly URLs are Uniform Resource Locators (URLs) intended to advance how websites and web services are being used and accessed by being immediately and intuitively meaningful to users [7]. Such URL structures often show the conceptual structure of the web application, therefore separating the user interface from the server's internal representation of files. This URL structure is not only common with web applications that were not built with frameworks, but in the proposed web framework – Init Framework. Clean URLs would be designed so that the user can define how the URLs of the application would look like. The framework would also be able to hold URL parameters.

An example of a conventional website URL is https://www.myapplication.com/users.php?role=customers; but with Init Framework, this URL would look like this – https://www.myapplication.com/users/customers.

In the former, the server's internal file structure is exposed to malicious users as they can tell that there is a file called “users.php” in the root directory of the application, but with the latter, malicious users cannot deduce the file structure of the server from the URL.

Additionally, clean URLs being implemented in Init Framework would improve search engine optimization (SEO), and would aid web applications in conforming to the representational state transfer (REST) style of software architecture. It will also ensure that individual web resources remain consistently at the same URL. Thus, this would make web applications a more stable and useful system, and allow for more durable and reliable bookmarking of web resources [7].

c) *Routing*

Routing is a prominent feature of web applications because it plays a central role in static HTML pages as well as in most complex web applications. Routing comes into play whenever an application uses URLs to navigate resources. The traditional way of web programming uses the physical address systems for navigation, which is like web-based routing with URLs [8]. A different routing system would be developed for the proposed web framework to interpret the URLs, since the framework would not be navigating the web with the physical address system. The framework's routing system will link a URL to a controller method. The controller is a class with several methods. All possible URLs are registered when developing the application so that when a request is made to the server, the request URL is matched against all the registered URLs. When a match for the URL is gotten, a new instance of the controller tied to the URL is created and the method tied to the URL is called. It is from this controller method that all actions (like performing database operations, uploading files, sending view templates to the client) would occur. Most frameworks that implement a routing system are most times MVC structured.

d) Model View Controller (MVC)

As stated earlier, the traditional method of web programming had the complexity of knowing how to arrange files and separate the application logic codes from the database models codes, as well as separating the resources from the service providers. The proposed framework would have a structured layer that separates the internal representations of information from the ways this information are being presented to the user. Traditional method of web programming never had a defined way of implementing this design concept, but the proposed framework would implement the Model-View-Controller (MVC) structure. MVC is a software design pattern that is commonly used for developing user interfaces which splits the related program logic into three interconnected elements. This concept was originally used for desktop graphical user interfaces (GUIs), but has now become popular for designing web applications [9].

e) Scaffolding

In programming, scaffolding is a source code that is incomplete and serves as a basis for further development. Scaffolding is a technique supported by some model–view–controller systems that allows the developer to decide how to use the application server [10]. Most software projects also share such file structures and specifications in order to simplify the development of such projects in accordance with these conventions; at the beginning of each project, "scaffolding" tools can generate pprogram files automatically. It would also be necessary for the proposed system to create new component files such as controllers and providers. This functionality is not available in the traditional creation of a web application. Even after the project was developed, Init Framework would also be able to create different application modules.

f) Object Oriented

Before the introduction of object-oriented programming, procedural programming was the custom. Procedural programming had its benefits, but it also had some disadvantages that the proposed framework would try to eliminate. [11] and [15] identifies some of the disadvantages of procedural programming as follows:

- Poor expression of complex abstractions
- Optimization and extension is more difficult
- It doesn't scale well to large systems.
- Complex processes lead to "spaghetti code"
- Hence, higher risk of errors when editing

Object-oriented programming, however, is a programming language template that organizes programs around data or objects rather than functions and logic. The proposed architecture would be built using the object-oriented model, including some of the object-oriented programming properties such as abstraction and inheritance. The object-oriented paradigm, according to [12] contradicts the historical approach to programming, where focus was put on how the logic was written rather than how to interpret the information within the logic.

g) View Directives

Traditionally, when developers are displaying data from the database with HTML, they often mix PHP database codes with HTML codes. This traditional way of web programming frequently increases difficulty when trying to debug programs; it also exposes the data from the database to the whole program. Init Framework would have a better way of representing the data using View Directives. These directives would act like syntactic sugar to the HTML codes. It would help the programmer to write conditional and iterative statements in HTML. All of these would be translated to pure HTML with the Template Engine that the framework would provide.

h) Authentication and Authorization

Authentication in web programming is when the server of an application needs to know exactly who is accessing the site or the information on the site. Typically, a server encryption requires a user name and password to be used. Encryption does not determine what tasks the individual can perform or what files they can see. Authorization in web programming is a way of determining if a user or client has permission to use a resource or access a file. Authorization is usually together with authentication so that the application knows the client that is requesting access [13]. Ordinarily, web resources and web pages do not require authentication or authorization before granting access to who is requesting them. Hence, developers have to write the authentication logic and the authorization logic for every application that is built in order to ensure some level of security. The proposed framework, however, would have to implement them so users do not have to write them again.

2) Model Of The Proposed Framework

a) Architecture Of Init Framework Application

Init Framework application is the web application that allows users to create a new project using the framework, and download it on their local machine. Fig. 2 shows the architecture of Init Framework application.

Figure 2. Architecture of Init Framework Application

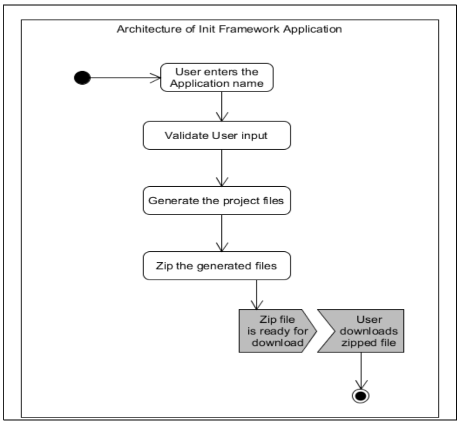
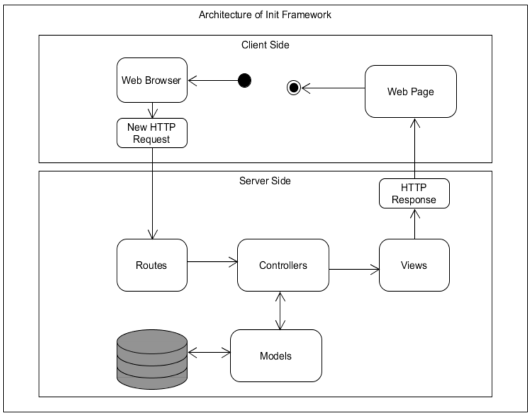

Architecture of Init Framework Application
User enters the Application name
Validate User input
Generate the project files
Zip the generated files
Zip file is ready for download
User downloads zipped file

Fig. 3 is an illustrative diagram of what happens when a request is made to the server; the Route class uses the set routes to match the request URL, then it calls the Controller method that is tied to the URL. The Controller method calls with the necessary Model(s) to get interact with the database, and then pass the data to the View to render the final page.

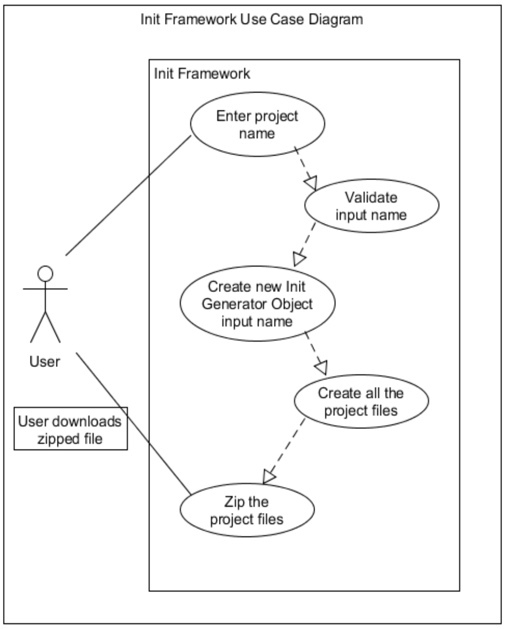
Figure 3. Architecture of Init Framework



b) Use Case Diagram

The use case diagram that provides a view of the behavior of the proposed system and subsystem is presented in Fig.4. It shows the actor's interaction with the system and illustrates the functionality of the system.

Figure 4. Use Case diagram of Init Framework


Init Framework Use Case Diagram
Init Framework
Enter project name
Validate input name
Create new Init Generator Object input name
Create all the project files
Zip the project files
User
User downloads zipped file

c) Class Diagram Of Init Framework

Fig. 5 depicts a class diagram of the Init framework. It is a type of a fixed structure diagram that defines the structure of a system by showing the classes, properties and methods in a system. It also shows the relationships among the objects.

d) Input Design

To create a new application or a new project using Init Framework, the only required input is for the user to give the name of the application, the framework then generates the required source codes for the user.

e) Output Design

When a user creates a new project using the framework, the framework will create all the required files in the earlier designed file structure. These files will be put in the application root directory, which would then be zipped and ready for download. The user then downloads the zipped file, moves the file to the required directory and unzips it there. Then the user can continue developing on the framework.

III. IMPLEMENTATION AND TESTING

A. Choice of Programming Tools

The following tools were used in developing the Init Framework:

1) UMLet

UMLet is an open-source Java-based Unified Modelling Language tool that was designed to teach Unified Modelling Language and for rapidly making UML drawings. Diagrams are made by dragging the UML components into the view section. It is rather a drawing tool than a modelling tool because there is no fundamental vocabulary or manual of reusable design objects. This software was used in the modelling of the Init framework during the design phase.

2) XAMPP

XAMPP is a free and open-source web server software stack package built by Apache Friends; the package consists mainly of the MariaDB database, Apache HTTP Server, and PHP and Perl programming language interpreters for scripts. XAMPP provided the necessary software needed for the server side development.

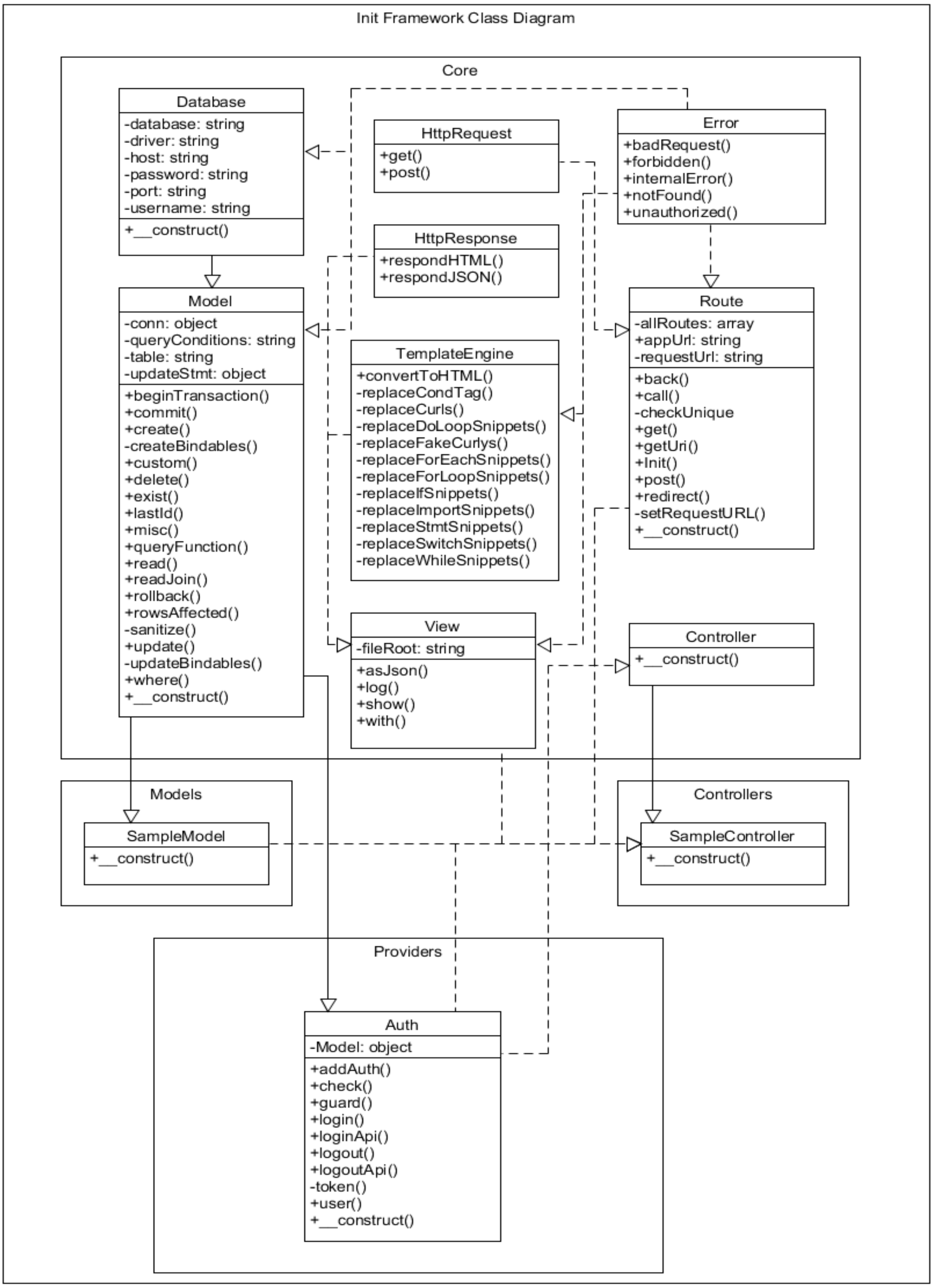
3) Google Chrome

Google Chrome is a web browser that runs on cross-platform. It was developed by Google and was first released for Microsoft Windows in 2008; it was later built for the Linux OS, macOS, iOS, and later for Android. The browser went on to become the main component on which Chrome OS is built upon, where it serves as the platform for most web applications. Init Framework Application was developed and tested on the Google Chrome browser.

4) Visual Studio Code

Visual Studio Software is a Microsoft-based Integrated Design Environment (IDE). It has been developed for Windows, Linux and macOS. The IDE supports debugging, smart software completion, Git and GitHub embedded command, code refactoring, and syntax and code snippets. All the codes that were developed for Init Framework were written using this code editor. Some of the reasons for choosing Visual Studio Code are its support for complex and large projects and its robust built-in Git support.

Figure 5. Class drawing of Init Framework



B. Framework Implementation

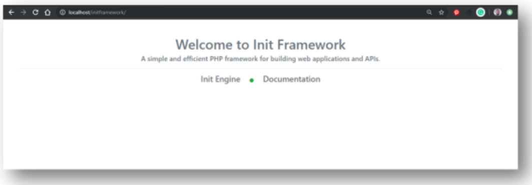
1) Home Screen

Fig. 6 is the first page that appears when the application is opened.

2) Init Engine

Fig. 7 is a snapshot of the page where the user creates a new application by entering the application name into the input box and clicking “Create New Application”.

Figure 6. Snapshot of the Home Screen



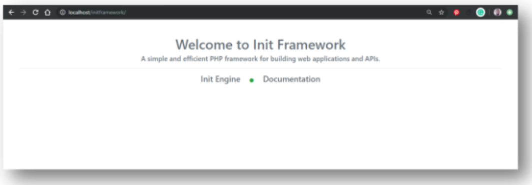
localhost/initframework/

Welcome to Init Framework

A simple and efficient PHP framework for building web applications and APIs.

Init Engine ● Documentation

Figure 7. Snapshot of the Init Engine page.



localhost/initframework/

Welcome to Init Framework

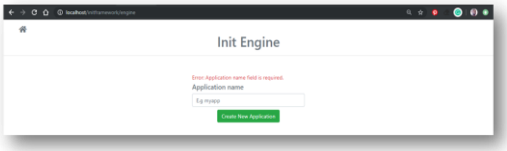
A simple and efficient PHP framework for building web applications and APIs.

Init Engine ● Documentation

3) *Error Messages*

Fig. 8 shows the error text highlighted in red indicating the required input field.

Figure 8. Snapshot of the Init Engine page with error message



localhost/initframework/engine

Init Engine

Error: Application name field is required.

Application name

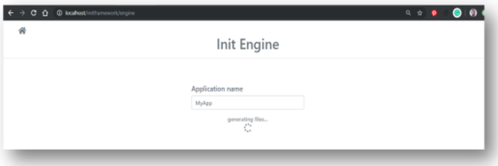
Eg myapp

Create New Application

4) *Generating Application Files*

The text "generating files" is an indication that the application is generating the user's application files as depicted in Fig. 9.

Figure 9. Snapshot of the Init Engine page when generating application files



localhost/initframework/engine

Init Engine

Application name

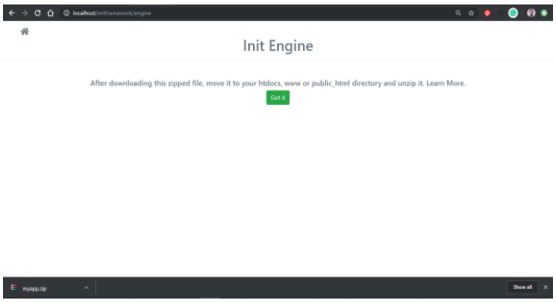
MyApp

generating files...

5) *Download Zipped File*

Fig. 10 is a snapshot that shows the downloaded zipped file and an instruction on the web page on how to set it up for usage.

Figure 10. Snapshot of the Init Engine page after downloading the new zipped application



localhost/initframework/engine

Init Engine

After downloading this zipped file, move it to your htdocs, www or public_html directory and unzip it. Learn More.

Got it

myapp.zip

Show all

C. *System Testing*

An essential step in creating a stable and error-free application is system testing. Once the source code has been created, the program must be checked before it is delivered to the consumer to discover and fix as many errors as possible.

1) *Testing Strategies*

The basic strategies used for testing the framework include:

a) *Acceptance Testing*

Acceptance testing is conducted to determine the acceptability of the software. Some of the aims of this test is to confirm that the software correctly implements system and software requirements and to assure the customer that the software is safe [16]. After developing the application, the application was tested to see that it met the design requirements.

b) *Regression Testing*

Regression Testing is used to ensure that updates made to a software doesn't have accidental side effects. Regression Testing is, in the real sense, a full or partial selection of already completed test cases that are re-executed to ensure that the existing system's functionalities are still working well [16].

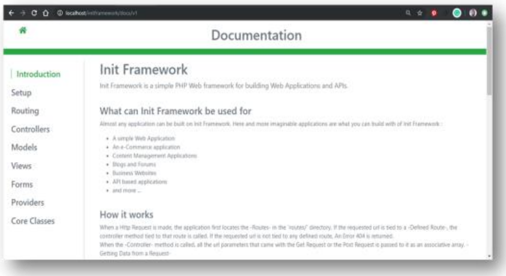
c) *Unit Testing*

Unit testing is used to confirm that each individual software units/components perform the capability assigned to it. A unit is the smallest part of any code that is testable. Usually, it has one or several outputs and usually a single output [16].

2) *User Documentation*

User documentation refers to the documentation provided to the end users for a product or service. The documentation for the user is designed to help end users use the product or service. This is frequently referred to as client assistance. The client report is a part of the customer's overall product [14]. The documentation for Init Framework is provided in the application to guide and assist users in the setup of their application and to provide (example) codes of how to use the methods provided by the Core classes. Fig. 11 shows a snapshot of the Init Framework's documentation page.

Figure 11. Snapshot of Init Framework



localhost/initframework/docs/v1

Documentation

Introduction
Setup
Routing
Controllers
Models
Views
Forms
Providers
Core Classes

Init Framework

Init Framework is a simple PHP Web framework for building Web Applications and APIs.

What can Init Framework be used for

Almost any application can be built on Init Framework. Here and more imaginable applications are what you can build with of Init Framework :

- A simple Web Application
- An e-Commerce application
- Content Management Applications
- Blogs and Forums
- Business Websites
- API based applications
- and more ...

How it works

When a Http Request is made, the application first locates the -Routes- in the 'routes/' directory. If the requested url is tied to a -Defined Route-, the controller method tied to that route is called. If the requested url is not tied to any defined route, An Error 404 is returned.
When the -Controller- method is called, all the url parameters that came with the Get Request or the Post Request is passed to it as an associative array. -Getting Data from a Request-

CONCLUSION

The study identified web frameworks as tools that help developers to create applications faster and easier. The features identified as required for the proposed framework was designed and developed. Though the web framework has been tested and used to develop several applications on the web, some issues that need to be addressed/corrected in subsequent release of the web framework have been identified. They include:

1. A service provider for hashing and encryption

2. A better Authentication Provider

3. A provider to curb the vulnerabilities following the OWASP standards. etc.

Init Framework is a back-end web framework and some of its advantages over other back-end web framework are listed in the Table I below.

TABLE I. COMPARISON BETWEEN INIT FRAMEWORK AND OTHER FRAMEWORKS

Frameworks	Comparisons
Express.js	Unlike Express.js that defaults to several call-backs, Init framework does not have multiple call-backs. Instead, it uses the string method of calling controller methods.
Spring	Unlike Spring whish have a steep lerning curve, Init Framework is easy to learn.
Ruby on Rails	Unlike Ruby on Rails where the number of gem dependencies affect the boot time of the application, Init Framework is not affected by the number of packages added by the developer.
Django	Unlike Django, Init Framework is not monolithic as developers can use micro-services in their applications.
Traditional web programming	Unlike traditional method of web programming, the Clean URL feature of Init Framework makes web applications more SEO friendly.

The framework is open sourced to the public so that developers can review and possibly suggest new features to be added, as the framework is also being used currently to develop web applications. The user only needs to make sure that the machine on which the framework is to operate on has been configured to the minimum hardware and software requirements.

REFERENCES

[1] Rashidah, I. Thouhedul, & A. Norashikin (2015). An Empirical Study of the Evolution of PHP MVC Framework. August, 2021 from https://www.researchgate.net/publication/312737643

[2] Software framework. (2021). Retrieved August, 2021, from https://en.wikipedia.org/wiki/Software_framework

[3] D. Rick (2017). Exploring frameworks: What do frameworks have to offer that programming languages lack? Retrieved August, 2021 from https://jaxenter.com/frameworks-vs-programming-languages-134954.html

[4] M. Jigar (2021). Detailed Comparison of Web Frameworks With Pros And Cons. Retrieved August, 2021 from https://www.monocubed.com/web-development-framework-comparison/

[5] I. Paul (2018). Handling complex MVC applications - How to scale and avoid Controller chaos. Retrieved August, 2021 from https://dev.to/pavlosisaris/handling-complex-mvc-applications----how-to-scale-and-avoid-controller-chaos-lb9

[6] C. Alex (n.d.). Creating a Basic Laravel 5 MVC Application in 10 Minutes. Retrieved August, 2021 from https://selftaughtcoders.com/from-idea-to-launch/lesson-17/laravel-5-mvc-application-in-10-minutes/

[7] Clean URL (2021). Retrieved August, 2021 from https://en.wikipedia.org/wiki/Clean_URL

[8] T. Mark (2017). Routing with React in Modern Front-End Web Applications Retrieved August, 2021 from https://freecontent.manning.com/routing-with-react-in-modern-front-end-web-applications/

[9] Model–view–controller. (2021). Retrieved August, 2021 from https://en.wikipedia.org/wiki/Model–view–controller

[10] Scaffolding (2021). Retrieved August, 2021 from https://en.wikipedia.org/wiki/Scaffold_(programming)

[11] Imperative programming: Overview of the oldest programming paradigm (2020). Retrieved August, 2021 from https://www.ionos.com/digitalguide/websites/web-development/imperative-programming/

[12] R. Magaret and L. Sarah (2021). Object-oriented programming (OOP) Retrieved August, 2021 from https://searchapparchitecture.techtarget.com/definition/object-oriented-programming-OOP

[13] Understanding Authentication, Authorization, and Encryption (n.d.). Retrieved August, 2021 from https://www.bu.edu/tech/about/security-resources/bestpractice/auth/

[14] User documentation. (2021). Retrieved August, 2021 from https://computersciencewiki.org/index.php/User_documentation

[15] Programming Paradigms. (n.d.). Retrieved August, 2021 from https://www.clear.rice.edu/comp310/course/design/prog_paradigms.html

[16] Haigh, D. Fred (2020). STP - Software Test Plan. Retrieved August, 2021 from https://swehb.nasa.gov/display/SWEHBVC/STP+-+Software+Test+Plan