ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

OSC

The 4th Open Society Conference

PROCEEDING BOOK

Perspectives and Impacts of Metaverse on Sustainable Development Goals

Online International Conference

Thursday, July 7, 2022



Faculty of Law, Social and Political Sciences
Universitas Terbuka - Indonesia

TABLE OF CONTENTS

FOREWORD ..... **i**
DEAN OF FACULTY OF LAW, SOCIAL AND POLITICAL SCIENCES REMARKS ..... **iii**
RECTOR REMARKS ..... **vi**
TABLE OF CONTENTS ..... **ix**

SUB-THEME: CYBER SECURITY CHALLENGES IN LAW PERSPECTIVE
Legal Analysis of The Criminal Act of Selling Personal Data Through the Deep Web
Mohammad Dery Rifani, Ivan Zairani Lisi, Rini Apriyani 1-9
The Existence of Digital Platforms and the Challenges in the Enforcement of Indonesian Competition Law
Rhido Jusmadi, L. Budi Kagramanto, M. Hadi Shubhan 10-24
Cyber Security: Legal Challenges and Opportunities in The Metaverse World
Dian Ratu Ayu Uswatun Khasanah, Yeni Santi, Megafury Apriandhini 25-32
Mechanism For Implementing Criminal Liability of Campaign Funds
Sukarno, Juajir Sumardi, Muhadar, Anwar Borahima 33-42
Criminal Threats in The Personal Data Protection Bill in Indonesia
Muhammad Rusli Arafat, Hevi Dwi Oktaviani 43-51
Cyber Security Challenges in Law Perspective "Challenges of Criminal Law Enforcement in Misuse of Social Media (Medsos)"
Rivaldhy N. Muhammad, Abdul Mutalib, Rahmat Hi. Abdullah 52-61
Digital Espionage in New Era of Cyber: Political and International Law Perspectives
Syarifurohmat Pratama Santoso, Richard Togaranta Ginting, Fransiska Timoria Samosir 62-66
Guarantee of Information and Communication Technology Application Security in Indonesia: (Regulations and Challenges?)
Nurfaika Ishak 67-74
Legal Policy of The Personal Data Protection Bill in Indonesia
Hevi Dwi Oktaviani, Muhammad Rusli Arafat 75-84
The Urgention of Personal Data Protection on Metaverse Era: a Potential Threat to Privacy and Security
Rina Elsa Rizkiana, Dian Ratu Ayu Uswatun Khasanah 85-96
The Legality of Marriage for Giving Dowry in The Form of Money in The Metaverse According to The Perspective of Islamic Law
Nur Sulistiyaningsih 97-104
Readiness Of the Government in Legal Regulations to Face the Metaverse Era
Yeni Santi, Dian Ratu Uswatun Hasanah 105-111
Potential of Money Laundering in the Metaverse Era
Nadia Nurani Isfarin, Sani Nur Imamy, A. Rachmat Wirawan 112-121

SUB-THEME: DIGITAL TRANSFORMATION AND THE CHANGING SOCIAL LIFESTYLE
Synchronous Method: Bimbingan Teknis Pelaksanaan "Take Home Exam" Universitas Terbuka
Steven Anthony 122-130

Transformation and Continuity: An Environmental Anthropology Study on the Digitalization Process by Kelompok Pengawas (POKMASWAS) Taman Kili-Kili Sea Turtle Conservation in Trenggalek
Basyarul Aziz 131-144
Android-Based Learning Video Innovation to Improve Digital Literacy as A Social Lifestyle in Education for Elementary School Teachers
Dewi Nilam Tyas, Sri Sulistyorini, Sri Sukasih, Desi Wulandari, Suparti 145-152
Mobile Based Exam Training Using Android Controlled by Website: A Prototype
Djoko Rahardjo 153-159
Komunikasi Manusia di Era Transformasi Digital: Literature Review
Windra Irawan 160-170
Eksistensi Pendidikan dan Pembelajaran IPS di Era Metaverse
Yulian Widya Saputra, Rahmadi, Made Ngurah Partha 171-177
Digitalisation of Shamanic Practice in Serving Remote Patients
Perdamaian Daulay, Momy A. Hunowu, Windra Irawan 178-188
Strategi Programming Acara "Si Otan" di Stasiun Tevevisi Trans 7
Vayol Parama Adrianjara, Susi Andrini 189-200
Kajian Sintesis Cashless Society di Indonesia
Dwi Rahmawati, Raden Aswin Rahadi 201-208
Website Development Using the Codeigniter Framework – A Literature Review
Erman Arif, Imelda Paulina Soko 209-216
Work or Study: Observing Informal Learning Among Employees and Teachers in Schools
Astrid Wangsagirindra Pudjastawa, Rina Kusweni 217-228

SUB-THEME: HUMAN CAPITAL DEVELOPMENT THROUGH METAVERSE
Challenges and Opportunities for Remote Working after COVID-19 Pandemic
Winin Maulidya Saffanah, Rizky Wahyu Permana 229-239
Patterns Of Social Interaction of Child-Assisted Citizens in Special Child Development Institutions
Randi, Eva Lidya, M. Fajar Ganta Saputra, Wahyu Vergiawan Lestanto, Khevin Wiriski 240-246
Reframing Individual's Bureaucratic Personality in Public Service Interaction in Digital Era
Yeni Widyastuti, Susanti 247-256

SUB-THEME: LIBRARY AND INFORMATION SYSTEM IN METAVERSE SOCIETY
Media Literate As A Determinant Of Student Performance In Communication Science Undergraduate Program Of Universitas Terbuka
Arifah Bintarti, Djoko Rahardjo 257-264
Library Paradigm and Challenges Towards The Era of Metaverse Society
Richard Togaranta Ginting 265-269
Digital Learning Ecosystem In Indonesian Open University Library
Dewi Maharani Rachmaningsih 270-282
De-Contextual Communication; A Transformation Model Of Library Services In The Metaverse Era
Sri Sediyaningsih, Mohammad Pandu Ristiono, Septian Dwi Cahyo 283-290

SUB-THEME: CREATIVE DIGITAL ECONOMY THROUGH THE WORLD OF METAVERSE

Capaian *Smart Economy* dan Tantangannya di Era Metaverse
Dedy Juniadi,Siti Aisyah, Zainur Hidayah 291-303
The Potential of Virtual Methods As A Means Of Ecotourism Education In The Era Of Society 5.0
Putu Nomy Yasintha, Richard Togaranta Ginting, I Dewa Ayu Putri Wirantari 304-311
The Effect Marketing Media Social On Competitive Advantage and Performance
Khoiriyah Trianti, Karina Utami Anastuti 312-334
Branding In Metaverse: Why Big Brand Advertising In Metaverse
Arsa Widitiarsa Utoyo, Tobias Warbung, Nunnun Bonafix, Hervina Diah Aprilia, Julianto Oei 335-342
Kube Development (Joint Business Group) In Jatiluwih Village, Bali Through Digital Marketing In The New Normal Era
I Dewa Ayu Putri Wirantari, Putu Nomy Yasintha, Richard Togaranta Ginting 343-348
Storytelling Marketing In The Digital Economy: Utilizing Burke's Pentad On Product's Stories
Natalia Widiasari 349-359

SUB-THEME: TREND AND DIRECTIONS OF TRANSLATION IN METAVERSE

Students' Ability in Assessing Translation Quality and Translating Figurative Language in English
Afriani 360-372

SUB-THEME: THE FUTURE GOVERNANCE IN THE METAVERSE WORLD

Improving City Plan Certificate Services Electorally (E KRK) Based On Digital City Spatial Planning
Dwi Putranto Riau 373-391
Analysis Of Transparency Information The Procurement Of Goods/Services Through The Media Website (E-Procurement) In Realizing Goods Governance In Polewali Mandar Regency
Miswar, Sudirman Karnay, Jeanny Maria Fatimah 392-400
The Future Of Cities In The Metaverse Era: Are Indonesian Cities Ready?
Bani Pamungkas 401-418
The Model Of Government Communication In A Level Of Homogen Community In Era The Government 4.0.
Susanti 419-423

Website Development Using the Codeigniter Framework – A Literature Review

Erman Arif[1], Imelda Paulina Soko[2]
[1,2]Universitas Terbuka
e-mail: erman.arif@ecampus.ut.ac.id

Article Info

Article history:

Received May 17th, 2022
Accepted June 05th, 2022
Published Nov 17th, 2022

Article of OSC 2022

Abstract

CodeIgniter is a framework created using the PHP language, which can be used for fast web development. The framework itself can be interpreted as a structure of libraries, classes, and run-time infrastructure that can be used by programmers to develop web applications quickly. The purpose of using the framework is to make it easier for web developers to develop robust web applications quickly without losing flexibility. Based on the results of the theoretical study, it is concluded that CodeIgniter applies a very dynamic and easy-to-use MVC (Model, View, Controller) model to create websites. Compared to other frameworks, CodeIgniter is more flexible, giving web developers the freedom to develop web-based applications even without a framework.

***Keywords**: Codeigniter, Framework, Website Development*

Copyright © 2022

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

INTRODUCTION

Current technological developments, including those related to the dissemination of information using internet media, have reached every region to remote areas. Advances in existing internet technology can expand information through cyberspace and advances in internet technology as easily as people access information. This progress also marks the progress in the development of the website as a medium of access for various purposes developed by the developer. This condition has led to so many developers building various website applications for the wider community to use to develop websites as needed. Various web applications develop a framework, which is a collection of basic commands or functions that form certain rules and interact with each other. so that in making website applications it makes it easier for developers to make web easily and quickly (Wardana, 2010). The use of the framework aims to accelerate and facilitate the development of a web application, facilitate the maintenance process and the framework provides facilities that are commonly used so that we do not need to build from scratch, and are freer in development when compared to CMS (Arrhioui, et al., 2017).

Until now there are various frameworks that have been developed and can be used to develop the web such as CodeIgniter. CodeIgniter is an application development toolkit framework for people who build websites using the PHP cake, symphony, yii, zend and prado programming languages. CodeIgniter applies a very dynamic MVC (Model, View, Controller) model to create websites (Betari, et al., 2017). This model was created to separate the code generation section from creating templates for presentation purposes (views) so that website developers can jointly develop websites focusing on their respective parts so as to speed up the website development process. This article will try to discuss about the CodeIgniter framework, its advantages.

METHODOLOGY

The method used by the researcher is a literature review method which is carried out by browsing various reference libraries in the form of books, and articles/journals related to CodeIgniter. This study is carried out through the collection of scientific data or articles intended for research subjects or data collection that is librarian or studies conducted to solve basic problems or provide comprehensive information related to an issue or theme so that readers can understand the existing theme. This article is directed towards an important and thorough study of the relevant documents from the available literature. The library source database used is Google Scholar and Research gate for the study of CodeIgniter framework implementation research, while theoretical concepts are explored from various other sources.

RESULTS AND DISCUSSION

CodeIgniter is an application development toolkit framework for user who build websites using the PHP programming language which was first developed in 2006 by Rick Ellis. The goal is to allow developers to develop projects faster than they would if they were coded from scratch, by providing a rich collection of libraries for common tasks required, as well as simple, structured logic for accessing the libraries. CodeIgniter allows developers to focus creatively on the project by minimizing the amount of code required for a given task (Vuksanovic & Sudarevic, 2011). As much as possible, CodeIgniter is made as flexible as possible, allowing developers to work the way they want without having to work in a certain way (www.codeigniter.com).

CodeIgniter applies a very dynamic MVC (Model, View, Controller) model to create websites. This model was created to separate the code generation section from creating

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

templates for presentation purposes (views) so that website developers can jointly develop websites focusing on their respective parts so as to speed up the website development process. The model section describes a php file placed in the application/models directory which is used to retrieve data or information in the database in the form of a text file or xml file. This model contains classes and functions for retrieving, updating, and deleting site data. This model will usually handle SQL queries. The model is specifically used to connect to the database, so the programming logic in the model must also be linked to the database. For example, select condition but to select the query to be executed. The View display section describes the information displayed to website visitors. A view is simply a view commonly known as HTML. Then the controller is the link between the model and the view. Inside the controller there are classes and functions that handle requests from views into the data structure model. Also, the controller must not contain any code to access the database. The controller's job is to provide various variables to display in the view, call the model to access the database, provide error handling, perform application logic processing, and validate or validate, verify entries. So, in a nutshell, the order of requests is as follows: user is associated with a view, where all information is displayed in this view. When the user makes a request or request, for example clicking a button, the request will be handled by the controller. What you have to do, what data you want, whether you want to view the data, import it, or maybe do some pre-validation, are all handled by the controller. The controller will then ask the model to complete the request, be it a query or something else. From the model, the data will be returned for further processing in the controller and the new data from the controller will be displayed in the view. The CodeIgniter framework workflow can be seen in Figure 1. the request will be handled by the controller. What you have to do, what data you want, whether you want to view the data, import it, or maybe do some pre-validation, are all handled by the controller. The controller will then ask the model to complete the request, be it a query or something else. From the model, the data will be returned for further processing in the controller and the new data from the controller will be displayed in the view. The CodeIgniter framework workflow can be seen in Figure 1. the request will be handled by the controller. What you have to do, what data you want, whether you want to view the data, import it, or maybe do some pre-validation, are all handled by the controller. The controller will then ask the model to complete the request, be it a query or something else. From the model, the data will be returned for further processing in the controller and the new data from the controller will be displayed in the view. The CodeIgniter framework workflow can be seen in Figure 1. From the model, the data will be returned for further processing in the controller and the new data from the controller will be displayed in the view. The CodeIgniter framework workflow can be seen in Figure 1. From the model, the data will be returned for further processing in the controller and the new data from the controller will be displayed in the view.

The CodeIgniter framework workflow can be seen in Figure 1.

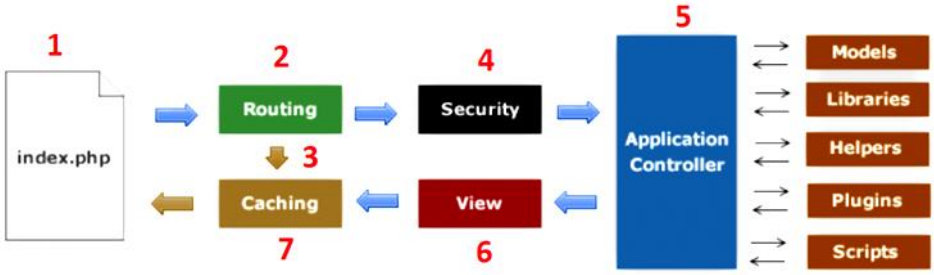


Figure 1. Codeigniter Framework Workflow

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

In Figure 1 it can be seen that Index.php functions as the first file in the program to be read by the program, then the Router will check the HTTP request to determine what the program should do, if the program already has a cache file, the file will sent directly to the browser. This cache file can make a website open faster. The cache file can skip the actual process that must be done by the CodeIgniter program. Before the controller file is loaded as a whole, HTTP requests and data sent by the user will be filtered first through the security facilities owned by CodeIgniter, after that the Controller will open the model file, core libraries , helpers and all the resources needed in the program, the last thing it will do is read all the programs in the view file and send it to the browser for viewing. If an existing view file is cached, the new uncached view file will update the existing view file.

CodeIgniter has several advantages, namely: a) it is open source, b) has a small size compared to other frameworks, c) applications made using CodeIgniter can run fast, d) CodeIgniter uses the MVC (Model-View-Controller) design pattern so that one file doesn't contain much code. This makes the code easier to read, understand and maintain in the future, e) CodeIgniter can be extended as needed; f) CodeIgniter is well documented, information about libraries and functions provided by CodeIgniter can be obtained through the documentation included in the distribution package, g) CodeIgniter has complete libraries and helpers, h) CodeIgniter has reliable security such as xss filtering, session encryption, etc.,

The CodeIgniter installation process is carried out in several stages, starting from downloading CodeIgniter via www.CodeIgniter.com then setting up a we server such as WAMPSERVER, MAMP, or XAMPP as the basis for CodeIgniter to run. In this case, the web server as a medium to put CodeIgniter ready to be installed via localhost. After the installation process ends, proceed with basic configuration such as autoload.php, this file is used to set functions that will be automatically loaded at the beginning when the program is run; config.php, in this file there are several configurations that have been configured by default; database.php, the database.php file is used to perform configurations related to the database configuration of the website to be created, as for the configuration that needs to be considered, namely: hostname,

Website development using the CodeIgniter Framework has been carried out by many researchers for various needs such as institutional/company information systems, libraries, e-learning, online stores, and so on. Afuan (2010) developed an information system for data collection of practical work reports for students of the Unsoed Informatics Engineering Study Program. In designing the data flow in this information system the author uses the Data Flow Diagram (DFD) method.

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

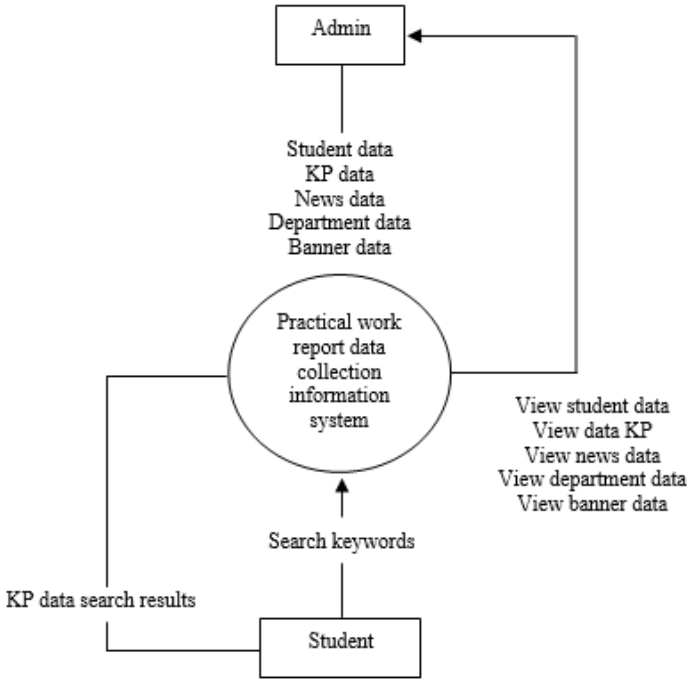


Figure 2. Data Flow Diagrams
Source: Afuan, 2010

This DFD context diagram describes the system in general (Figure 2). This system involves admin and student entities. Each entity provides input in the form of the required data into the system. The system will provide output in the form of data required by admin and students. The data needed in this system is represented in several tables, each of which is connected or related. Wibisono (2012) developed the development of SMS Gateway.

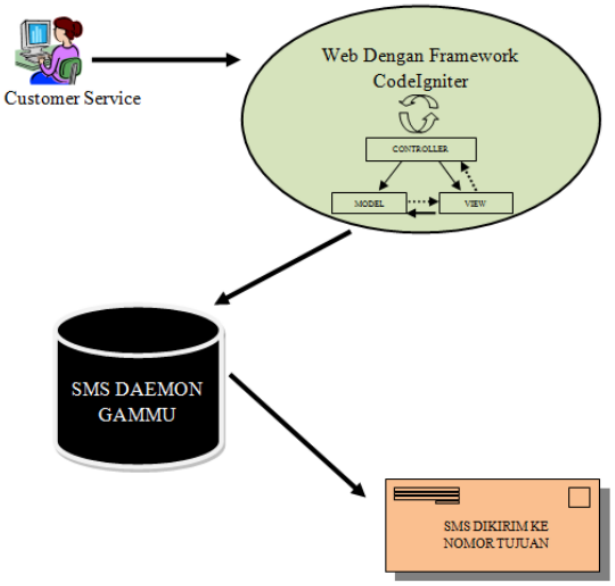


Figure 3. Design Architecture
Source: Wibisono, 2012

The development of this system is based on the old system, namely the service recording information system on the Bali iPhone which was developed so that the system can

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

automatically send short messages to its customers when the repair of the customer's cellphone or Apple product has been completed. This new feature can send short messages to many of its customers as a means of promotion. The architecture developed for this purpose is presented in Figure 3.

The user, in this case the customer service inputs short message data through a website that already uses the CodeIgniter framework, the web is displayed by calling the controller file, the controller file will process what will be displayed to the user (the controller will call the view), after the user enters the sms data, the view will send the data to the controller. The controller will send the sms data to the model which will take care of the database. Muzakir (2014) developed a library management website using the CI framework. U The goal is to facilitate service and access to information and management of library data, among others, facilitate the search for information on book collections, member information, registration, visitors, borrowing and returning books, and reports. clear picture of what needs to be done. System design can be divided into two parts, namely general system design and detailed system design. The general system architecture is presented in Figure 4. U The goal is to facilitate service and access to information and management of library data, among others, facilitate the search for information on book collections, member information, registration, visitors, borrowing and returning books, and reports. clear picture of what needs to be done. System design can be divided into two parts, namely general system design and detailed system design. The general system architecture is presented in Figure 4. U The goal is to facilitate service and access to information and management of library data, among others, facilitate the search for information on book collections, member information, registration, visitors, borrowing and returning books, and reports. clear picture of what needs to be done. System design can be divided into two parts, namely general system design and detailed system design. The general system architecture is presented in Figure 4.

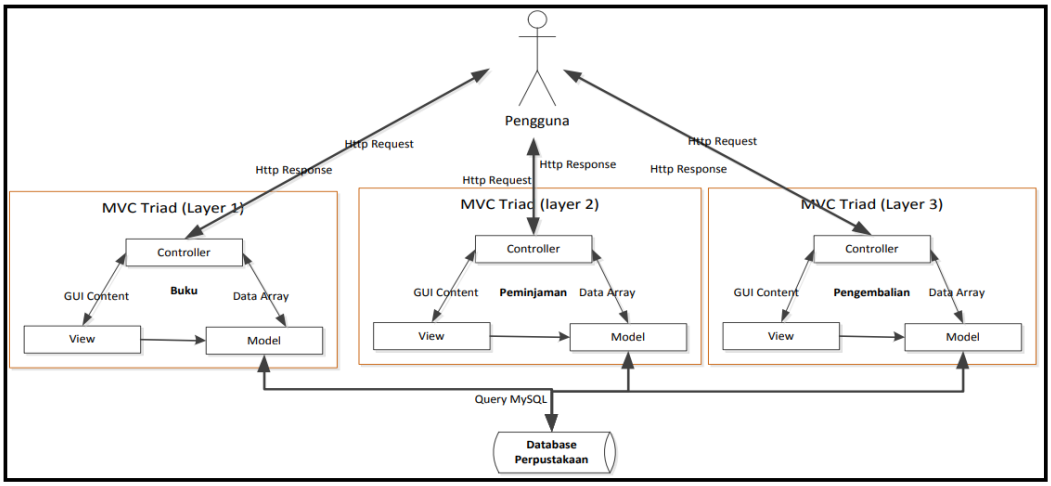


Figure 4. General Design Architecture
Source: Muzakir, 2014

In this library management system, there are three levels of user access rights, namely admin level, officer level, and student level. At the admin level, all the facilities in this system can be used by admin level users. Then at the officer level, officers can only manage borrowing and returning books, member management, and report management. While at the student level, there are several menus, namely ebook download menus, book lists, borrowing information, and book return information. Silviana & Thalib (2018) developed a website as a forum for sharing journals using the CodeIgniter framework. The purpose of this development is to create

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

a forum or place to publish journals for writers and also as a place for audiences to share information by publishing their own journals and to make it easier for the general public who want to search for journals to search only on one website. The navigation structure developed is presented in Figure 5.

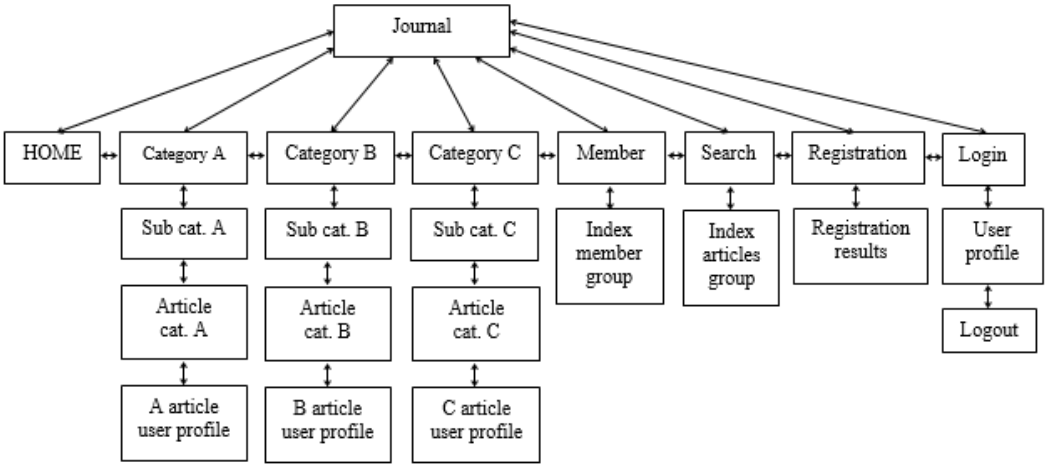


Figure 5. Navigation Structure
Source: Silviana & Talib, 2018

In the database design, there are 11 tables, namely user, admin, article, inbox, friend, type, favorite, category, comment, report, and comment_report tables. The design of the interface page of this application consists of the main page, category page, news page, login page, dashboard page, user profile page, article writing page, activity page, friendship page, inbox page, biodata settings page, article settings page, page collection of favorites, and the index page is the page from the search results through the search box.

CONCLUSION

CodeIgniter applies a very dynamic MVC (Model, View, Controller) model to create websites (Irish, 2009). The model describes a php file placed in the application/models directory which is used to retrieve data or information in the database in the form of a text file or xml file. View describes the information displayed to website visitors. The controller is the link between the model and the view. Inside the controller there are classes and functions that handle requests from views into the data structure model. Also, the controller's job is to provide various variables to display in the view, call the model to access the database, provide error handling, perform application logic processing, and validate or validate, verify entries. In general, CodeIgniter is very easy to use in building the web because it has a small size, MVC design pattern (Model, View, Controller) so that one file does not contain too much code, it is flexible so that it gives web developers the freedom to develop web-based applications even without a framework. The use of the CodeIgniter framework has been widely used, including building websites for various information systems, libraries, online stores, journals, and so on.

REFERENCES

Afuan, L. (2010). Pemanfaatan framework CodeIgniter dalam pengembangan system informasi pendataan laporan kerja praktek mahasiswa program studi teknik informatika Unsoed. *Jurnal Informatika*, 1(2), 39-44.

ISBN: 978-602-392-329-8
e-ISBN: 978-602-392-330-4

Arrhioui, K., Mbarki, S., Betari, O., Roubi, S., Erramdani, M. (2017). A model driven approach for modeling and generating PHP codeIgniter based applications. Transactions on *Machine Learning and Artificial Intelligence*, 5(4), 259-266.

Betari, O., Erramdani, M., Roubi, S., Arrhioui, K. & Mbarki, S. (2017*). Model transformations in the MOF meta-modeling architecture: from UML to CodeIgniter PHP framework*. In: *Europe and MENA Cooperation Advances in Information and Communication Technologies*, 227–234.

CodeIgniter documentation website. http://CodeIgniter.com/

Irish, H. (2009). *Creating a file hosting site with CodeIgniter*.

Muzakir, A. A. (2014). Implementasi manajemen perpustakaan menggunakan framework CodeIgniter (CI) dengan teknik hierarchical model–view–controller (HMVC). In *Seminar Nasional Sains dan Teknologi Informasi (SeNASTI). Pusat Penelitian dan Pengabdian Masyarakat STMIK Kharisma Makassar*.

Setiadi, M. F. (2018). *Tutorial CodeIgniter untuk pemula*. Diakses dari https://mfikri.com/artikel/tutorial-CodeIgniter

Silviana, A. B., & Thalib, F. (2018). *Pengembangan situs web sebagai wadah berbagi jurnal menggunakan framework CodeIgniter*. Skripsi, Teknik Informatika, Universitas Gunadarma, Depok, Indonesia.

Vuksanovic, I. P. & Sudarevic, B. (2011). Use of web application frameworks in the development of small applications. *Prosiding MIPRO, the 34th International Convention*.

Wardana. (2010). *Menjadi master PHP dengan framework CodeIgniter*. Jakarta: PT Elex Media Komputindo.

Wibisono, S. B. (2012). Pengembangan website menggunakan framework CodeIgniter sebagai penunjang sistem informasi SMS gateway di iphone Bali. *(JELIKU) Jurnal Elektronik Ilmu Komputer Udayana*, 1(1)

Faculty of Law, Social and Political Sciences
Universitas Terbuka - Indonesia
(FHISIP-UT)

1. Departement of Public Administration
2. Departement of Government Studies
3. Departement of Business Administration
4. Departement of Taxation
5. Departement of Communication Science
6. Departement of Library Science
7. Departement of Archival Science
8. Departement of Sociology
9. Departement of Law
10. Departement of English Literature and Translation Studies

UNIVERSITAS TERBUKA

OSC SECRETARIATE

Faculty of Law, Social, and Political Sciences Universitas Terbuka
Tel. +6221-7490941 ext. 1923 (Sarah) ext. 1945 (Megafury)
Website: http://osc.fhisip.ut.ac.id
Email: osc.fhisip@ecampus.ut.ac.id