Performance with Eloquent and Query Builder in Crowdfunding System with Laravel Framework

Putu Adi Guna Permana
System Information, Institute of Technology and Bussiness STIKOM BALI
Email: putuadi_guna@stikom-bali.ac.id

Evi Triandini
System Information, Institute of Technology and Bussiness STIKOM BALI
Email: evi@stikom-bali.ac.id

Abstract: Performance is a point of interest that is quite interesting for application owners, how could it not, besides being rich in features and following what they want, there are things that are no less important, namely the issue of the speed of use of an application that must be considered for developers. Because the speed level will affect the user experience in using it. Many factors influence the performance of an application, especially in the website category, one of which is the developer's ability to minimize large amounts of data load. This is the importance of being able to categorize which large data loads need to be considered and which are not. There is a system crowdfunding website that is currently operating using the laravel framework, but there are several obstacles faced where some of the processes in it are rather slow in the process. In this research, how do we choose which process uses process eloquence and which one needs to use the query builder. So that combining eloquent and query builder can be optimal

Index Terms: Eloquent, Query Builder, Crowdfunding, Performance.

1. Introduction

Business development from time to time is growing. This must also be supported by technology as a support for advancing business. Currently, offline shops or businesses are less developed than online, this is because online marketing covers a wide market. For example, in the past, if you wanted to shop for clothes, credit, food, you had to visit a certain store or outlet, but now many people prefer not to leave the house and choose to use the marketplace to shop for clothes, credit or food. With this, it is necessary to increase trade marketing via the internet.

Many people want to have a business with relatively small capital because the risk is also small. Currently, there is a business model with the concept of a joint venture/cooperation with many people where the capital required is relatively small as needed. Crowdfunding is a business model with the collection of funds from several people where the funds will later be used as capital to build a business that will be created [6]. The business that is being funded will be owned by the multiple people who fund the venture. However, there is some inconvenience that users feel when they access the site with a slightly slow/long process. From this, the author wants to research performance data used by one of the crowdfunding sites which uses a combination of the Eloquent and Query Builder methods in data processing [11]. This will also make users feel comfortable when using site access quickly and efficiently. With the combination of the eloquent method and the query builder, it is hoped that the problem of access speed in the crowdfunding application can be resolved and users can be more comfortable in accessing the website. In the future, application developers will be able to choose more appropriate and efficient methods to use.

2. Related Research

2.1 CrowdFunding

The problem that often becomes a stumbling block when developing a business is the source of funding or capital. One of the most common sources of funding for companies that want to expand their business is in the form of loans, either through bank loans or loans from other parties. However, the source of funding in the form of loans is not necessarily suitable for startups and SME types, because generally such businesses have limited cash flow or income to be able to pay for loan principal and interest payments soon. Besides, generally, startup companies and SMEs do not

have sufficient assets to be used as collateral.

With the current development of financial technology, startups and SMEs can raise funds without being burdened by the obligation to pay interest and principal investments, using the equity crowdfunding system. In the equity crowdfunding scheme, entrepreneurs are also not obliged to provide collateral to get funding. With equity crowdfunding, a fundraising company only needs to offer a share of shares in the company as compensation for the investment given [9]. That way investors will get part of the company's ownership and receive the company's profits according to the number of shares they own.

If you are unfamiliar with the term crowdfunding, crowdfunding or fundraising is the process of raising a small amount of money for a project or venture by a large number of people, usually done through an online platform. Crowdfunding itself consists of three types based on the form of rewards given to the funder, namely reward-based / donation-based crowdfunding, equity-based crowdfunding, and loan-based crowdfunding.

Reward-based crowdfunding is a form of crowdfunding based on prizes or rewards. Reward-based crowdfunding will reward investors, usually in the form of a first version or a limited edition of the product or service being funded. The rewards given are generally adjusted to the number of funds given, the bigger the funds, the more exclusive the rewards that investors get. Meanwhile, donation-based crowdfunding is a form of crowdfunding without reward and is usually related to a philanthropic activity [4]. Examples of reward-based crowdfunding platforms are Kickstarter and Indiegogo, while Kitabisa is an example of a donation-based crowdfunding platform.

The second type, loan-based crowdfunding or crowdlending, is a form of raising funds in the form of debt. Loan-based crowdfunding itself has grown rapidly and has been put into a separate category so that it is now better known as Peer-to-Peer (P2P) Lending.

The third type, equity crowdfunding, is a form of crowdfunding where the fundraiser will reward the crowd with shares. The size of the shares varies depending on the offer made by the fundraising company based on the valuation of the company.

2.2 Eloquent

Eloquent is one of the interesting features that Laravel has, with it we can interact and manipulate the database as desired. Each Eloquent model that has been created is a table representation of the existing database [10].

The Query Scope allows you to create a query function that can be reused throughout the application you are building. The scope has two parts, namely, Global Scope and Local Scope [4]. These two things of course can help you to minimize writing the same thing over and over again so that you can maintain it easier and look tidier.

Then what is the difference between the two? As the name implies, if the Global Scope will run the defined function every time the Model is used. Whereas Local scope will only run when the defined function name is written.

2.3 Query Builder

Query Builder is a class in Laravel for working with Databases [12]. With the Query Builder, we can perform Insert, Select, Update & Delete commands with minimal queries. With this Query Builder, we can also use PL SQL in data that uses large data loads.

2.4 Framework Laravel

Laravel is a PHP framework that emphasizes simplicity and flexibility in its design. Just like other frameworks, Laravel is built based on MVC (Model-View-Controller) [5]. Laravel includes a command-line tool called "Artisan" which can be used for bundle packaging and bundle installation. According to a survey conducted by Sitepoint.com in December 2013 regarding the popularity of the PHP framework, Laravel was in the top rank, thus making Laravel the best PHP framework for 2014 [1]. Currently, Laravel is a framework with an up-to-date version of PHP, because Laravel hints at PHP version 5.3 and above.

Reasons to use Laravel over other frameworks:

1. Simple coding,
2. There is a superior and convenient generator, Artisan CLI,
3. Schema Builder feature for various databases,
4. Migration & Seeding features for various databases,
5. Query Builder Features,
6. Eloquent ORM (Object Relational Mapper),
7. Package and bundle creation features

Laravel is a PHP-based framework created using the MVC approach. Laravel has many advantages, one of which is fast, easy to use, and has complete documentation.

2.5 MySQL

MySQL is an implementation of the relational database management system (RDBMS) which is distributed free of charge [7]. Each user can freely use MySQL, but with limitations the software cannot be made a commercial derivative product. MySQL is actually a derivative of one of the main concepts in existing databases; SQL (Structured Query Language) [3]. SQL is a database operation concept, especially for selecting or selecting and entering data, which allows data operations to be done easily automatically. The reliability of a database system (DBMS) can be seen from the way the optimizer works in processing SQL commands made by users and application programs that use it. As a database server, MySQL supports both transactional database operations and non-transactional database operations. In the non-transactional mode of operation, MySQL can be said to be superior in terms of performance compared to other competitor database server software. However, in non-transactional mode there is no guarantee of the reliability of the stored data, therefore non-transactional mode is only suitable for types of applications that do not require data reliability such as web-based blogging (wordpress) applications, CMS, and the like. For system requirements aimed at business, it is highly recommended to use transactional database mode, it's just that as a consequence MySQL's performance in transactional mode is not as fast as that in non-transactional mode.

MySQL Privileges:

1. Portability. MySQL can run stably on various operating systems such as Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, and many more.
2. Open source software (open source). MySQL is distributed as open source so it can be used free of charge.
3. Multi-user. MySQL can be used by several users at the same time without experiencing problems or conflicts.
4. Performance tuning, MySQL has amazing speed in handling simple queries, in other words it can process more SQL per unit time.
5. Various data types. MySQL has a very rich variety of data types, such as signed / unsigned integer, float, double, char, text, date, timestamp, and others.
6. Commands and Functions. MySQL has full operators and functions that support the Select and Where commands in the command (query).
7. Security. MySQL has several layers of security such as an encrypted password.
8. Scalability and Restrictions. MySQL is capable of handling large-scale databases, with more than 50 million records and 60 thousand tables and 5 billion rows. In addition, the index limit that can be accommodated is up to 32 indexes in each table.
9. Connectivity. MySQL can connect with clients using TCP / IP protocols, Unix sockets (UNIX), or named pipes (NT).
10. Localization. MySQL can detect error messages on clients in more than twenty languages. However, Indonesian is not included in it.
11. Interface. MySQL has an interface for various applications and programming languages using the API (Application Programming Interface) function.
12. Clients and Equipment. MySQL is equipped with various tools that can be used for database administration, and each tool includes an online guide.
13. Table structure. MySQL has a table structure that is more flexible in handling ALTER TABLE, compared to other databases such as PostgreSQL or Oracle.

3. Proposed Model

The conceptual framework of this study will focus on the user side (User). Selecting the Role User for testing the comparison of the query builder with eloquent intends to make the user more comfortable when quickly opening the desired menu access. In this research, both methods will be used for each module that will be tested respectively. The Research Team will test using the query builder first by entering the access load calculation script in the page being tested, then the query is replaced using the eloquent method as a comparison. The results of this comparison will be used as a test material when we use the query builder and when to use eloquent. The development method used in this research is the classic Systems Development Life Cycle method which is commonly called the waterfall method. SDLC (Systems Development Life Cycle).
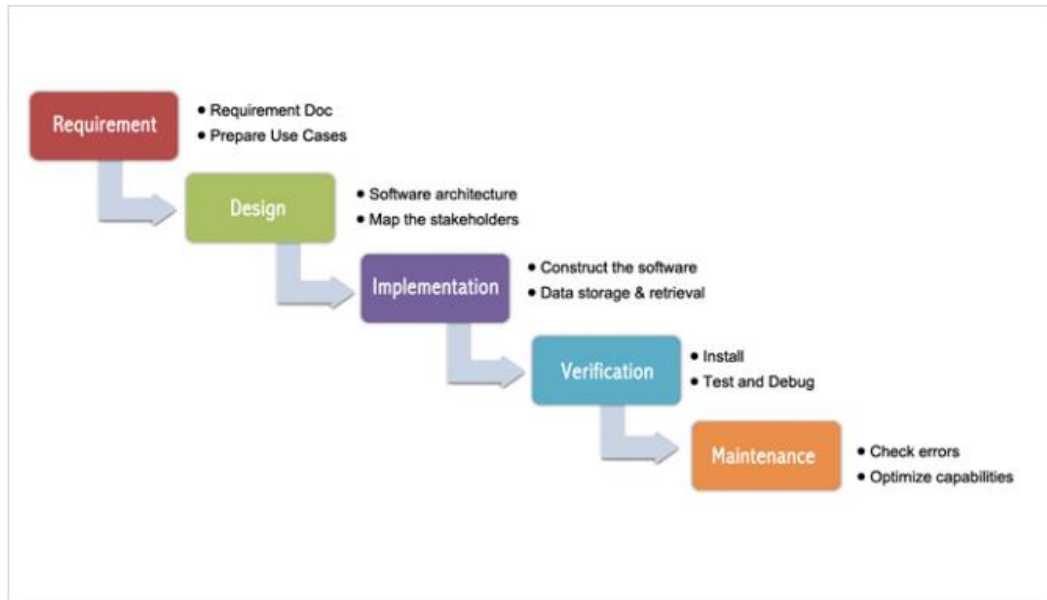
Fig. 1. SDLC

4. Result in Analysis and Discussion

From the analysis method that has been implemented, it can be seen the results of the analysis for the comparison of the query builder method with the eloquent that has been adjusted to the needs and circumstances. Here is a further explanation.

1. Analysis of Input Needs

Data input in this system is carried out by the user (community), where the process of accessing the page will be tested

2. Process Requirements Analysis

The process requirements of this comparison of methods include:
a. Time Calculation Script posted on the page.
b. Query both methods for each page that is tested.
c. User account access
d. Access Database

3. Output Needs Analysis

The resulting information, namely: the performance comparison using the query builder and eloquent methods in each module being tested.

After the analysis and design process has been carried out, the next step to take is the implementation of the comparison of the two methods on each page which is the focus of the trial. In this trial, the researcher inserted the script used for calculating the system load time, where the script is as follows:

```php
<?php
$starttime = explode(' ', microtime());
$starttime = $starttime[1] + $starttime[0];
$load = microtime();
$loadtime = explode(' ', microtime());
$loadtime = $loadtime[0]+$loadtime[1]-$starttime;echo "Waktu yang diperlukan : ".number_format($load,2)." detik";
?>
```

A. Trial of Daily Report Pages

This page is used to view daily reports of the sale of investment products. On this page, we try to use both the query builder and eloquent methods. Here are the results :

Code Query Builder :

```
$dailys = DB::table('daily_sales_report')
    ->leftJoin(user', 'daily_sales_report.user_id','user.id')
    ->leftJoin('products','trx_product_purchases.product_id','products.id')
    ->leftJoin('products_details', 'daily_sales_report.product_id','products.id')
```

```
->where('sales_date','>=',$firstDatePeriod->toDateString())
->where('sales_date','<=',$lastDatePeriod->toDateString())
->orderBy('sales_date', 'DESC')
->select('daily_sales_report.id','daily_sales_report.trx_date','products.product_name','products.ec_rate', 'products_details.omset','products.status') ->get();
```


Fig. 2. Daily Report (Query Builder)

Code Eloquent :

```
$dailys = DailySalesReport::with(['user','product',' 'products_details''])
    ->where('product_id',$product_id)
    ->where('sales_date','>=',$firstDatePeriod->toDateString())
    ->where('sales_date','<=',$lastDatePeriod->toDateString())
    ->orderBy('sales_date', 'DESC');
```


Fig.3. Daily Report (Eloquent)

B. Ticketing Page Trial

This page is used to generate reports on issues that users encounter in the crowdfunding system. On this page, we try to use both the query builder and eloquent methods. Here are the results :

Code Query Builder :

```
$ticket = DB::table('tickets')
    ->leftJoin(user', 'tickets.user_id','user.id')
    ->leftJoin('tickets_detail','tickets_detail.ticket_id','tickets.id')
    ->orderBy('ticket_date', 'DESC')
    ->select('tickets.id','tickets.trx_date','tickets.title','tickets.layanan','tickets.prioritas','tickets.status', 'tickets_details.chat') ->get();
```


Fig.4. Ticketing (Query Builder)

Code Eloquent :

```
$ticket = Tickets::with(['user' , 'tickets_details'])
    ->where('ticket_id',$ticket_id)
    ->orderBy('ticket_date', 'DESC');
```


Fig.5. Ticketing (Eloquent)

C. Trial of Ask Admin Pages

This page is used to ask questions related to products that have been invested. On this page, we try to use both the query builder and eloquent methods. Here are the results :

Code Query Builder :

```
$msgs = DB::table('tanya_admin')
    ->leftJoin('user', 'tanya_admin.user_id','user.id')
    ->leftJoin('details','details. tanya_admin_id',' tanya_admin.id')
    ->leftJoin('products',' tanya_admin.product_id','products.id')
    ->orderBy('tx_date', 'DESC') ->select('tanya_admin.id','tanya_admin.tx_date','tanya_admin.title','details.chat','details.user', 'products.products_name',
    'tanya_admin.status') ->get();
```
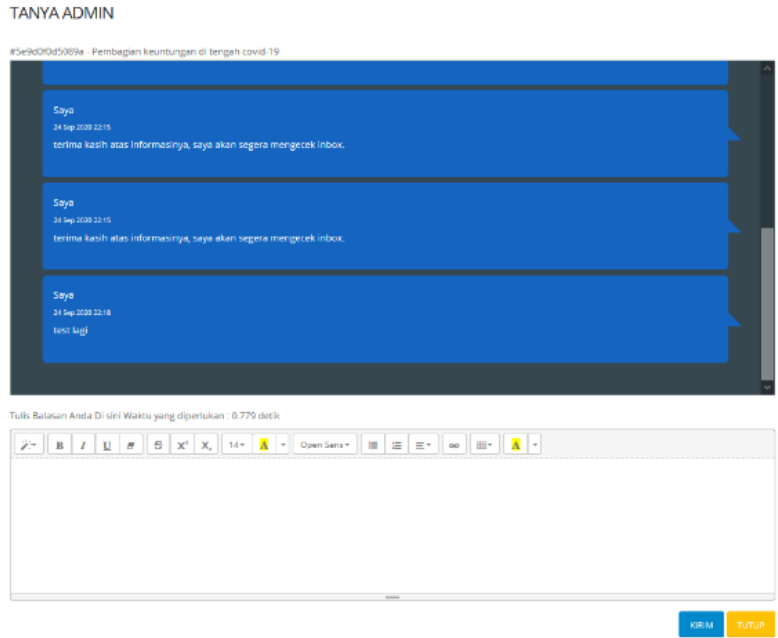

Fig.6. Ask Admin (Query Builder)

Code Eloquent :

```
$msgs = TanyaAdmin::with(['details' => function($query) {
        $query->orderBy('id', 'DESC');
            }, 'user'])->where('product_id', $product->id)->where('user_id',
                    $user->id)->get();
```
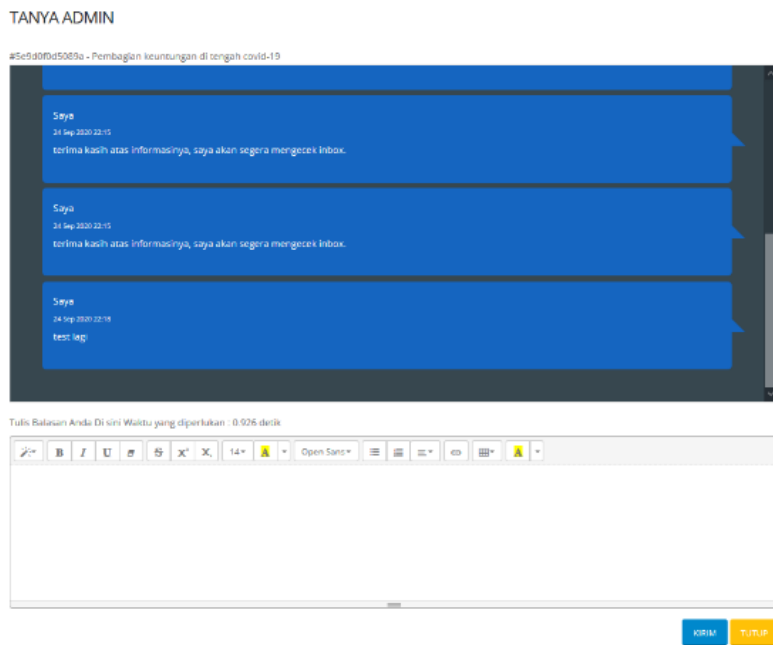

Fig. 7. Ask Admin (Eloquent)

In the above discussion, it can be seen that each page has a different table structure and relations so that pages that use quite a lot of table relations will look faster if you use the query builder method, but on the other hand, if you only use a single table or a minimum relationship, you can simply use the eloquent method.

6. Conclusion

After observing the results of research trials from this study, a module that uses table relations should use the query builder method because, in terms of faster access, Modules that do not have table relations or those that have fewer relations can use the fluent method. the method has its advantages - individual and invincible with implementation in each module that uses. Application developers in this case can already choose a method that can be used to improve performance.

The research that has been done is still not perfect, so some further research is needed to get maximum results. Some suggestions for development that can be done include the need for further analysis related to other modules that can support in terms of speed of access and the addition of several index tables on transactions that use multiple tables for maximum results.

Acknowledgment

References

[1] Chen, X. et al. (2017) 'Restful API Architecture Based on Laravel Framework', in Journal of Physics: Conference Series. DOI: 10.1088/1742-6596/910/1/012016.

[2] Davis, W. S., Yen, D. C. and Davis, W. S. (2020) 'Data flow diagrams', in The Information System Consultant's Handbook. DOI: 10.1201/9781420049107-24.

[3] Greenspan, G. (2017) 'Blockchains vs Centralized databases', Multichain.

[4] Halimi, H. and Jound, I. (2016) 'Comparison of performance between Raw SQL and Eloquent ORM in Laravel', Faculty of Computing Blekinge Institute of Technology, Sweden.

[5] Laravel (2019) Laravel - The PHP Framework For Web Artisans, Laravel.

[6] Madir, J. and Chapman, P. (2019) 'CROWDFUNDING', in FinTech. DOI: 10.4337/9781788979023.00015.

[7] Pamungkas, R. (2018) 'Optimalisasi Query Dalam Basis Data My Sql Menggunakan Index', RESEARCH : Computer, Information System & Technology Management. doi: 10.25273/research.v1i1.2453.

[8] Pendidikanku (2016) Pengertian Entity Relationship Diagram (ERD), www.pendidikanku.org.

[9] Short, J. C. et al. (2017) 'Research on Crowdfunding: Reviewing the (Very Recent) Past and Celebrating the Present', Entrepreneurship: Theory and Practice. DOI: 10.1111/tap.12270.

[10] Sinha, S. and Sinha, S. (2017) 'Eloquent', in Beginning Laravel. DOI: 10.1007/978-1-4842-2538-7_9.

[11] Sinha, S. and Sinha, S. (2019) 'Database Migration and Eloquent', in Beginning Laravel. DOI: 10.1007/978-1-4842-4991-8_5.

[12] Soto-Rey, I. et al. (2015) 'User Satisfaction Evaluation of the EHR4CR Query Builder: A Multisite Patient Count Cohort System', BioMed Research International. DOI: 10.1155/2015/801436.

[13] Umamageswari K, Kalpana R (2017) 'Web Data Extraction from Scientific Publishers Website Using Heuristic Algorithm', I.J.Intelligent Systems and Applications, 2017, 10, 31-39. DOI: 10.5815/ijisa.2017.10.04

[14] Emmanuel C. Paul (2016) 'Methodology of Compiling Web-Applications into Executables, Obtaining Seamless Server Installations and GUI Navigations through Qt and C++ Process Communications', I.J. Information Engineering and Electronic Business, 2016, 6, 28-36. DOI: 10.5815/ijieeb.2016.06.04

[15] Yuying Jiang, Zhan Huang , Zhanhong Huang (2012) 'Design and Implementation of a General Web-basedCourse Teaching ManagementSystem', I.J.Education and Management Engineering,2012, 11, 1-7. DOI: 10.5815/ijeme.2012.11.01

[16] Adiansah, W., Mulyana, N., & Fedryansyah, M. (2016). Potensi Crowdfunding di Indonesia dalam Praktik Pekerjaan Sosial. In Prosiding KS: Riset & PKM (hal. 155–291)

[17] Belleflamme, P., Lambert, T., & Schwienbacher, A. (2014). Crowdfunding: Tapping the right crowd. Journal of Business Venturing, 29, 585–609. https://doi.org/10.1016/j.jbusvent.2013.07.003

[18] Chang, S. E. (2018). Regulation of Crowdfunding in Indonesia. Law Review, XVIII(1), 41–71.

[19] Gleasure, R., & Feller, J. (2016). Does Heart or Head Rule Donor Behaviors in Charitable Crowdfunding Markets? International Journal of Electronic Commerce, 20(4), 499–524. https://doi.org/10.1080/10864415.2016.1171975

Authors' Profiles

Putu Adi Guna Permana is a master of Informatics Engineering at Amikom University Yogyakarta, He is currently a Lecturer at the Campus Institute of Technology and Business (ITB) STIKOM BALI. His interests are Information Systems, Systems Analysis and Design, Databases and Programming, Game Programming.

Evi Triandini is a Lecture at the Campus Institute of Technology and Business (ITB) STIKOM BALI. His interests are Information Systems, Systems Analysis and Design