

# IMPLEMENTATION OF KANBAN APP ON BORNEO CELL USING GO PROGRAMMING LANGUAGE AND TAILWINDCSS

<sup>1</sup>Ilham Ramadhan

<sup>2</sup>Yuhilza Hanum

<sup>1</sup>Universitas Gunadarma, ilham26ramadhan@gmail.com

<sup>2</sup>Universitas Gunadarma, yuhilza@staff.gunadarma.ac.id

## ABSTRACT

*The development of information technology has changed business dynamics with the use of efficient project management systems. Borneo Cell Shop, as a retailer that sells gadgets and accessories, needs an effective task management system to optimize employee workflow and task priorities. This research implements the kanban method at the Borneo Cell shop in the form of an application for its operations. This research includes the stages of planning, design, development and testing. The UML diagrams used for the design are use case diagrams and activity diagrams. The navigation structure used is the composite (mixed) structure. Database was created with PostgreSQL, and consists of users, categories and tasks. The interface design is made using wireframes. The programming language and tools used are Go language and Tailwind CSS. This application provides the form for new users to register, and then create the record of email address and password in the existing database. Functions that are related to the workflow are creating/adding, moving, and deleting categories and tasks. There are also informative messages for the cases of wrong input. This research is similar to previous ones, but differ in the field of implementation and design details. Test results show that this application can function correctly and is ready to be used in Borneo Cell Shop.*

**Keywords:** kanban, Borneo Cell Shop, workflow, application

## INTRODUCTION

Borneo Cell Shop is a retail shop that sells various gadgets and accessories in the Cikarang area. Like other retail businesses, Borneo Cell needs to continue to adapt to market changes and improve their operational efficiency to remain competitive. In an effort to increase operational efficiency, Borneo Cell needs an effective task management system to optimize work flow and prioritize tasks that must be completed by employees. One solution that can be implemented is the use of a kanban application.

This research implements the kanban method at Borneo Cell with the aim of increasing the efficiency and effectiveness of task management and improving workflow within the store. The kanban method was chosen because

of its flexibility, which can be adapted to changing business needs and changes, as well as its ability to provide clear visibility into workflow and task priorities. The aim of this research is not implementing kanban method to develop this application, but to develop an application that implement the kanban method for the workflow at Borneo Cell. The actor for this application, later referred simply as “user”, is the staff of this shop. Eventhough the terms used for this application are similar to peer applications, the tasks and categories are specific for the operation of the shop.

Central to the definition of kanban is the concept of flow. Flow is the movement of potential value through a system. The strategy of kanban is to optimize value by optimizing flow.

Optimization does not necessarily imply maximization (Vacanti, 2020). The visualization of the Definition of Workflow (DoW) is called a kanban board.

Figure 1 shows a typical kanban board for software development. It is divided into stages or phases, such as “Backlog,” “Analyze,” “Develop,” “Test,” and “Deploy.” Each stage occupies one column on the board, further divided into two subcolumns: 1) “Doing” and 2) “Done” (Damij & Damij, 2021).

Some researches implemented the kanban method either for the development of systems or applications, or as the object of the software development, that is, an application is created for the kanban management. Suparman and Huda (2021) developed under graduate thesis management system by using kanban. The system catered to three actors, namely, study program as administrator, thesis supervisors, and final-year students. The system was divided for those three actors, and provided the relevant tasks. Kanban was implemented as the flow that made the procedure of doing thesis efficient.

Ganindra and Findawati (2021) developed a similar system to those of Suparman and Huda’s. They created a web-based programmer performance recording and measurement information system for a company. They use a kanban board similar to that of Figure 1 that consisted of “to do,” “doing,” and “done” columns for tasks. There are also “Task Evaluation” and “Programmer Task Report” forms provided. Black-box testing was conducted on the functions of Login, Create Task, Move Task, Create Programmer, Task Evaluation, and Programmer Task Report.

This research used Go programming language to develop the application. Go bears a surface

similarity to C and, like C, is a tool for professional programmers, achieving maximum effect with minimum means. Go is especially well suited for building infrastructure like networked servers, and tools and systems for programmers, but it is truly a general-purpose language and finds use in domains as diverse as graphics, mobile applications, and machine learning (Donovan and Kernighan, 2016).

Tailwind CSS, which is also used in this research, works by scanning all HTML files, JavaScript components, and any other templates for class names, generating the corresponding styles and then writing them to a static CSS file. Azhariyah and Mukhlis (2024) developed a frontend for a website store by using Tailwind CSS as a CSS framework. They utilize utility classes to create user interface elements in Tailwind CSS during the front-end design phase.

PostgreSQL that was used in this research to create database uses a client/server model of communication. A PostgreSQL server is continually running, waiting for client requests. The server processes the request and returns the result to the client. Choosing an interface because the PostgreSQL server runs as an independent process on the computer, a user cannot interact with it directly. Instead, client applications are designed specifically for user interaction (Momjian, 2001).

Wireframes, that are created as the initial design of user interfaces, are the basic blueprints that illustrate the core form and function found on a single screen of a web page or application. The fidelity of these wireframes will increase in detail as they are refined. The first version is likely to just utilize basic black and white outlines and shapes to hint at where navigational elements, text, and graphics will be placed on the screen. The collection of these wireframes should give a

comprehensive skeletal view of the entire product (Hamm, 2014). Figure 2 shows an example of a first draft wireframe of a website home page.

## **RESEARCH METHODS**

The first step of this research is analyzing the hardware and software requirements to develop the application. The hardware required are processor Intel Core i5-8365U @1.60GHz, NVME SSD 256GB of storage, and 24GB Dual Channel RAM. This application is developed on Linux Ubuntu 22.04 operating system. Software requirements are Visual Studio Code, and PostgreSQL. Google Chrome was used as the browser.

The second step is to draw diagrams, which are navigation structure, use diagram, activity diagrams, and class diagram. These steps are explained further.

### **Designing System Navigation**

The navigation structure used in this application is a composite (mixed) navigation structure, which is a combination of hierarchical and linear structures. Figure 3 shows that when the website is first opened a landing page will appear containing a login and registration menu. After logging in, a page is displayed containing todo, in progress, done, back log and log out. Users can also add new categories according to their needs.

### **Designing User Use Case Diagram**

A use case diagram is a diagram that explains the user access flow in using this system and illustrates what can be done. This diagram in Figure 4 shows that the user can access the system in the box and browse the todo, done, in progress, backlog, and add category menus.

### **Designing User Registration Activity Diagram**

In the activity diagram in Figure 5, it is illustrated how to register for the kanban application. To start the process, the user types the website address in a web browser, then the system will open the landing page of the kanban website. To register an account, the user clicks the sign up button in the middle of the page. On this page the user enters registration data, namely full name, email and password. After that, the API system will check the format of the email entered to see whether it complies with the requirements provided. If the data is valid then the registration process has been successful and the system will save the data into the database.

### **Designing Activity Diagram Design for Creating New Tasks**

Figure 6 is an activity diagram for creating a new task. To start the process, users are required to log in first by typing the website address in a web browser. The system will display a login page, then check whether the email address and password are in the database. If there is, a dashboard page will be displayed. On the dashboard page, several categories are displayed that are available to choose from. The system displays the add new task page for the user to write the new task they want to create. Data is saved into the database and displayed on the dashboard. The steps for creating a new category are the same as a new task, but have a different menu.

### **Designing Class Diagram**

Class diagrams are used to design a database where each class has a relationship to collaborate with each other. This web-based Kanban App application has 3 classes, namely users, categories and tasks. Each class represents data and is interconnected

between one entity to another. This class diagram is shown in Figure 7.

### **Designing Database**

At this stage, the database structure that will be used in creating the API is designed. In this application the API will be used as a link between the database and the application. This web-based Kanban application has a database with the name `kanban_app.db` and has 3 tables, namely users, categories and tasks.

Table 1 shows the Users table which is used to store user data. This table has an ID field which will be the primary key and auto-incremented because each user has a different ID. Then there is the email address and password used by the user to log in. Next there is the fullname field which is used to accommodate data entered by the user when creating an account. The `created_at` and `updated_at` fields have a timestamp data type to record the time a user created or changed an account.

Table 2 shows the Categories table which is used to store the category data created. This table has an ID field which will be the primary key and auto-incremented, because each category has a different ID. The `user_id` field is a foreign key that refers to the ID in the Users table. The `type` field is used to store the latest category data created by the user. The `created_at` and `updated_at` fields use timestamp data type to see when a category was created or changed.

Table 3 shows the Tasks table which is used to store task data. This table has an ID field which is the primary key and auto-incremented, because each task has a different ID. The `category_id` and `user_id` fields are foreign keys that refer to the ID in the Categories and Users tables. The `title` and `description` fields are used to store the latest task data created by the user. The `created_at`, `updated_at` and `deleted_at` fields use

timestamp data type to see when a task was created or changed.

### **Designing Wireframe**

At this stage, a wireframe design or application display is created that suits the needs of the previous design stages. The wireframe design for the web-based Kanban application that was built includes the main page, login page, register page, dashboard page, add task page, and add new category page wireframes.

### **Testing**

This stage aims to find out whether all the functions in the application can work well, and to find out whether the application can be used according to the needs and expected goals. On this kanban website, testing is carried out using the black box method.

## **RESULTS AND DISCUSSION**

### **Backend Implementation**

Backend, which is related to the server or web service and database is implemented using the Go language. What was done in this research was to install the Go Extension in the Visual Studio Code text editor. Figure 8 shows the Go installation view in Visual Studio Code.

### **Database Creation**

In this application the database used is PostgreSQL using the GORM library. Figure 9 shows the PostgreSQL database view.

### **Middleware Creation**

The Borneo Cell kanban application uses an intermediary in handling requests or middleware because of the large amount of routing used. A routing handler receives various requests, such as Post, Get, Put, and Delete.

### API Creation

The process of using the Borneo Cell kanban application begins with logging in or authenticating the user. When creating a login API (Application Programming Interface), the user logs in or authenticates using email and password to gain access to the database. Apart from the login API, according to what has been described in the research method, an API for register, log out, create new category, and add new task was also created. The next stage is creating an API route whose role is to manage the entrance in the form of requests in the application.

### Frontend Implementation

The frontend display was created by using a package from the Go language, namely the Golang template, and using a CSS framework, namely TailwindCSS. The wireframe design that was created previously is implemented at this stage.

### Home Page Creation

The home page is the first page that appears when a user accesses this website address. On the home page, the Sign Up menu for registering and Sign In for logging in is displayed as shown in Figure 10.

### Creation of Login Page

The login page is the page that appears when the user clicks the Sign In

button. The login page display can be seen in Figure 11.

### Creation of Register Page

The register page is the page that appears when the user clicks the Sign Up button. The display of the register page can be seen in Figure 12.

### Creating Dashboard Page

The dashboard page is the page that appears when the user successfully logs in. The dashboard page display can be seen in Figure 13.

### Add Task Page Creation

The add task page is a page that appears when the user clicks the plus (+) icon. The Add Task page display can be seen in Figure 14.

### Creation of Add New Category Page

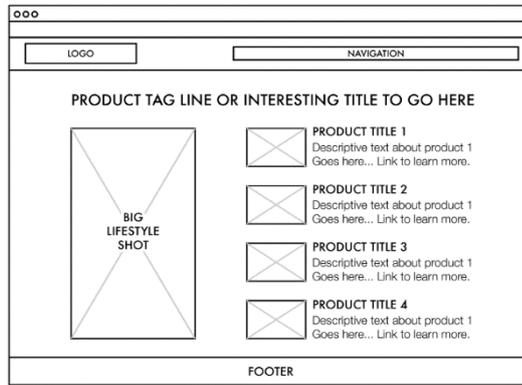
The add new category page is the page that appears when the user clicks on the Add Category menu. This option works if the user wants to add a new category. The appearance of the add new category page can be seen in Figure 15.

### Black-Box Testing

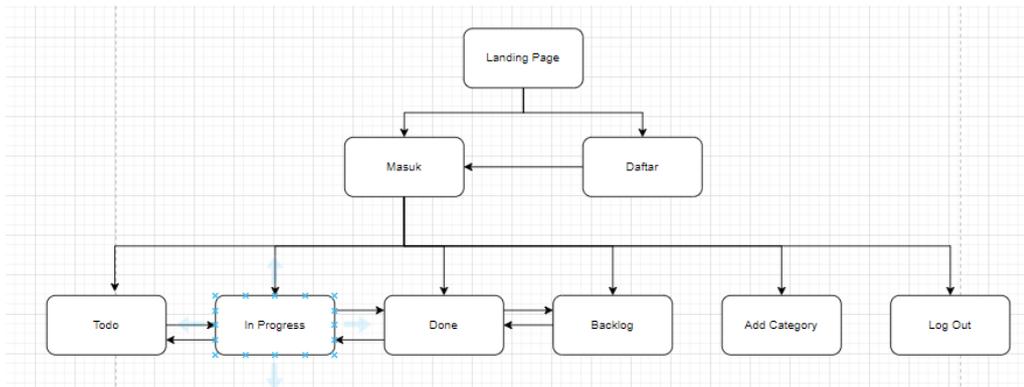
Black-Box testing is a software testing method that focuses on the functionality side, especially on whether the application input is as expected or not. Results of black-box testing on this application is as shown in Table 4.

Backlog	Ready (4)	Analyse (2)		Develop (3)		Test (2)	
		Doing	Done	Doing	Done	Doing	Done

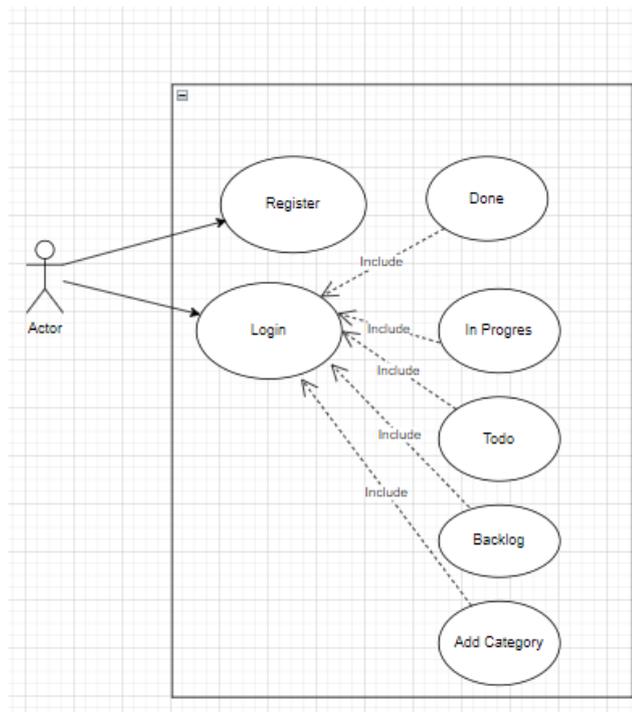
Figure 1. A typical kanban board for software development (Damij & Damij, 2021)



**Figure 2. A First Draft Wireframe (Hamm, 2014)**



**Figure 3. Navigation Structure**



**Figure 4. Use Case Diagram**

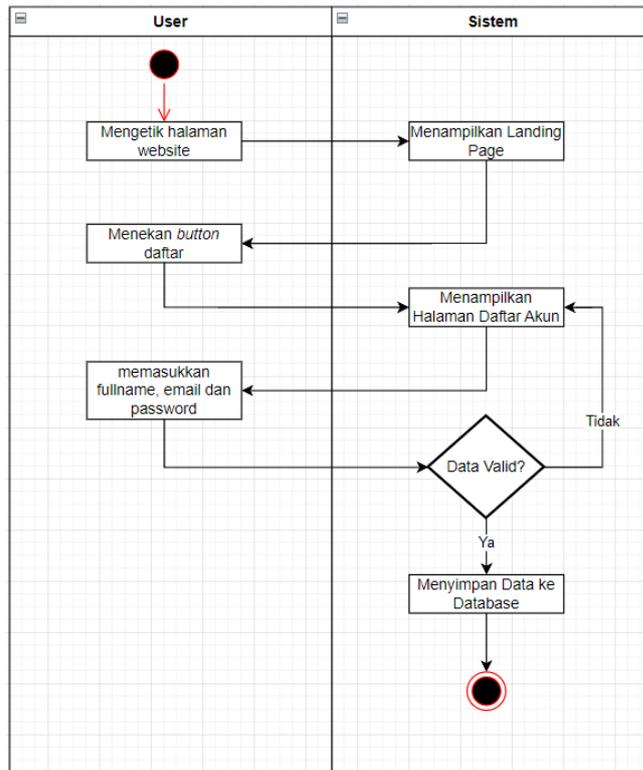


Figure 5. Activity Diagram for User Registration

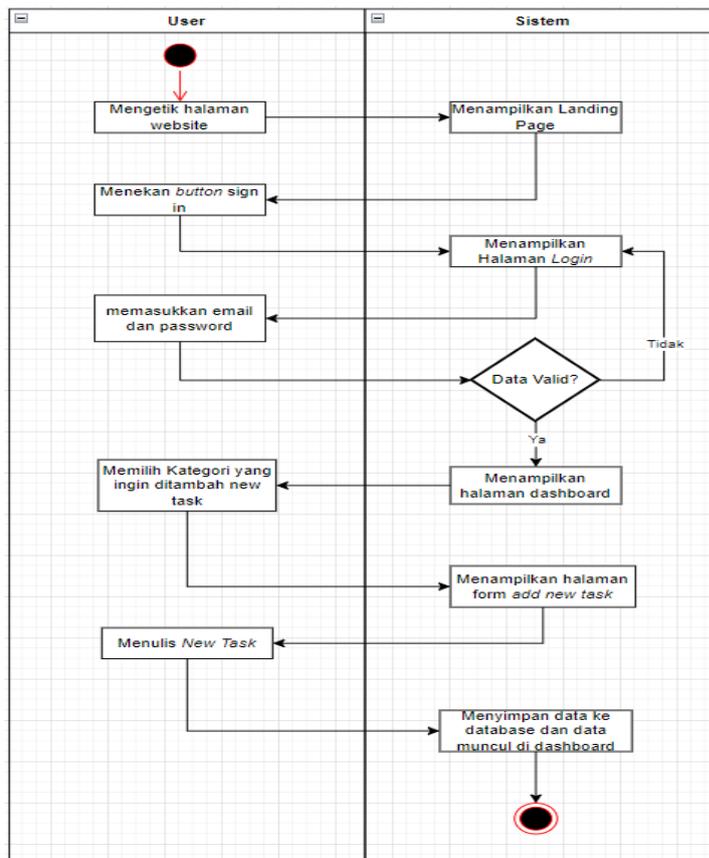
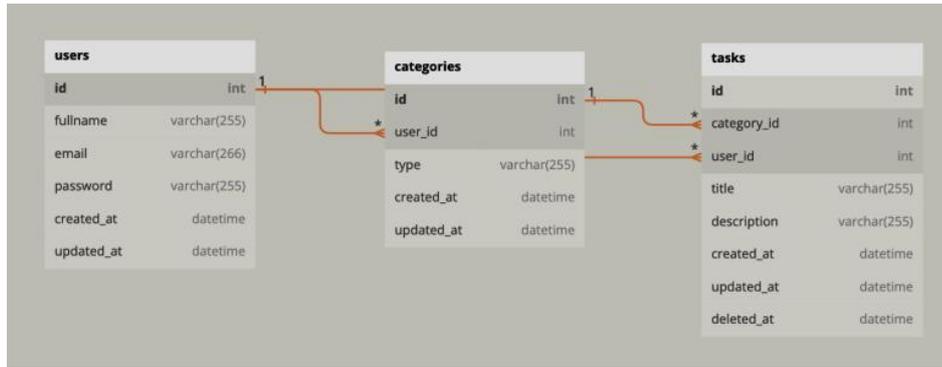


Figure 6. New Task Activity Diagram



**Figure 7. Class Diagram of the Application**

**Table 1.  
Users Table Database Structure**

Field name	Data type	Character length	Constraint
id	Integer	Default	Primary Key
fullname	Varchar	255	
email	Varchar	266	

**Table 2.  
Categories Table Database Structure**

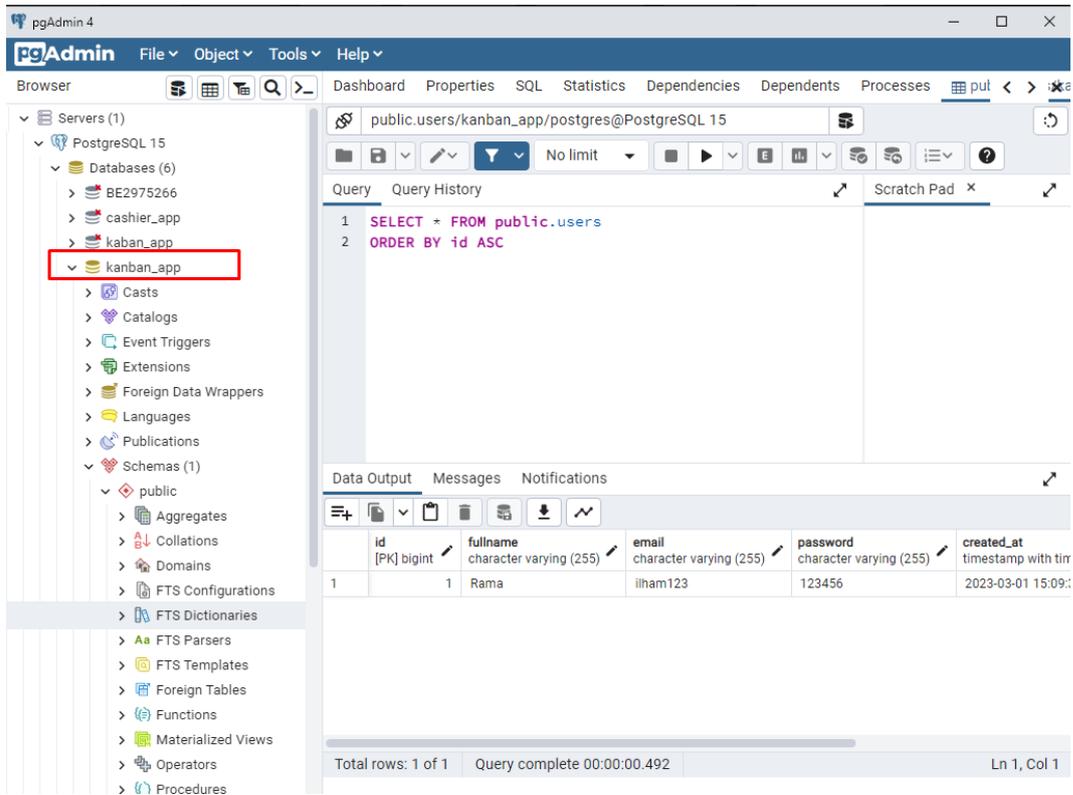
Field name	Data type	Character length	Constraint
id	Integer	Default	Primary Key
user_id	Integer	Default	Foreign Key Reference from <i>users table</i>
type	Varchar	255	
created_at	Timestamp	4 bytes	
updated_at	Timestamp	4 bytes	

**Table 3.  
Tasks Table Database Structure**

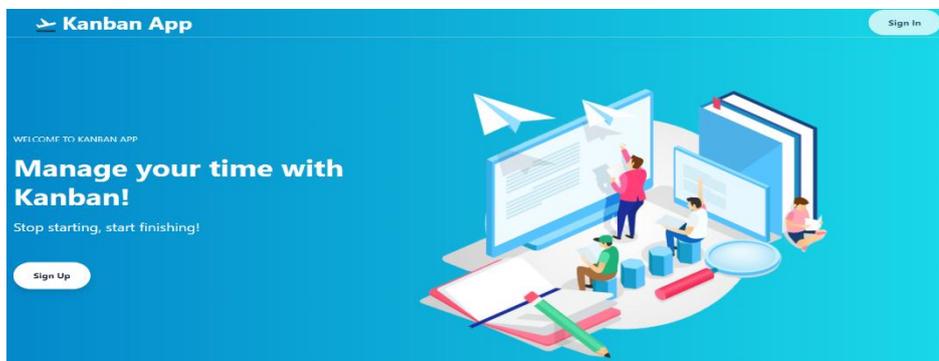
Field name	Data type	Character length	Constraint
id	Integer	Default	Primary Key
category_id	Integer	Default	Foreign Key Reference dari tabel <i>categories</i>
user_id	Integer	Default	Foreign Key Reference dari tabel <i>users</i>
title	Varchar	255	
description	Varchar	255	
created_at	Timestamp	4 bytes	
updated_at	Timestamp	4 bytes	
deleted_at	Timestamp	4 bytes	



**Figure 8. Installing Go Extensions on Visual Studio Code**



**Figure 9. PostgreSQL Database**



**Figure 10. Home Page Display**

**Login**

**Email**

**Password**

**Login** [Create account](#)

**Figure 11. Login Page Display**

**Register**

**Fullname**

**Email**

**Password**

**Register** [Login](#)

**Figure 12. Register Page Display**

Logout

**Kanban App**

Todo + X In Progress + X Done + X Backlog + X [add Category](#)

7 Agustus Sudah Dapat Tanda Tangan Dosen Pembimbing	Juli Bab 3 - 4 Penulisan	Juli Bab 1 - 2
--	-----------------------------	-------------------

**Figure 13. Dashboard Page Display**

←

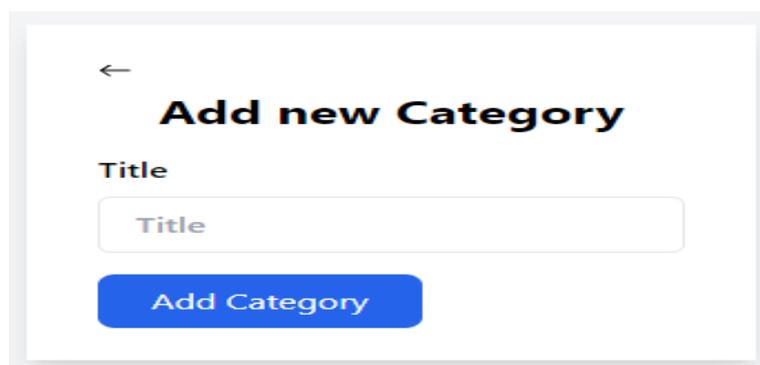
**Add new task**

**Title**

**Description**

**Add Task**

**Figure 14. Add New Task Page Display**



**Figure 15. Add New Category Page Display**

**Table 4.  
Black-Box Testing Results**

No	Input tested	Expected result	Test result	Conclusion
1.	Correct username and password	User successfully logs in and is then directed to the dashboard page	Success	Valid
2.	Wrong username and password	Failed to log in and the form will be empty again	Success	Valid
3.	Empty Email, fullname, and password	Displays message code 400 and message "register data is empty"	Success	Valid
4.	Logout	Displays message code 200 and message "logout success"	Success	Valid
5.	Create New Category	Displays message code 201 and message "success create new category"	Success	Valid
6.	Todo, In Progress, Done, and Backlog categories	Can display categories that have been created based on default	Success	Valid
7.	Deleting category	Category deleted	Success	Valid
8.	Adding category	Category added	Success	Valid
9.	Adding Task	Task added	Success	Valid
10	Deleting Task	Task deleted	Success	Valid
11	Moving Task	Task moved	Success	Valid

The results of the black box testing shows that the various inputs produced the correct results. The intentional mistakes also produced the correct message. This kanban application for Borneo Cell shop realized the concept explained by Vacanti (2020). To some extent, it also implemented the kanban board described by Damij and Damij (2021). Tools and design explained by Donovan and Kernighan (2016), Azhariyah and Mukhlis (2024), Momjian (2001), and Hamm (2014) provided the background for tools and design of this application, and were used to the fullest. This research is in line with the work of Suparman and Huda (2021) and Ganindra and Findawati (2021) who developed similar systems/applications.

## CONCLUSION AND SUGGESTION

Based on the results of the development of this kanban application, it can be concluded that a dynamic application that caters to the workflow of an institution, in this case the Borneo Cell Shop, succeeded in implementing this method. The dynamic feature is provided in the add, move, and delete categories and tasks as they are finished or not relevant anymore. The application showed good performance in black box testing.

The design and development of this application can still be completed or perfected. Some suggestions that can be given are adding actors, such as the higher management of the shop. The existing staff can also be classified according to their specific jobs and tasks. This research can be further extended by analyzing the performance of the staff or the operation of the shop as a whole after using the application. This can be preceded by creating business process diagram for the shop, that might be realized as another research.

## REFERENCES

- Akbar, G.G.A., & Findawati Y. (2021). Sistem Informasi Pencatatan dan Pengukuran Kinerja Programmer Berbasis Web (Studi Kasus: PT. Bits Miliartha). *Procedia of Engineering and Life Science Vol. 2. No. 1 October 2021, Seminar Nasional Fakultas Sains dan Teknologi (SENASAINS 3 rd )*, Universitas Muhammadiyah Sidoarjo.
- Azhariyah, S., & Mukhlis, M. (2024). Framework CSS: Tailwind CSS Untuk Front-End Website Store PT. XYZ. *Jurnal Informatika*, 3(1), 30-36.
- Damij, N., & Damij, T. (2021) . An Approach to Optimizing Kanban Board Workflow and Shortening the Project Management Plan. *IEEE Transactions on Engineering Management*, 1-8.
- Donovan, A.A.A., & Kernighan, B.W. (2016). *The Go Programming Language*. Indiana: Addison-Wesley.
- Hamm, M.J. (2014). *Wireframing Essentials: An introduction to user experience design*. Birmingham: Packt Publishing.
- Momjian, B. (2001). *PostgreSQL: Introduction and Concepts*. New Jersey: Addison-Wesley.
- Suparman, P., & Huda, M. (2021). Penerapan Kanban Agile Development dalam Pengembangan Sistem Manajemen Skripsi dan Tugas Akhir STMIK Cikarang Menggunakan Framework CodeIgniter. *Jurnal Informatika SIMANTIK*, 6(1), 7-11.
- Vacanti, D.S. (2020). *The Kanban Guide*. Orderly Disruption Limited.