Applied Information Technology And Computer Science Vol. 4 No. 2 (2023) 1282-1299

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

GunungHub – Guide and Reservation Online Application

Amir Asyraaf Nor Azman[1], **Suhaila Mohd. Yasin**[1]

[1]Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2023.04.02.073
Received 23 June 2023; Accepted 09 November 2023; Available online 30 November 2023

Abstract: Most hikers in Malaysia find it challenging to join hiking trips and identify climbable mountains. The current method for joining a hiking expedition is complicated. At present, advertising trips and making reservations are not systematic and time-consuming. Further, it is difficult to handle trip booking information by numerous participants. Thus, a system is developed that incorporates all of the essential operations and information into a single web application. The database used makes booking management and mountain information more efficient. Customers may learn about the trip and make bookings in a structured manner. Errors between administration and customers are minimal. The system was developed using an object-oriented approach, with the prototype model as the development model. The system was developed using the Laravel framework. This system improves the efficiency of current procedures in booking management. This methodology will assist both the climbing community and the organization by providing a systematic way.

Keywords: Hiking expedition reservation, Web application, Database

1. Introduction

GunungHub is a Malaysian company founded by Muhammad Amir Mu'im that is located at De Centrum Tower, Kajang [1]. The organization is a centralized hub for Malaysia outdoor lovers that specializes in providing adventure trips, especially hiking to its almost 10,000 customers. As for now, the reservation and subsequent payment are managed manually through WhatsApp instant messaging app while announcements are blasted on its social media platform. The customer fills in the reservation form manually through WhatsApp, and the information received will be transferred to an Excel file by the administrator. Besides that, the information about the trails for hikers is limited and not readily available. The hikers need to do their research about the mountains and trails. The current method requires a lot of human effort and time to handle almost thousands of customer conversations at a certain time. The limited information about the mountains or trails makes it less attractive to potential customers. The hikers will just ask the admin about the mountain and its trails through WhatsApp, leading to additional effort to explain to the customers.

**Corresponding author: ysuhaila@uthm.edu.my*

Therefore, this system is proposed to make the slot reservation more efficient. The GunungHub Guide and Reservation Online Application will guide the customers to complete the reservation and subsequent payment by themselves. The customer information will be automatically updated in the system after completed the payment. The system will also provide information about the mountains such as the level of difficulties, the location, and the attractiveness of the mountain. The information about the mountain will be provided in the system that will be manually updated by the administrator. As a result, the administrator has the authority to the effective method for managing the reservations. As for customers, they can systematically access the trip reservation. The proposed system is developed to manage the reservation for the hiking trip effectively and minimize the mistakes occurring between customers and administrators. The mountain information acts as guidance to attract more people who love hiking including the hiking community to grow bigger.

The main objective of the GunungHub Guide and Reservation Online Application is to design using an object-oriented approach, to develop using a mobile web-based application approach, and to test the developed system using beta testing. The scope of this project focuses on the administrator of GunungHub and the customer as a general user. There are several modules contained in the system which are the register module, login module, manage mountain information module, manage reservation module, manage reservation history module, make payment module, and generate report module. This paper is divided into five chapters. The first chapter provides context for the project. The second chapter discusses the linked works. The approach is elaborated on in Chapter 3, and the results and discussion are summarized in Chapter 4. Finally, chapter 5 provides the project's conclusion.

2. Related Work

2.1 Reservation Online Application

An online reservation application is a convenient and efficient way to book services or resources over the Internet. It allows users to browse and compare available options, select, and customize their reservations, and complete the booking process all in one place. With an online reservation application, users can easily make reservations from any device with an internet connection, at any time of day. This means that users don’t have to physically visit a business or call during business hours to make a reservation. Additionally, an online reservation application can help streamline the reservation process for both customers and businesses, by automating many of the tasks involved and providing a clear and organized overview of reservations. As a result, an online reservation system, in essence, enables a potential customer to book and pay for a service directly through a website. A customer needs to choose which adventure trip package is based on the destination, price, date, and time. That is, from the moment a customer decides they want to book a slot for your service to selecting a date, selecting a time, and paying for the booking, everything is handled online.

2.2 Mobile-Based Web Application

A web application is accessed through the Internet using a web browser [2]. It enables them to exchange information with their target market and conduct quick, secure transactions. However, a mobile-based web application can be thought of as a different version of a website. It is designed to be used on smartphones and other mobile devices such as tablets. A generic mobile web application is another title for website mobile versions [3]. Technically, there are several approaches to constructing and developing mobile versions; nevertheless, the common concept is that the desktop version of a website checks for mobile devices using the user-agent identity from the web browser. the company provides a service to its customers by offering an adventure trips. A mobile-based web application is more dynamic in terms of flexibility than a mobile application for the customer to do the reservation slot in joining the adventure trip. Administrators can check and manage customer reservations anywhere and at any time, even if they do not have a laptop with them.

2.3 Study of existing related system

Sabahparks [4] performs as a system for making reservations online for Sabah Parks, a company in charge of maintaining and managing the national parks in Sabah, Malaysia. The system's user-friendly interface, which makes it simple for users to check availability, make bookings, and obtain details on park policies and conservation initiatives, is one of its standout features. Mountain Project [5] is a comprehensive web reference for mountaineering and rock climbing. The website's vast database of climbing routes, which includes thorough descriptions, evaluations of the routes' difficulty, and user reviews, is a unique feature. Climbers may use this tool to find new routes, gauge their difficulty, and learn from the climbing community. GAdventure [6] is an online platform that provides a broad choice of escorted adventure travel experiences worldwide. The organization behind the website, G Adventures, specializes in offering immersive and sustainable travel experiences to locations all over the world. The website stands out for its wide range of adventure travel itineraries, which appeal to different interests, tastes, and travel types. The website provides a wide variety of experiences, from animal safaris and trekking trips to cultural immersion and community-based tourism. Table 1 shows the existing system and proposed system comparison.

Table 1: Existing system and proposed system comparison

Module/System	Sabahparks	Mountain Project	GAdventure	GunungHub Guide and Reservation Online Application
Login	√	√	√	√
Register	√	√	√	√
Manage Mountain Information	√	√	X	√
Manage Reservation	√	X	√	√
Reservation History	√	X	√	√
Payment	√	X	√	√
Generate Report	X	X	X	√

3. Methodology

The prototyping model methodology is adopted as the project methodology in the development of GunungHub – Guide and Reservation Online Application because developers can have a better understanding of the product requirements. This model was broken down into five phases: planning, analysis, design, prototype, and implementation phases. The prototype model is shown in **Figure 1** [7].

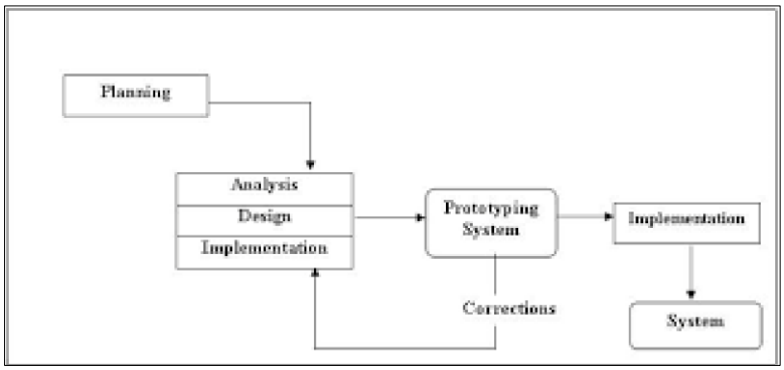

Figure 1: Prototyping model [4]

3.1 System requirement analysis

A functional requirement defined what was done by identifying the required activity, task, or action that had to be completed [8]. **Table 2** shows the modules and the functionalities of the proposed system. Non-functional requirements, rather than specific behaviors, specify the criteria that can be used to judge the operation of a system [8]. The non-requirement and the proposed system are described in **Table 3**.

Table 2: Functional Requirements

Modules	Functionalities
Register	• The customer shall register an account by full name, email, and password.
Login	• The customer and administrator shall log in and log out of their account using email and password.
Manage mountain information	• The customer shall view and add their comment to the mountain information. • The customers are allowed to give a review of the mountains. • The administrator shall manage the mountain information such as create, update, and delete.
Manage reservation	• The customer shall view and reserve a trip offered. • The customer shall fill out the reservation form and makes payments for the reservation. • The administrator shall manage the trip information such as creating, updating, and deleting.
Manage reservation history	• The customers are allowed to cancel the reservation within the amount of time. • The administrator is allowed to cancel the reservation if needed.
Payment	• The customer shall able to do the payment for reservation purposes using the online banking payment method. • The system will decrement the slot setup by the administrator after the customer completes the payment.
Generate report	• The administrator can generate a report for the total bookings and sales for the trip.

Table 2: Non-Functional Requirements

Requirements	Descriptions
Operational	• The system can be used in any web browser such as Google Chrome or Internet Explorer.
Security	• The system can only be accessed using email and a password.
Usability	• All types of users can easily understand the visual design and flow of the system.
Integrity	• The system's database will be properly maintained and protected from corruption.
Availability	• The system is designed to be simple to use and accessible at all times.
Maintainability	• Maintenance will be performed at regular intervals to minimize failure and maximize the system's consistent quality.

User requirements detail what users expect from the system. In other words, user requirement describes what the user expects the application to be capable of. **Table 3** summarizes the user requirements for the proposed system.

Table 3: User Requirements

No	User Requirements
1.	All users should be able to input their full name, email, and password for registration and login purposes.
2.	All users should be able to log in and log out of the application.
3.	Customers should be able to view the mountain information section.
4.	Customers should be able to view the specific mountain information.
5.	Customers should be able to navigate to the mountain within the system.
6.	The customer should be able to add their comment to the mountain information.
7.	Customers should be able to view the trip catalog.
8.	Customers should be able to view specific trip information.
9.	The customer should be able to fill out the reservation form and makes a reservation.
10.	The customer should be able to make a payment to complete the reservation process.
11.	The customer should be able to receive a receipt as proof of complete payment via email or SMS.
12.	Customers should be able to view the reservation history either upcoming, active, or completed.
13.	The customer should be able to cancel the reservation within a period amount of time.
14.	Administrators should be able to manage mountain information such as create, update, and delete.
15.	Administrators should be able to manage trip information such as create, update, and delete.
16.	The administrator should be able to view the reservation history made by the customer.
17.	The administrator should be able to cancel the reservation if needed.

No	User Requirements
18.	The administrator should be able to generate a report for every trip that shows the total bookings and total sales for each trip

3.2 System analysis

The object-oriented use case diagram is used to illustrate the system requirements with use cases, identify the actors, and explain how the actors interact with the use cases. **Figure 2** illustrates the proposed application's use case diagram with two actors: the customer and Administrator. Login, register, manage mountain information, manage reservations, manage reservation history, make payments, and generate reports are the seven use cases.

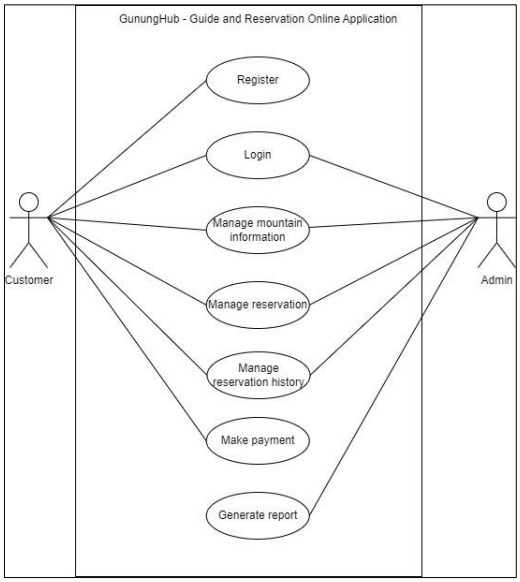

Figure 2: Use case diagram

A class diagram describes the attributes and operations of a class as well as the system constraints. Because they are the only UML diagrams that can be mapped directly to object-oriented languages, class diagrams are widely used in the modeling of object-oriented systems. Use cases are an important aspect of UML in telling a cohesive story about the behavior of a system [9]. **Figure 3** shows that six classes are identified in the proposed system which are Customer, Administrator, Reservation Form, Payment, and Mountain. Each class has its operations and features for this proposed system.

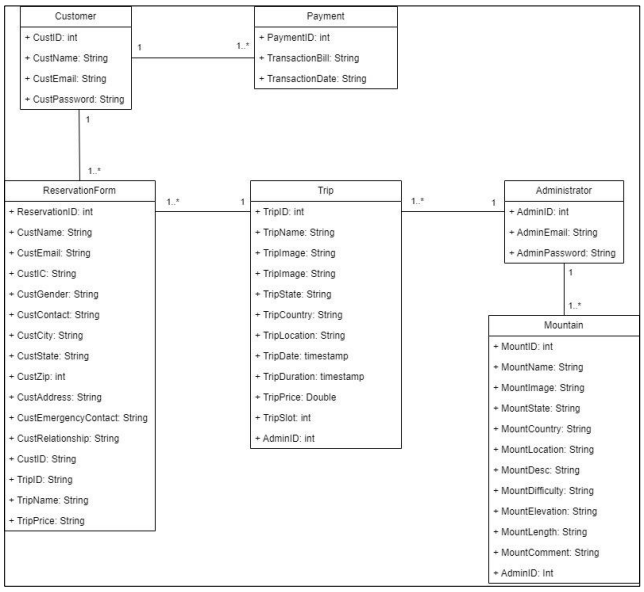

Figure 3: Class Diagram

A multi-layered architecture is a client-server architecture in which the front-end and back-end display, business rule processing, and data management operations are physically separated [10]. To decrease complexity and promote modularity, each layer is designed to execute a specific set of duties and is segregated from the other levels. Developers may better control the system's complexity and make it easier to maintain and alter by splitting it into layers. **Figure 4** shows each layer for multi-layer architecture that connects to each layer to execute better outcomes.

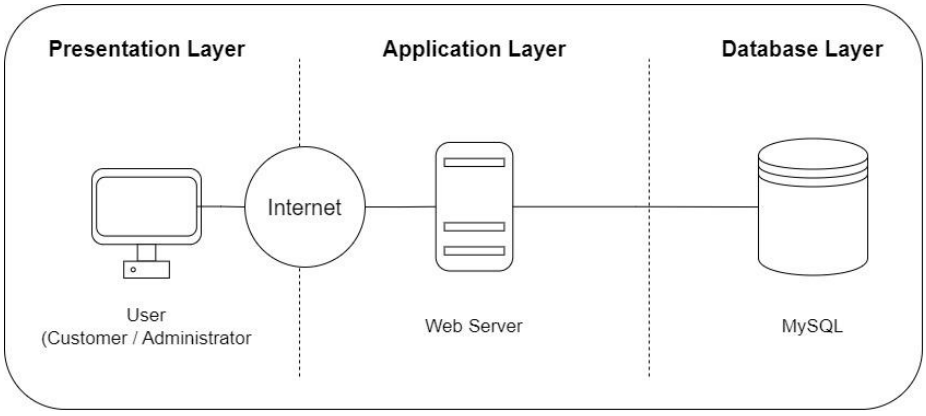

Figure 4: Multi-layered architecture

A database schema is an abstract idea of how data is stored in a database. It is utilized to define the data organization of a database as well as the relationships between tables. Developers create a database schema in advance to ascertain which elements are required and how they will connect. The Class Diagram-extracted database schema is listed as follows:

i. Customer (CustID, CustName, CustEmail, CustPassword)
ii. Administrator (AdminID, AdminEmail, AdminPassword)
iii. Reservation Form (ReservationID, CustName, CustEmail, CustIC, CustGender, CustContact, CustCity, CustState, CustZip, CustAddress,

CustEmergencyContact, CustRelationship, CustID, TripID, TripName, TripPrice, TripDate)

iv. Trip (TripID, TripImage, TripName, TripState, TripCountry, TripLocation, TripDate, TripDuration, TripPrice, TripSlot, AdminID)

v. Mountain (MountID, MountImage, MountName, MountState, MountCountry, MountLocation, MountDifficulty, MountDesc, MountElevation, MountLength, MountComment, AdminID)

vi. Payment (PaymentID, TransactionBill, TransactionDate)

3.3 Interface Design

The process of developing the user interface of a software program to make it easy and intuitive to use is known as interface design. This covers the layout and aesthetic design of the user interface, as well as information organization and flow. **Figures 5** to **9** show how the interface works, from the login screen to the checkout interface, as a preview of the system. Both the customer and the administrator will utilize the Guide and Reservation Online Application.

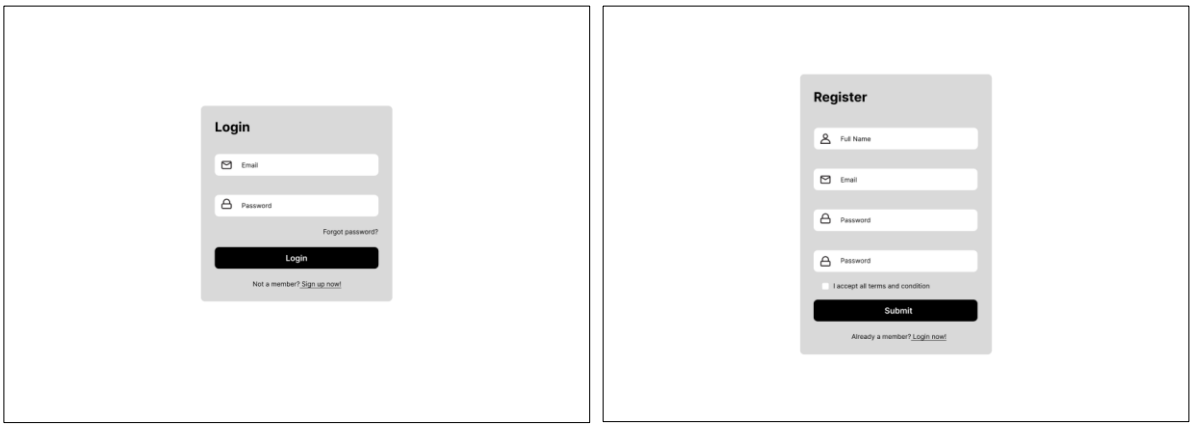

Figure 5: Login and register interface for user

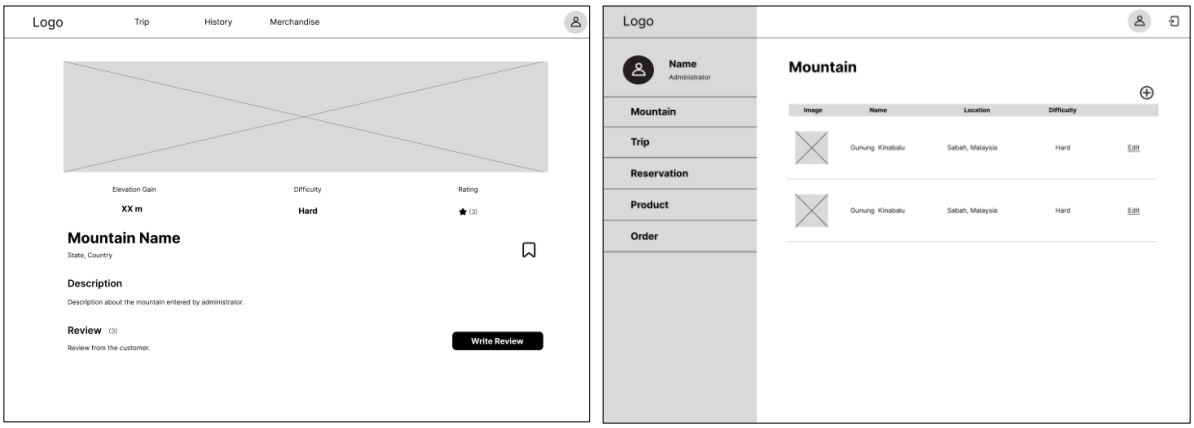

Figure 6: Manage Mountain information interface for both users

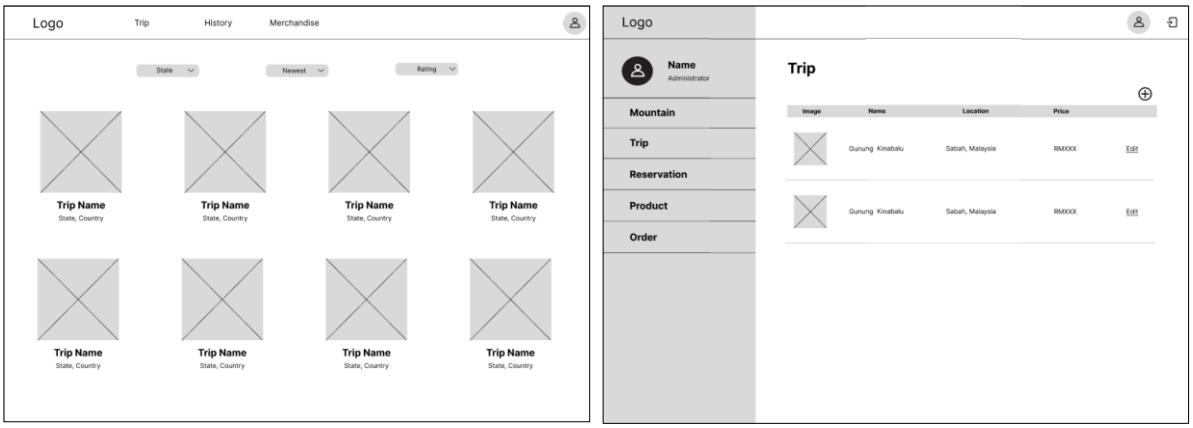

Figure 7: Manage trip for both user

Figure 8: Manage reservation history interface for both users

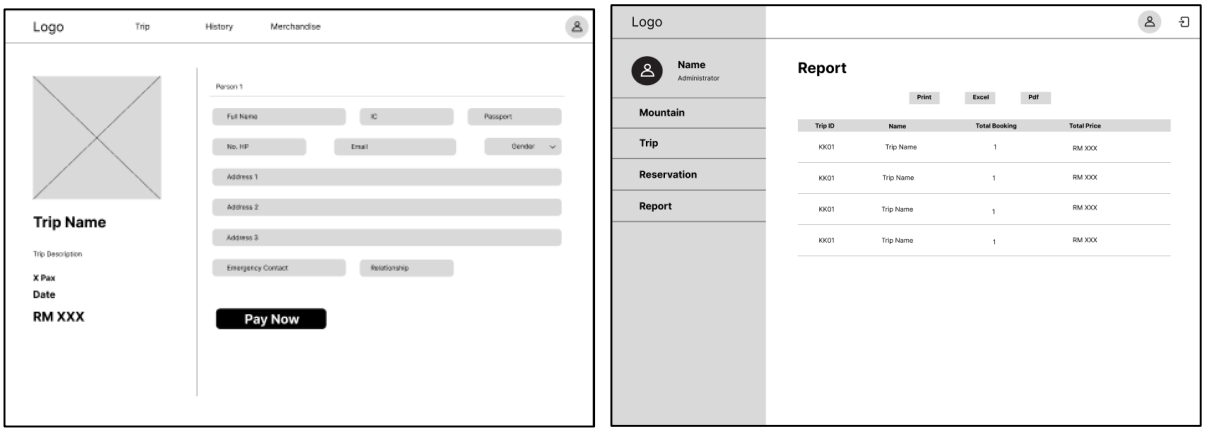

Figure 9: Booking form interface for customer and generate report interface for administrator

4. Result and Discussion

This section discusses the programming process that utilizes frameworks, tools, and programming languages to develop software modules, classes, and functions. The best practices for coding, such as code organization, documentation, and adherence to coding standards, are taught to students. The software used in system development is Visual Studio Code for programming, while the data will be managed in phpMyAdmin.

4.1 Module Implementation

(i) Registration interfaces

Only the customer needs to register an account besides the administrator due to early registration for an admin account for stakeholders. The registration form required the customer to enter a valid email, name, and password. The system will prompt an error for the existing email that has been registered to the database. **Figure 10** shows the registration interface with a code segment.

Figure 10: Registration interface with a code segment

(ii) Login interface

The login module is used by both users, administrators, and customers. The login form requires users to enter the email and password that has been registered to the system. The application will validate the email and password and display an error message to the user. After successful login, the user will be directed based on user type; either admin or customer. The administrator will be redirected to the dashboard interface, and the customer will be redirected to the homepage interface. **Figure 11** shows the login interface with a code segment.

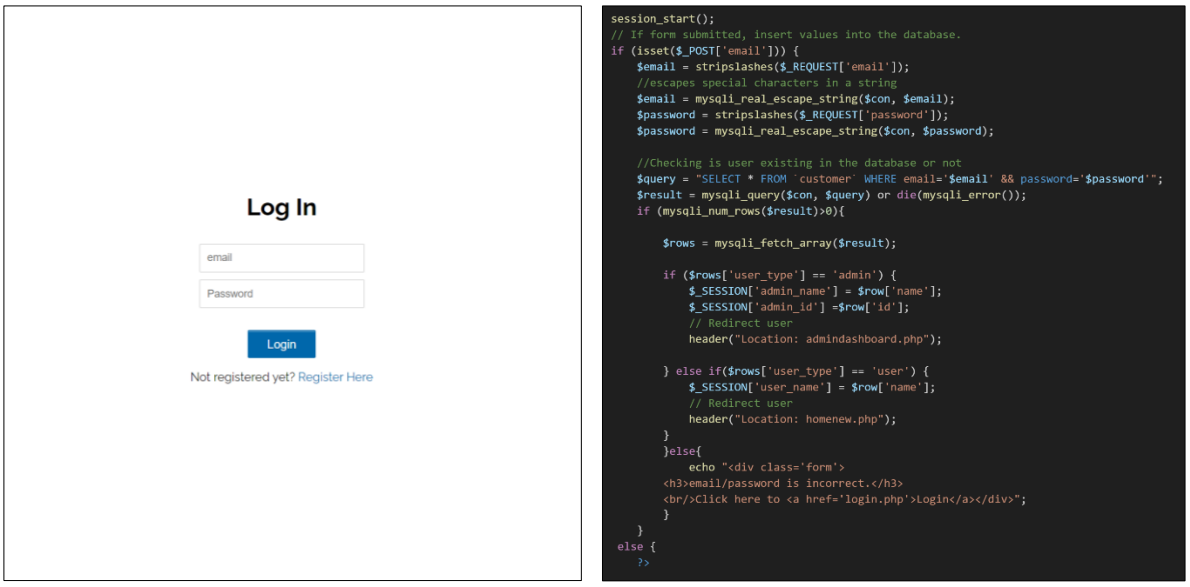

Figure 11: Login interface with a code segment

(iii) Manage mountain information

The administrator can manage the mountain of information that involves create, read, update, and delete (CRUD) functions. They can add new mountain information, view and edit the existing mountain in the database and delete the mountain from the database. The customer will be able to view the mountain information added in a more attractive and appealing interface. Besides this, customers can leave a review or comments on the mountain information as a personal opinion for hikers as a hiker. **Figure 12** shows the code segment for manage mountain information as an administrator and customer. **Figure 13** shows manage mountain information interface for the administrator, while **Figure 14** shows the manage mountain interface for the customer.

Figure 12: A code segment for manage mountain information as an administrator and customer

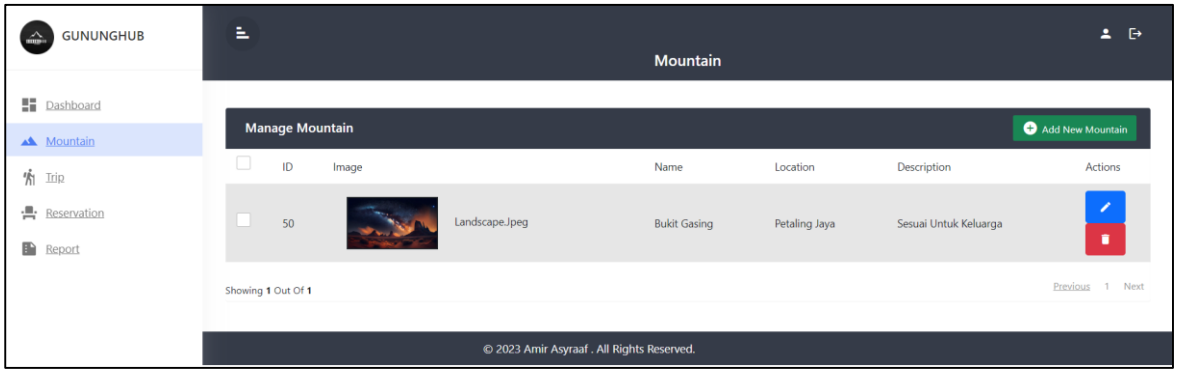

Figure 13: Manage mountain information interface for an administrator.

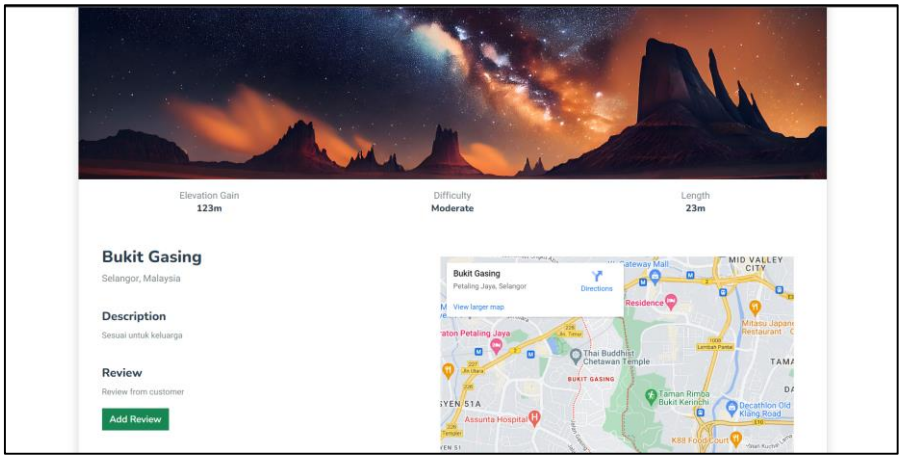

Figure 14: Manage mountain information interface for customers.

(iv) Manage reservation interface

The reservation process starts with an administrator providing a trip to the customer. Manage reservations requires the administrator to enter the trip information into the system. The administrator can also view, edit and delete the trip from the system. Subsequently, the customer can choose and select their desired trip and do the reservation by completing the booking form and payment process. Completing payment indicates the payment is successful. Then, the information about the reservation will be saved to the database. **Figure 1**5 shows the code segment for manage reservations as an administrator and customer. **Figure 16** shows the trip interface for the customer. **Figure 17** shows the booking form interface for customers. **Figure 18** shows the manage tip interface for an administrator.

Figure 15: A code segment for manage reservations as an administrator and customer

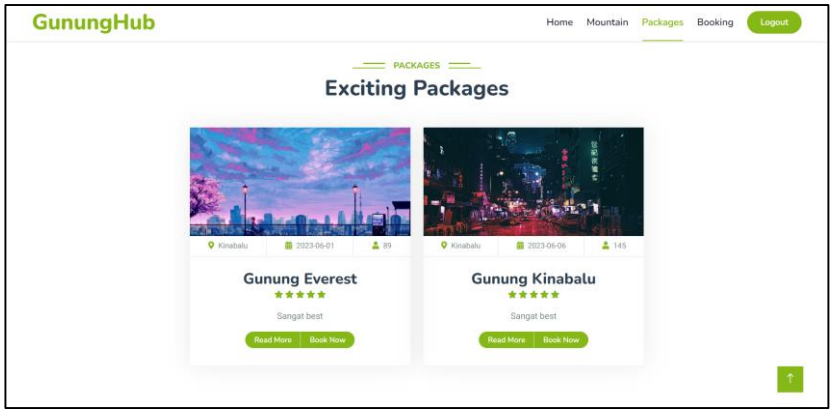

Figure 16: Trip interface for customer

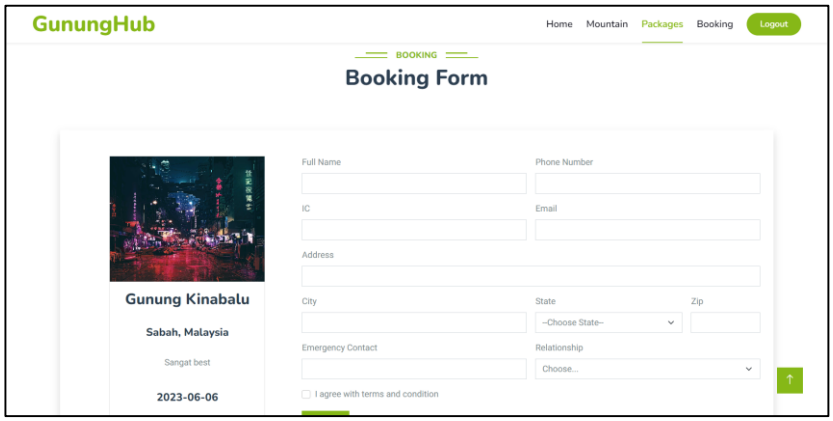

Figure 17: Booking form interface for customer

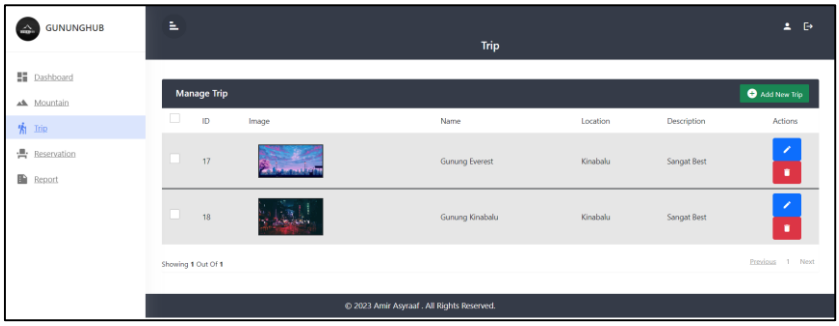

Figure 18: Manage trip interface for administrator

(v) Manage reservation history interfaces

Customer can view their reservation which will display the reservation information including the reservation status whether its upcoming, active, or completed. Meanwhile, the Administrator can also view the reservation that the customer made and delete it if necessary. **Figure 19** shows the code segment for reservation history as administrator and customer. **Figure 20** shows the manage reservation history interface for an administrator. **Figure 21** shows the reservation history interface for customers.

```php
if(isset($_POST['deleterecord'])){

    $id = $_POST['delete_id'];
    $sql = "DELETE FROM bookform WHERE bookform_id='$id'";
    $result = $conn->query($sql);

    if ($result == TRUE)
    {
        echo '<script> alert("Data Deleted"); </script>';
        header("Location:bookingadmin.php");
    }
    else
    {
        echo '<script> alert("Data Not Deleted"); </script>';
    }
  }
```

```php
<?php
$currentDate = date("Y-m-d");
$activeTripsQuery = "SELECT bf.*, t.trip_image, t.trip_desc FROM bookform bf JOIN trip t ON bf.trip_id = t.trip_id
WHERE bf.trip_date = '$currentDate'";
$activeTripsResult = $conn->query($activeTripsQuery);

if ($activeTripsResult->num_rows > 0) {
    while ($trip = mysqli_fetch_assoc($activeTripsResult)) {
        // Display active trips data
        displayTripCard($trip);
    }
} else {
    echo '<p class="text-center">No active trips found.</p>';
}
?>
```

Figure 19: A code segment for reservation history as administrator and customer

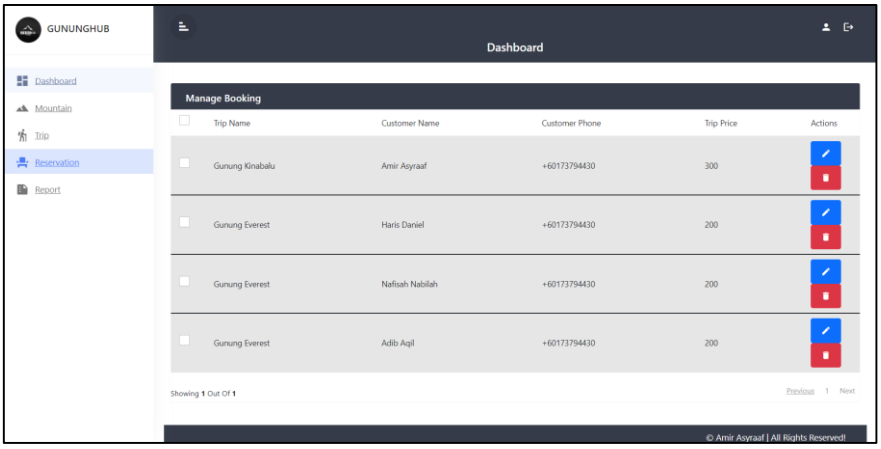

Figure 20: Manage reservation history interface. administrator

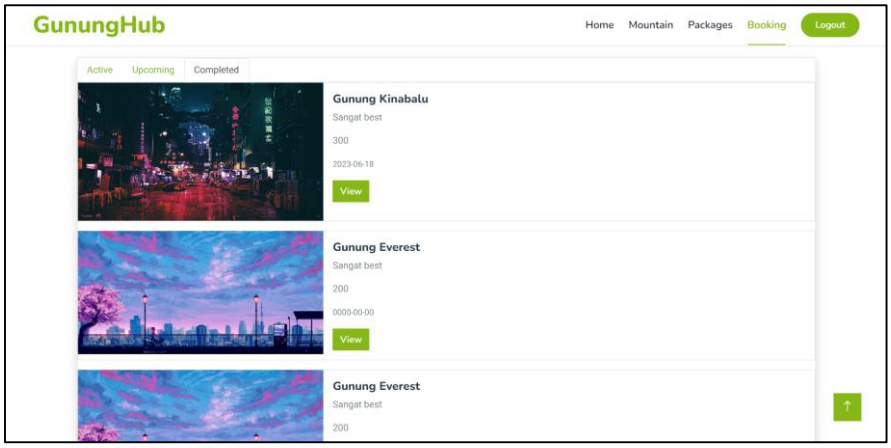

Figure 21: Reservation history interface for customer

(vi) Generate report interfaces

From trip information and booking information, the administrator can generate a report to see the data in a structured manner that has been digitalized. The generate report interface is only for an administrator to see the total sales for each trip. An administrator can select the report type either in Excel file, PDF file or print right away. Figure 22 shows the generate report interface with a code segment.

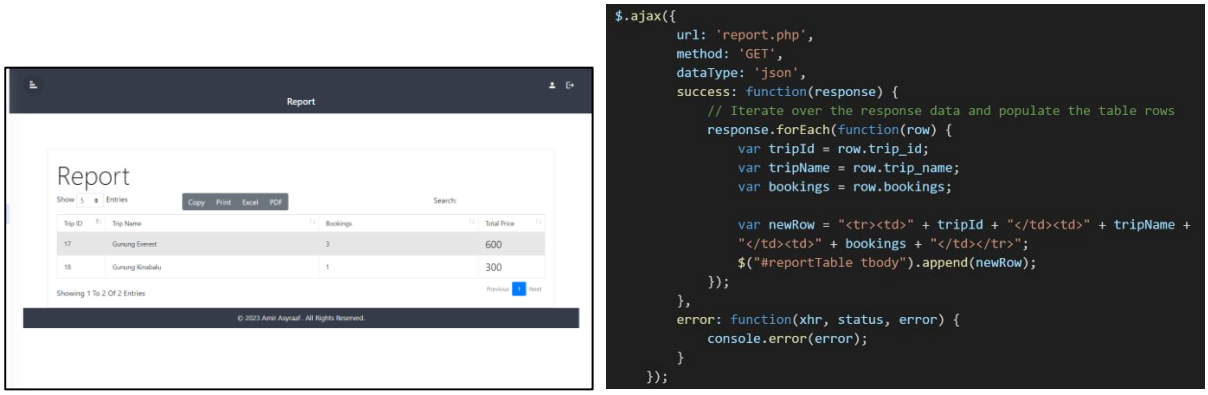

Figure 22: Generate report interface with a code segment

4.2 Testing

The User Acceptance Testing (UAT) and Functional Testing for the GunungHub – Guide and Reservation Online Application was conducted with a total of 27 test cases across different modules. The testing included registration, login, slot manage mountain information, manage reservations, manage reservation history, payment, and generate report. The questionnaires were designed using the Perceived Usefulness and Ease of Use (PUEU) instrument [11], a tool that allows us to gauge how users perceive the system in terms of its practicality and ease of use. Its purpose is to assess users' perceptions of a system or technology in terms of its usefulness and ease of use. This data helps in understanding users' attitudes and opinions about the system and can be used to identify areas of improvement or to validate the effectiveness of design changes. The testing results of the test cases that have been tested are summarized in **Table 5. Figure 23** shows the user acceptance testing result for evaluating the interface for hikers.

Table 5: Testing Results

Main Features	Total Number of Test Cases	Total Number of Pass	Total Number of Fail
Registration	3	3	0
Login	4	4	0
Manage mountain information	6	6	0
Manage Reservation	7	7	0
Manage Reservation History	4	3	1
Payment	2	2	0
Generate report	1	1	0

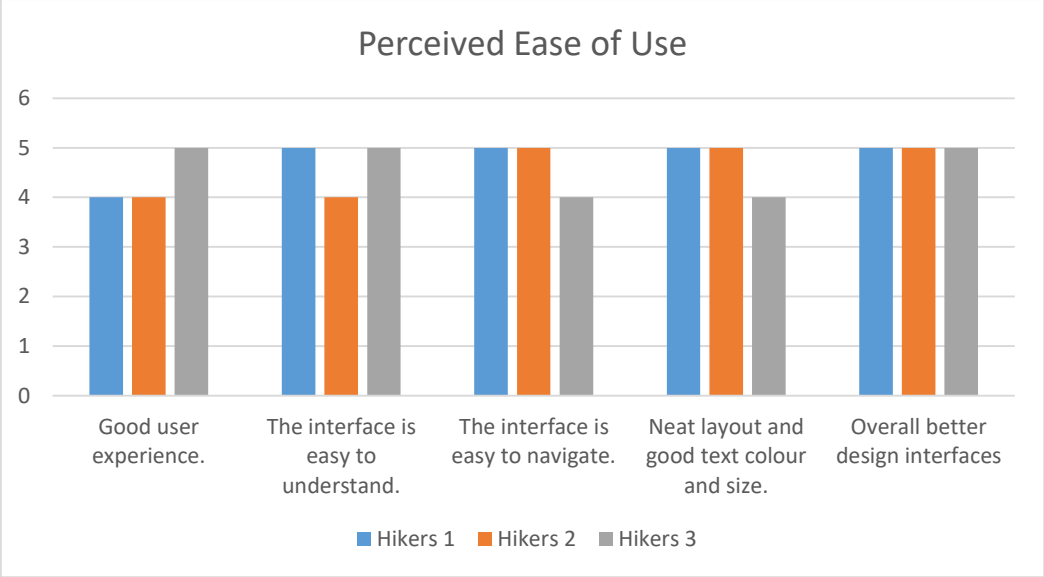

Figure 23: User acceptance testing result for evaluating interface for hikers

The fact that users found the user experience satisfactory, easily understood the interface, found it simple to navigate, appreciated the tidy layout and appropriate text color and size, and generally thought the design interfaces were better, suggests that users have a positive perception of the ease of use. **Figure 24** shows the user acceptance testing for evaluating functionality for hikers.

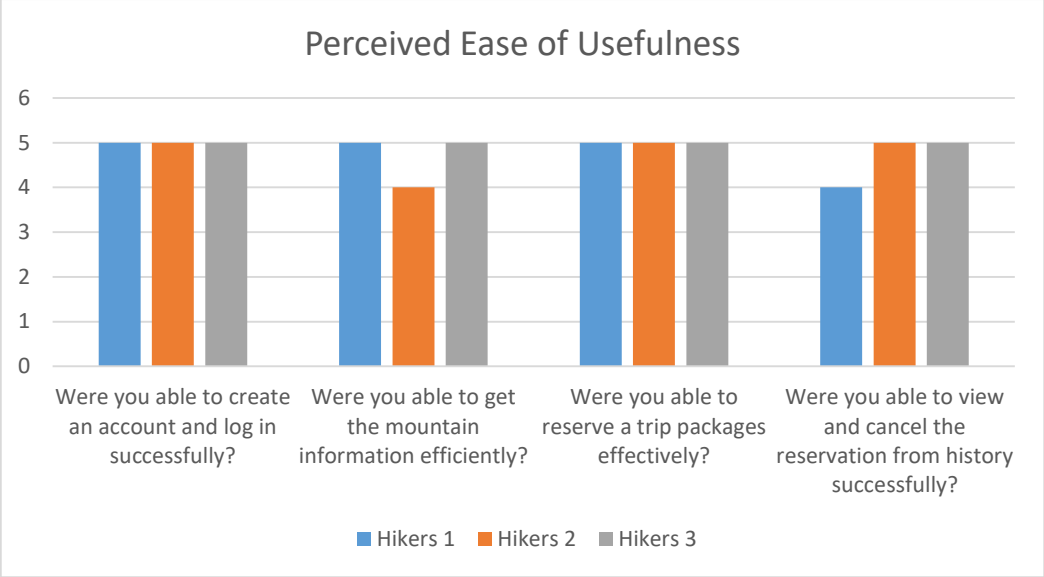

Figure 24: User acceptance testing for evaluating functionality for hikers

The respondents' unanimous agreement suggests a positive perception of the system's ease of use, as they were able to perform key tasks such as account creation, accessing mountain information, reserving trip packages, and managing reservations without any difficulties or complications. Figure 25 shows the user acceptance testing for evaluating the user interface for an administrator.

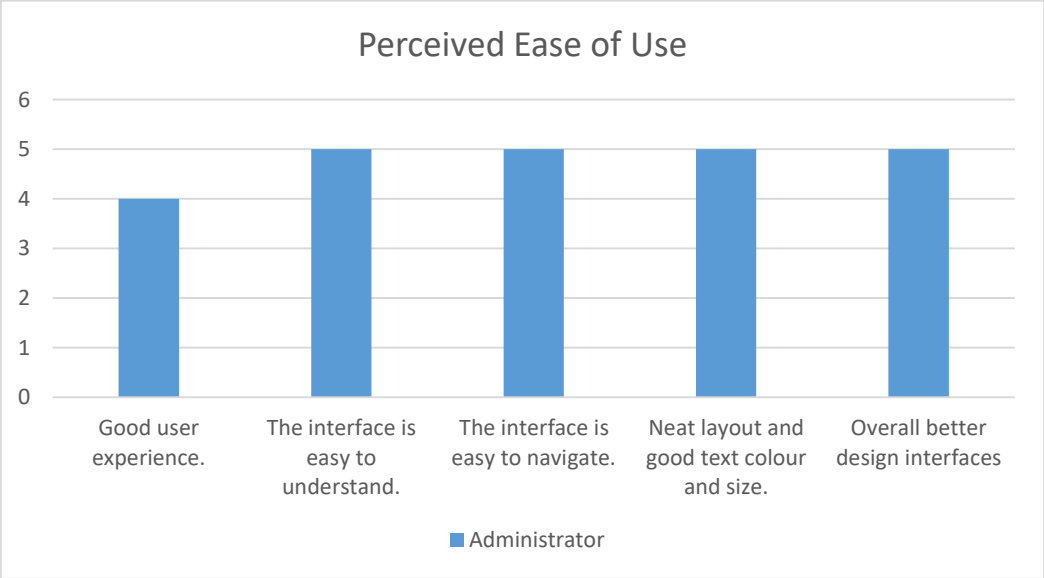

Figure 25: User acceptance testing for evaluating user interface for administrator

This indicates that the administrator perceives the system as having a positive user experience, finding the interface easy to understand and navigate. The administrator also appreciates the neat layout, suitable text color, and size, and considers the design interfaces to be overall better. The agreement of the administrator further suggests that the system is user-friendly and provides a seamless experience for customers. **Figure 26** shows the user acceptance testing for evaluating the function of an administrator.

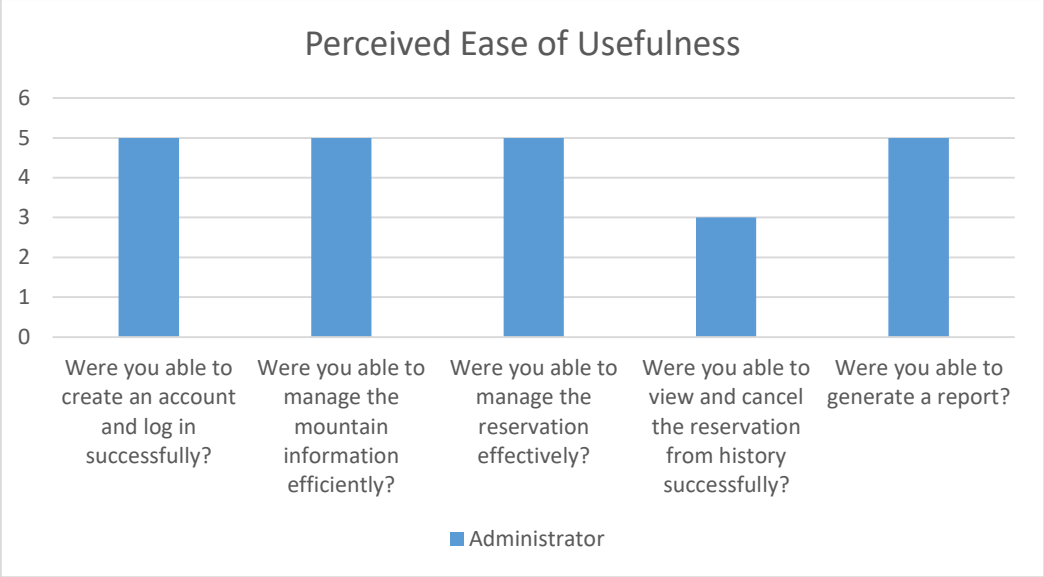

Figure 26: User acceptance testing for evaluating functional for administrator

The administrator's agreement with most of the questions indicates a generally positive perception of the system's ease of use. However, the difficulty in viewing and canceling reservations suggests a potential area for improvement. Further clarification is needed regarding the ability to generate a report.

5. Conclusion

Finally, the system development and design have enabled administrators and customers to benefit from it. Following an analysis of management difficulties with reservation trips and an analysis of the present system, several modules are proposed as alternatives. There are two types of system users: customer and administrator. The system was developed to allow for better and more efficient reservation management and minimize the mistakes occurring between the users. The mountain information serves as a guide to attract more individuals who like hiking and help the hiking community thrive. The Guide and Reservation Online Application is planned to eventually include all of the skills necessary to efficiently administrate and maintain an online booking system.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, UniversitiTun Hussein Onn Malaysia for its support.

References

[1] "Gh Gunung Hub Enterprise," Gh Gunung Hub Enterprise. [Online]. Available: https://central.mymagic.my/network/organization/3784/GH+GUNUNG+HUB+ENTERPRISE%C2%A0. [Accessed: Oct. 27, 2022]

[2] M. Jazayeri, "Some trends in web application development," in Future of Software Engineering (FOSE'07). IEEE, May 2007, pp. 199-213.

[3] W. Jobe, "Native Apps vs. Mobile Web Apps," International Journal of Interactive Mobile Technologies, vol. 7, no. 4, pp. 28, 2013.

[4] SabahParks (2023) *Mount Kinabalu, sabahparks*. Available at: https://reservation.sabahparks.org.my/ (Accessed: 16 June 2023)

[5] Project, M. (no date) *Rock climbing guides: Routes, Photos & Forum, Mountain Project*. Available at: https://www.mountainproject.com/ (Accessed: 16 June 2023).

[6] GAdventure (no date) *Bring on the world with small group travel., Adventure Travel & Tours - Book Your Trip - G Adventures*. Available at: https://www.gadventures.com/ (Accessed: 16 June 2023).

[7] R. A. Carter, A. I. Antón, A. Dagnino, and L. Williams, "Evolving beyond requirements creep: a risk-based evolutionary prototyping model," in Proceedings Fifth IEEE International Symposium on Requirements Engineering. IEEE, August 2001, pp. 94-101.

[8] K. Pohl, Requirements engineering: fundamentals, principles, and techniques. Springer Publishing Company, Incorporated, 2010.

[9] R. Klimek and P. Szwed, "Formal analysis of use case diagrams," Computer Science, vol. 11, pp. 115-131, 2010.

[10] A. Oliveira, V. Bischoff, L. J. Gonçales, K. Farias, and M. Segalotto, "BRCode: An interpretive model-driven engineering approach for enterprise applications," Computers in Industry, vol. 96, no. 1, pp. 86-97, 2018.

[11] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," MIS Quarterly, pp. 319-340, 1989.