iSKed: DEVELOPMENT OF FACILITIES AND EQUIPMENT RESERVATION MANAGEMENT SYSTEM FOR BARANGAY WESTERN BICUTAN

A Thesis Project
Presented to the Faculty of
Computer Studies Department
College of Science
Technological University of the Philippines - Manila
Ayala Boulevard, Manila

By

Corpin, Jade S.
Corpuz, Mark Jhay
Malic, Austine Bill Ryan N.
Ricerra, Jeremy Paul V.
Tuazon, Marivic O.

In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Information System

January 2025

		Index No.	TUPM-F-COS-16-TAU
TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES Ayala Blvd. Ermita, Manila 1000, Philippines	Tel No. +632-5301-3001 local 608 Fax No. +632-8521-4063	Email: cos@tup.edu.ph	Website: www.tup.edu.ph		Revision No.	00
THESIS APPROVAL SHEET FOR THE UNDERGRADUATE PROGRAMS OF THE COS		Date	05212025
		Page	1 / 1

This thesis hereto entitled:

iSKed: DEVELOPMENT OF FACILITIES AND EQUIPMENT RESERVATION MANAGEMENT SYSTEM FOR BARANGAY WESTERN BICUTAN

prepared and submitted by Jade S. Corpin, Mark Jhay Corpuz, Austine Bill Ryan N. Malic, Jeremy Paul V. Ricerra, Marivic O. Tuazon in partial fulfillment of the requirements for the degree Bachelor of Science in Information Systems, has been examined and is recommended for approval and acceptance.

PERAGRINO B. AMADOR JR.
Adviser

Approved by the Committee on Oral Examination with a grade of **PASSED** on January 17, 2025.

PRISCILLA S. BATOR
Member

DOLORES L. MONTESINES
Chairperson

Accepted in partial fulfillment of the requirements for the degree **Bachelor of Science in Information Systems.**

Date: **2 3 MAY 2025**

JOSHUA T. SORIANO, Ph. D.
Dean

Transaction ID	TUPM-COS-TAU-PBA-05212025-0700AM
Signature	

Transaction ID Legend: TUPX AAA (Office Code) BBB (Type of Transaction) CCC (Initial of employee) MMDDYYYY (month day year) HH:MM:AM/PM (hour:minutes:AM/PM)

ACKNOWLEDGEMENT

ABSTRACT

Businesses have widely embraced online reservation systems, offering enhanced efficiency and innovative solutions for managing various types of reservations. However, the SK Council Members of Western Bicutan still rely on a hybrid approach of manual and online methods for handling reservations. The challenges they face in managing these reservations highlight the need for a more efficient and streamlined reservation system. The study, titled "iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan," aims to improve the management of reservations and replace the existing processes. The system enables administrators to efficiently manage events, programs, reservations, inventory, and users, as well as generate reports. It s designed to be user-friendly and flexible, allowing administrators to maintain the website without requiring coding knowledge. The system simplifies the reservation process and includes features to prevent double booking, ensuring a smooth and reliable user experience. The study used Agile Scrum to develop, ensuring iterative progress with regular feedback. The system was evaluated using the ISO 25010 evaluation framework, focusing on Functional Suitability and Reliability to assess its performance. The result shows that the system is highly effective, meets necessary requirements under different conditions, is reliable, and works as expected. The system received an overall rating of 3.73, described as "Highly Acceptable," based on feedback from thirty (30) respondents. The study showed that developing the "iSKed" system improved the reservation process, enhancing operational efficiency and addressing the challenges faced by the SK Council.

Keywords: Reservation Management, Sangguniang Kabataan, Facilities and Equipment Booking, Online Scheduling, Inventory Monitoring

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

Chapter 1

THE PROBLEM AND ITS SETTING

Introduction

The Sangguniang Kabataan (SK) Reform Act, also known as Republic Act (RA) 10742, was passed in 2016 to introduce critical improvements to the Sangguniang Kabataan system. The SK was the Philippines' counterpart of a village youth council, with eight elected representatives representing each barangay: one chairman and seven Kagawads, or SK Council members, all between the ages of 18 and 24. These young leaders were responsible for representing their constituents while participating in policy formulation and program creation within their areas. The Sangguniang Kabataan's primary purpose was to actively engage youth in nation-building and local governance, supporting their well-being through effective and responsive procedures and activities (Flores et al., 2021).

Over the years, the Sangguniang Kabataan established several initiatives, events, and services aimed at empowering youth and promoting community development. These efforts covered education, leadership development, sports, health, and environmental awareness (National Youth Commission, n.d.).

In Taguig City, notably in Western Bicutan, the Sangguniang Kabataan often organized leadership projects and seminars and sponsored several sports leagues. The SK Council of Western Bicutan offers a variety of activities, including the TeenBayan fitness center, where youth can make reservations, a Printing Service, which allows individuals to submit data or photographs for free printing, and even equipment bookings. In addition to these

events, the council offered a variety of registration-required programs, uncluding a basketball competition and a mobile legend tournament.

Online applications and systems can streamline processes as technology advances and is used more frequently. It became possible to automate and optimize tasks traditionally completed by hand, which were ineffective and prone to human error. Real-time updates, rapid information access, and more efficient workflows were made possible by this digital revolution. For instance, online availability checks, service bookings, and confirmations were all possible with event management and reservations. Moreover, administrators oversaw and managed these procedures with a great deal less human involvement, guaranteeing accuracy, effectiveness, and a better experience for both organizers and users.

The iSKed was an apprised online reservation system designed to improve accessibility and eliminate inefficiencies for youth in Western Bicutan. Through this, the online reservation system enhanced the user experience, and also the SK Council was able to better serve the Western Bicutan youth. With this study, the SK better fulfilled its mission of inclusivity and effective service delivery, paving the way for a brighter future for the youth in the area.

Background of Study

The Sangguniang Kabataan played a vital role in the Philippines by advocating for youth interests and involving them in local governance (Castillo et al., 2020). The Sangguniang Kabataan served as the barangay youth council in the country, focusing on creating and promoting initiatives that addressed the needs of young people. It was

regarded as a venue for aspiring leaders to develop their public service skills (GMA Integrated News, 2023).

Youth participation in community-building was often criticized for the limited roles governments offered. While many sports councils emphasized athletics, some had successfully developed education, health, and civic engagement programs. This indicated a need for councils to better address community needs and foster unity (Rosales, 2023). Proposed initiatives included environmental, health, and educational projects, such as free mental health consultations, printing services for educational materials, and disaster risk training. Given the complex educational challenges, including poverty and the digital divide, initiatives like scholarships and learning hubs with tutors in conflict-affected areas supported students significantly (Ramiscal, 2023).

The Sangguniang Kabataan (SK) Council of Western Bicutan utilized a manual reservation process to manage various activities and events for the youth in their community. This approach involved individuals securing participation slots through face-to-face interactions and written requests; some were through the use of Google Forms, which both the council and youth encountered difficulties using. The process began with an announcement of available programs or events, which was disseminated through community boards, social media, or word of mouth.

One key aspect of this manual system was its emphasis on personal interaction. By engaging directly with the youth, council members better understood their needs and preferences, fostering a sense of community and trust. This direct communication also allowed for immediate feedback, enabling the council to adapt programs based on

participants' interests. Manual reservation was particularly beneficial for those without access to technology or internet services. It ensured that all youth had equal opportunities to participate in SK activities regardless of their digital literacy or resources. This inclusivity was vital for promoting community engagement from a diverse range of youth. The direct interaction fostered relationship-building between staff and customers, allowing for a more personalized experience. Furthermore, manual reservation methods were a cost-effective solution for small businesses that could not afford complex automated systems (Baker & Hsu, 2021).

Regardless of this, the manual reservation process used by the Sangguniang Kabataan (SK) in Western Bicutan posed several challenges that impacted youth engagement and operational efficiency. This system often led to miscommunication, scheduling conflicts, and errors, resulting in frustration for youth and potentially discouraging their participation in activities. The time-consuming nature of the manual process deterred tech-savvy young individuals who preferred quick, efficient solutions. Additionally, the lack of real-time tracking and updates limited access to information, further diminishing participation rates. The aforementioned process of Sangguniang Kabataan in Western Bicutan led to a higher risk of errors, limited accessibility, and time-consuming operations. The existing manual reservation process was fraught with inefficiencies, resulting in frequent data entry errors and scheduling conflicts. Such a manual system not only proved tedious and complicated but also hampered operational efficiency (Eunifridah et al., 2021).

One significant issue was the potential for human error; mistakes in entering information led to double bookings or lost reservations, negatively impacting customer

experience (Murphy, 2020). A direct example of this was their process of reserving their facility using Google Forms. Users often neglected to check the availability link, which resulted in overlapping appointments and further complicated the reservation process. The need to check a separate link for schedule availability caused inconvenience, leading to user frustration and making the system seem inefficient. Consequently, many individuals discovered that their chosen appointment time was unavailable only upon arrival, leading to confusion and disappointment. Youth often neglected to check the availability link when booking appointments, which led to overlapping reservations and complicated the reservation process. The inconvenience of consulting a separate link for schedule availability frustrated users, making the system feel cumbersome and inefficient (Kumar et al., 2021). This oversight not only impacted user satisfaction but also placed additional strain on service providers who managed these discrepancies, emphasizing the need for more integrated and user-friendly reservation systems (Zhao et al., 2022).

This situation forced users to reschedule in person, which added further inconvenience. The need for additional manual intervention to notify users of unavailable time slots increased administrative burdens and contributed to user dissatisfaction. Without an integrated and automated system to manage bookings and confirm availability in real time, the SK of Western Bicutan faced significant operational challenges that hindered the effective execution of its programs. The necessity for users to reschedule appointments in person added significant inconvenience, particularly when they were unaware of unavailable time slots until they arrived (Bharadwaj et al., 2020). This situation not only increased the administrative burden on staff who manually notified users of these conflicts

but also contributed to overall user dissatisfaction (Lee & Lee, 2023). Without an integrated and automated system to manage bookings and confirm availability in real-time, organizations like the SK of Western Bicutan faced considerable operational challenges, which hindered the effective execution of their programs and services (Nguyen et al., 2021) and could lead to confusion and disappointment (Siddiqui et al., 2023).

With all of this, it highlighted that manual reservation posed several challenges, primarily due to its time-consuming nature and reliance on face-to-face interactions. This led to delays, miscommunication, and scheduling conflicts, especially when information was recorded manually, increasing the risk of errors and inefficiencies.

While fostering personal interaction and inclusivity, the Sangguniang Kabataan of Western Bicutan also faced challenges, including miscommunication, scheduling conflicts, and participant dissatisfaction caused by human error and the absence of real-time updates. The reliance on manual processes, compounded by inconsistent use of technology, further contributed to declining youth engagement. These issues underscored the need for an automated system to streamline operations, minimize errors, and enhance the overall experience for participants.

Objectives of the Study

General Objective

The main objective of the study was to develop a web-based reservation system that aimed to streamline facilities and equipment management, simplify reservation processes, and enhance user engagement.

Specific Objectives

The following were the specific objectives of the study:

1. Create and design a reservation system with the following features and characteristics:

    a. Integrated a dynamic service creation module that allowed adding and updating new services, including links for other varying programs offered to the community, and highlighted their achievements.

    b. Optimized gym appointments and utilization schedules by automating the scheduling and allocation of facilities and equipment and displaying the calendar of appointments.

    c. Measured facility and equipment utilization, service popularity and usage, reservation trends, and user behavior analytics, including user satisfaction.

d. Ensured compliance with SK's policies and regulations, particularly on the use of facilities and equipment, by integrating appropriate approval processes into the reservation system.

e. Assessed the impact of the reservation and service management system in managing resources and organizing community events.

f. Generated reports for decision-support management of the users, facilities, equipment, and inventory by forecasting resource availability and maintenance needs.

g. Generated a unique code for each youth resident based on SKWB official records, requiring manual retrieval from the barangay office to complete account registration and ensure secure, verified access.

h. Implemented an inventory monitoring system to track the status, availability, and maintenance requirements of equipment.

i. Integrated a feedback feature accessible within the user dashboard, which focused on assessing user satisfaction, gathering input on the usability and functionality of the website, and improving program delivery in line with government standards, ensuring transparency and continuous improvement.

    j. Facilitated secure password management through an email verification process, which enabled users to update their credentials independently.

2. Develop the system using the following software development tools:

    a. User Interface:

- Figma

    b. Version Control and collaboration:

- GitHub

    c. Front-end Development:

- ReactJs
- CSS
- Bootstrap

    d. Back-end Development:

- ExpressJs
- JS
- NodeJS
- Postman
- SMTP

e. Database Management System:

- PostgreSQL

- Cloudinary

f. Integrated Development Environment (IDE):

- Visual Studio Code

g. Deployment:

- Heruko

- Render

3. Determine the level of acceptability, efficiency, functionality, and suitability to assess the system in meeting user needs and facilitating smooth operations.

4. Test and evaluate the acceptability of the application by having multiple users operate the system. The assessment tool used was ISO25010 to evaluate the application's sustainability, timeliness, accuracy, and maintainability.

Scope and Limitations of the Study

The study focused on developing iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan, a web-based platform designed to improve the efficiency of managing barangay facilities and equipment

reservations. The platform streamlined reservation processes provided insightful data for planning, and ensured transparency and compliance with SK policies.

The system included a dynamic service creation module that allowed administrators to add or modify reservation events and update service listings, ensuring the platform's adaptability to the community's evolving needs. Links to other varying programs offered by the barangay were also added, along with the ability to highlight their achievements. However, making timely updates is dependent on manual updates from the administrators. Key functionalities of the platform included an automated scheduling module with a calendar view that prevented double bookings, optimized gym appointments, and allocated facilities and equipment efficiently. Since the current SKWB process allowed users to book a facility whenever a slot was available, the system did not require reservations, streamlining the booking experience for users. However, for the equipment reservations, users were required to book in advance, as the system did not support real-time or immediate bookings. The system tracked utilization metrics such as frequent reservations and trends, generating data-driven resource allocation and planning insights. These reports enabled officials to forecast resource availability and maintenance needs, although the system relied on administrators to analyze and act on the data.

A unique code based on SKWB official records was generated for each youth resident. Users were required to retrieve this code manually from the barangay office to activate their accounts and enhance security. The system also facilitated secure password management via an email verification process, allowing users to update their credentials independently. A feedback feature was integrated within the user dashboard to assess user

satisfaction, gather input on usability, and improve program delivery in line with government standards, ensuring transparency and continuous improvement. Additionally, the system tracked equipment availability and maintenance requirements, ensuring operational readiness. It also supported inventory monitoring to track the status and availability of resources, helping to avoid overuse or neglect of equipment.

Furthermore, the system incorporated the ability to generate detailed reservation and utilization reports, enabling officials to assess resource usage trends, identify overused or underused facilities, and plan accordingly. These insights also contributed to forecasting maintenance needs and enhancing long-term planning. The system assessed the impact of its implementation on resource management and community event organization, aligning with SK-specific policies to ensure effective operations.

Despite its advanced features, the system had several limitations. The system relied heavily on administrators to provide timely and accurate updates. Failure to log reservations or update availability data led to inefficiencies, such as scheduling conflicts or underutilized resources. While the system generated insightful reports, it lacked real-time monitoring or automated adjustments, making administrators responsible for overseeing operations and ensuring optimal resource use. As a web-based platform, the system's functionality depended on stable internet connectivity. Interruptions affected user access to reservation features and verification processes.

Significance of the Study

In the current generation, the utilization of the media landscape is essential to many human activities. The success of programs and services is eventually impacted by the manual process's frequent errors, inefficiency, and length, which cause scheduling issues, additional work for staff, and decreased user satisfaction. In order to reduce the complexity, having a platform encouraged participation from Western Bicutan's young.

This study offered the SK Youth Council a thorough plan for switching to a more contemporary online reservation system. The council decreased mistakes related to manual processes, lessened administrative responsibilities, and increased operational efficiency. The council was able to properly address the requirements and preferences of the youth of Western Bicutan due to the availability of real-time data, which facilitated better decision-making and program planning. Additionally, developed targeted initiatives, boosting community impact and involvement, if it had the ability to gather and assess user data.

The system offered a valuable solution for the youth users who struggled with the current manual reservation process. This online platform provided a user-friendly interface for making reservations, ensuring accessibility and convenience. By having real-time scheduling, the system significantly reduced the likelihood of conflicts. Also, this web application fostered a sense of community by keeping the user informed regarding upcoming events and even programs, which encouraged greater participation and involvement in local governance and initiatives.

Future researchers could build upon the findings of this study to investigate various dimensions of digital transformation in community settings. They might explore how such

systems affect user satisfaction and the efficiency of administrative processes. Moreover, they could utilize this study as a reference if they undertake similar studies, for as such, the findings and insights can serve as a valuable foundation, challenges, and outcomes related to the implementation of online reservation platforms in community settings and could be their research design and have an informed analysis in a comparable context.

This study highlighted the potential for a modern reservation system to enhance the operational effectiveness and engagement of the youth in Western Bicutan.

Chapter 2

CONCEPTUAL FRAMEWORK

This chapter lays the groundwork for iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan. It contains a compilation of relevant literature and studies related to developing an online reservation system. The chapter also outlines the study's conceptual model and defines key terms for a clear understanding. This review provides a foundation to identify any gaps in existing knowledge.

Review of Related Literature

Introduction

The Sangguniang Kabataan (SK) of Western Bicutan has been actively organizing and facilitating various programs aimed at the holistic development of the youth. However, similar to other manual reservation systems, the existing process for registering and scheduling these programs was cumbersome and inefficient.

Traditionally, the reservation system for SK programs involves multiple steps that often result in errors, overlapping schedules, and administrative burdens. Users are required to fill out forms and manually check a separate link to view available time slots, which many skip, leading to scheduling conflicts. These inefficiencies consume time and create frustration for both SK councils and users. Additionally, the absence of a streamlined, real-time reservation platform limits the council's capacity to focus on other important responsibilities, such as planning future events, maintaining facilities, and engaging more

proactively with the youth.

Fortunately, the development of online reservation systems offers a viable solution to these challenges. Studies have shown that automated systems significantly improve user satisfaction and administrative efficiency. These platforms enable users to view available slots and easily make reservations in real time. Below is the related literature and related study on the development of the iSKed.

Reservation Systems

According to Omnify (2020), an online and free platform for class registration and scheduling, a reservation system was software mainly used to manage client service reservations. In line with the definition, there were a large number of reservation systems on different platforms that defined and provided a more structured exploration of reservation systems. A study from Perdana and Sancoko(2023), entitled Android-Based Reservation System at Dharmarini Health Center, explained and categorized which platform to deliver the system, the success rate, and the necessary needs to conduct the study. The system had been mainly an Android-based platform. It used Kotlin for the backend, Node.js for the server side, Firebase for the database, and React.JS for the frontend framework. The researchers also added features and objectives to the application, such as the health center's profile, which enabled users to make reservations and access necessary information. The category for check-ups and monitoring also reduced crowd congestion in the actual health center for medical check-ups by allowing users to save their slots. It allowed staff to manage all the patient's data.

This study was relevant in showing how the researchers could conduct the development of a web-based reservation system by gathering necessary information and referencing key features that would be beneficial for the actual study. In line with this, a project report conducted by Acharya(2024), titled Online Bus Reservation System Project Report, provided a broader overview of the development of an online bus reservation system, which included electronic ticketing. It aimed to innovate the manual and traditional ticketing systems in public transport. While this report focused on buses and ticketing, it served as a support reference for developing a web-based reservation system. According to Aldisa (2023), another study on web-based rapid application development for hotel reservations has been conducted. It stated the relevance and objectives of the hotel reservation system, which simplified and modernized the traditional business processes of searching for information or making inquiries about available hotels and bookings.

Perceptions and Participation of the Youth on Sangguniang Kabataan (SK)

The Sangguniang Kabataan (SK) was created through the Local Government Code of 1991 (Republic Act 7160), which provided an opportunity for young people to directly participate in local governance, composed of all youth aged 15 to 21 (National Youth Commission, n.d.). The specific functions of the SK included but were not limited to initiating programs to enhance social, political, economic, cultural, spiritual, and physical development; communicating barangay resolutions related to youth development; and coordinating the implementation of national youth development projects.

Research findings by Castillo et al. (2020) revealed that while respondents acknowledged the crucial role of the Sangguniang Kabataan (SK) in representing youth within local government, there were significant areas for improvement, particularly in developing youth-focused programs. The study also highlighted a notable lack of awareness regarding SK events and services. These findings emphasized the necessity of an accessible website that enabled youth to view announcements, upcoming events, and programs organized by the SK, thereby enhancing engagement and participation in local initiatives.

Additionally, a study by Palangdao et al. (2023) indicated that conducting youth profiling had been essential for accurately identifying the needs and preferences of young individuals. This finding reinforced the importance of incorporating a profiling feature within the proposed online reservation system. By collecting information about interests, needs, and demographics, the SK could design tailored programs that resonated with constituents, ensuring higher participation and engagement. This profiling also facilitated more effective communication, allowing the SK to send personalized announcements and updates based on user's preferences.

Scientific Review of Reservation System

Based on the findings of Permatasari et al. (2023), reservation systems have enhanced business and customer satisfaction. The system enhanced operational efficiency by streamlining scheduling processes, reducing conflicts, and optimizing resource utilization, thereby ensuring smoother operations and better service delivery. Customers benefited

from improved convenience and accessibility, allowing them to make reservations, view real-time availability, and receive timely confirmations and reminders.

Furthermore, the system enabled personalized services tailored to individual preferences, fostering greater customer satisfaction and engagement. Through effective data management and analytics, businesses gained valuable insights to inform decision-making and enhance marketing strategies. Integrated features such as reminders for cancellations and no-shows contributed to reducing losses and improving time utilization efficiency. Moreover, the trend of online reservation systems ensured better accessibility for customers, while integration with other management systems enhanced overall operational efficiency and data management. The easy-to-use interface also contributed to customer engagement, helping them to book appointments stress-free.

Additionally, according to Shakirah and Devis (2024), the impact of online reservation systems has significantly contributed to boosting business profitability and customer satisfaction. Integrating websites into enterprises provided customers with the necessary information and became a modern advertising tool. It facilitated strategic planning, reduced costs, improved competitiveness, and strengthened customer relationship management. The study also noted that websites positively affected financial performance by increasing customer satisfaction and trust.

Importance of an Online Reservation System

An online reservation system had been a software tool that helped businesses efficiently manage bookings for various services like pools, fitness centers, gyms, yoga studios, and

parks & recreation centers. The system enabled service providers to accept online bookings and appointments, simplifying the process of handling reservations both online and offline. Customers could easily reserve and pay for services through a website, selecting slots, dates, and times, making payments, and managing memberships online. The system reduced staff workload and the chances of double bookings (Omnify, 2020).

Based on the study Key Dimensions of Web-Based Reservation Systems in the Modern Tourism Sector by Marinova (2023), reservation systems have been widely utilized in the tourism industry as electronic tools that enhance sales and provide quick and accurate information on availability, pricing, and booking of travel services. Initially developed by airlines, these systems now catered to various tourism products, enabling reservations for accommodations, dining, flights, tours, and events. The systems created individual records for guests/passengers, facilitating personalized offers based on user preferences. Real-time processing remained a crucial feature, ensuring immediate responses to user requests.

Faridah et al. (2023) emphasized that small businesses can leverage online reservation platforms to strengthen their digital presence and compete with larger enterprises. By offering online booking options, companies can expand their customer reach and improve accessibility. Similarly, Oscar et al. (2023) noted that online consumers benefit from a wider selection of services, competitive pricing, and greater convenience, allowing them to make well-informed purchasing decisions.

In the current study, the importance of a reservation system had been paramount in streamlining the booking process for the services and programs provided by SK Western Bicutan. Implementing iSKed enabled the SK Youth of Western Bicutan to efficiently

manage sports facilities by allowing users to reserve preferred time slots and dates online, mitigating the risk of overlapping bookings and ensuring optimal resource utilization. Additionally, the platform enhanced the user experience by offering convenience and flexibility in reserving events or services, fostering positive interactions between users and the councils. The real-time availability feature allowed users to view open slots and make reservations accordingly, enhancing transparency and accessibility. Data tracking and analysis through the system enabled the SK Council of Western Bicutan to gather valuable insights into booking patterns, empowering informed decision-making and improving service offerings.

Impact of Online Reservation System

An online reservation system solved the problems of the online booking function of sports venues, improved the efficiency of the venues, and met the needs of efficient use of the venues (Li et al., 2019). According to the side of the users who use the system, a study was conducted wherein the user assessed the overall usefulness and efficiency of an online reservation. The study results showed that the online platform was very easy to use. Additionally, having an online website platform will be able to monitor the reservations and the regular customers for their business (Agoda, 2023).

According to the study "Effect of an Online Appointment Scheduling System on Evaluation Metrics," using an online scheduling system will unquestionably assist staff in managing their time as well as in keeping up with things that are vital to the company (Habibi, 2019). Since it is online, this will increase the system's flexibility (Guan et al.,

2020). Furthermore, in terms of reducing data redundancy, inconsistent data in reservations, and paperwork, an online reservation system also saves the time and effort of the client (Marksman Drydale Travel, 2023). The beneficiary recognizes that having such websites will benefit their business. Overall, an online web-based reservation system is a comprehensive reservation system for a wide range of activities, an automated method of securing accommodation for its client or the user to avoid scheduling conflicts when using selected events, programs, and services.

Responsive Web Application

According to Olle Lundmark (2022), in Responsive Web Design for Modern Devices, the research discussed the approach of the study to investigate the differences between established and modern devices regarding their interaction with a webpage. The goal was to identify what methods of responsive web development worked best with these devices and what might become an issue when newer devices were released and widely used. Additionally, the study stated that a web application with good responsiveness adapted well to different device screen sizes and aspect ratios and displayed the content without removing or hiding important information. The research targeted prospective or new web developers, providing insights into how web responsiveness had evolved and how developers adjusted to today's devices. Furthermore, the study revealed that the probable advancement of web-enabled devices was leaning toward mobile devices taking over the market, as indicated by historical data, and the number of devices that could access a web browser continued to grow.

The study by Gurmeet Singh, Ruchi Rathi Saboo, and Ritesh Mathu (2019), entitled Responsive Web Design with HTML and CSS, discussed the responsive web layout, focusing on the symbiotic relationship between HTML and CSS in creating adaptable and user-friendly interfaces across various screen sizes and devices. The research mentioned that responsive web design (RWD) had emerged as a transformative approach to web development, fundamentally changing how websites were designed and accessed across a myriad of devices. The research examined the ideas, strategies, and best practices of responsive design.

The research, Responsive Redesign and Its Effects on Perceived Usefulness, by Kimberly Fischer and Damian Schofield (2021), described the introduction of a new website at TCGplayer, an online retailer of collectible Magic the Gathering cards. The paper described an experiment that tested a new, responsive design against the current, non-responsive design. The research highlighted the essence of responsive design, showing that most shoppers preferred mobile phone access to e-commerce and m-commerce sites. It was also demonstrated that consumer loyalty was higher for m-commerce sites that were responsive, aesthetically designed, and built from the user's perspective.

JavaScript

JavaScript (JS) is one of the most popular programming languages and the main scripting language for web browsers, and it is essential to modern web applications (Dou, 2020). While there were many choices for constructing the server, most client-side interactions were done in JavaScript, with 96.8% of client-side programming done in JS.

For this reason, JavaScript has been ranked as the most popular technology for eight consecutive years by Stack Overflow (Stack Overflow, 2024). JS was typically regarded as a high-level programming language due to its minimal interaction with the operating system.

In the development of a reservation management system by Shun and Hassim (2021), one of the languages they used was JavaScript, where JS played a vital role in creating dynamic elements, enabling real-time feedback on user actions, and aiding in interactive elements throughout the development process.

JavaScript was an interpreted language, meaning the code did not need to be compiled before execution (Tomar et al., 2021). This led to a decrease in overhead and an increase in the speed of execution of JavaScript code on web pages. The study also discussed that JS was a platform-independent programming language, which meant that it could run on any platform, making it a versatile and easy-to-use language. JavaScript empowered users to engage with the platform through functionalities like selecting dates, filtering availability, and submitting reservation requests.

React JS

ReactJS was an open-source library used to build UI specifically for single-page applications (Sousa et al., 2020). ReactJS empowered software engineers to create large web applications that could use data and change over time without reloading the page. In a study conducted by Venkat in 2021, it was explained that web development technology before 2015 focused on scripting and rendering. At that time, HTML and CSS were used

for the front end, while PHP was employed for the back end. Developers placed static HTML pages in some folders and rendered those files using PHP. Developers relied on this method for decades, so there was no significant revolution in a website until the introduction of a JavaScript library like React.js (Fariz, 2022). Additionally, in ReactJS, each component manages its state and transforms it into UI elements (Rawat, 2020).

Developers utilized this framework to build reusable UI components, and this modular approach enhanced both development efficiency and UI maintenance. Moreover, ReactJS features a virtual DOM (Document Object Model), allowing ReactJS to efficiently detect and update only the necessary UI elements when changes occur. By leveraging ReactJS, developers designed an interactive interface for iSKed, simplifying the reservation process for users.

Cascading Style Sheets (CSS)

Cascading Style Sheets, known as CSS, were a web-based style language used to present web documents (Ndia et al., 2019). CSS consisted of a sequence of style rules, each with a selector that targeted the elements to style in HTML or XML documents. According to Simo Kuparinen (2023), CSS was employed to style and layout web pages, such as altering the font, size, color, style, and other decorative elements. CSS performance reflected how much of the total load time and layout calculation depended on scripting, loading, and painting.

The proposed study utilized CSS to ensure a smooth user experience. It controlled the layout positioning of elements like calendars and forms and facilitated easy navigation.

CSS also dictated the visual design, enabling a branded look that reflected the SK Youth Programs portal.

PostgreSQL

PostgreSQL was an open-source object-relational database system built to manage complex data workloads reliably and efficiently. Originating from the POSTGRES project at the University of California, Berkeley, in 1986, it underwent over 35 years of continuous development. Known for its robust architecture, reliability, data integrity, and extensibility, PostgreSQL supported a wide range of data types, advanced indexing techniques, and essential features such as ACID compliance, along with backup and recovery capabilities. As of its version 16 release in September 2023, PostgreSQL adhered to a significant portion of the SQL:2023 Core standard, reinforcing its position as a leading open-source relational database. It offered developers and administrators powerful features, including customizable data types, procedural language support, advanced security controls, and internationalization capabilities. Highly scalable, PostgreSQL efficiently managed large datasets and numerous concurrent users, making it suitable for applications ranging from small projects to enterprise-level systems handling petabytes of data (PostgreSQL, 2024).

Based on the book Learning PostgreSQL 11 by Salahaldin Juba and Andrey Volkov (2019), PostgreSQL was an open-source object-relational database management system that competed with leading databases such as Oracle, MySQL, and SQL Server. Its licensing model allowed commercial use without restrictions, making it a popular choice for startups, as numerous companies provided commercial support. With extensive

extensions, PostgreSQL was widely used in research and served as the underlying technology for several other database solutions, including Greenplum and Amazon Redshift. The database was compatible with most modern operating systems, such as Windows, Mac, and Linux, and offered straightforward installation and configuration processes via common package managers or interactive installers. PostgreSQL adhered to ANSI SQL standards, facilitating ease of use for developers and database administrators, supported by comprehensive documentation and a robust community. It was versatile for both online transaction processing (OLTP) and online analytical processing (OLAP) applications, offering options for both pessimistic and optimistic concurrency control. PostgreSQL also included features like partitioning and parallel execution to efficiently manage large data volumes. Its scalability, enhanced by effective replication capabilities, made it an appealing choice for creating highly available and efficient data management solutions.

The iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan benefited greatly from integrating PostgreSQL as its backend database due to its robust features and reliability. PostgreSQL's adherence to ACID compliance ensured data integrity, preventing issues like double bookings, which was crucial for managing community resources effectively. Its advanced security measures protected sensitive user data while providing customizable features that allowed developers to tailor the system to specific needs, such as diverse data types and flexible structures for various facilities and equipment. Additionally, PostgreSQL's powerful indexing capabilities enhanced query performance, ensuring quick access to reservation

information, and its scalability allowed the system to accommodate growth within the barangay community. Furthermore, the database's support for internationalization ensured inclusivity for a diverse user base.

Simple Mail Transfer Protocol (SMTP)

According to Harshwardhan and Divekar (2019), since 1980, the Simple Mail Transfer Protocol has been utilized for sending and receiving emails. The primary function of this protocol is to transmit plain text messages without initially taking into account any security aspects. Furthermore, SMTP is part of the Application Layer of the TCP/IP protocol. SMTP dictates the movement of emails from one MTA to another, potentially traversing various mail servers before reaching the recipient's server. SMTP employs a "Store and Forward" mechanism for transmitting emails, offering codes to streamline server-to-server email communication. It segments emails for easier transmission and commonly operates on ports 25 (for standard mail transfer), 465 (though not compliant with RFC), and 587 (for TLS encryption). The domain address for Gmail's SMTP service is 'smtp.gmail.com' on port 587, while the domain for Yahoo's SMTP service is 'smtp.mail.yahoo.com' on port 465 (Jagadeeshwaran et al., 2023).

Based on the article entitled "What is SMTP (Simple Mail Transfer Protocol)" by Gillis (2023), SMTP uses a client-server model; an email server utilizes SMTP to transmit a message from an email client to another email server. The email server then acts as a relay service by using SMTP to send the email to the intended receiving email server. Finally,

the receiving email server retrieves incoming mail via an email client like IMAP, placing it in the recipient's inbox.

The current study will use SMTP to enhance user communication, improve user experience, boost engagement and satisfaction, and analyze communication strategies within the system. Key functions include sending reservation confirmations with details, providing timely booking updates, securely facilitating password resets and account notifications, and enabling feedback and communication through forms and surveys post-usage. This utilization of SMTP ensures efficient and effective communication with users, improving the overall functionality and user satisfaction of the iSKed system.

Node.Js

According to Jadhav and Gonsalves (2020), in Role of Node.js in Modern Web Application Development, the paper described Node.js as a JavaScript runtime environment built on Chrome's V8 JavaScript engine. It was a cross-platform runtime environment originally developed in 2009 by Ryan Dahl for creating server-side applications. Using Node.js, developers could easily build scalable, fast, and lightweight applications. V8 and Node were mainly written in C and C++, focusing on low memory consumption and performance. Node.js was considered server-side JavaScript and addressed performance issues related to network communication time by dedicating excessive time to processing web requests and responses. The paper identified why major companies like PayPal, LinkedIn, Netflix, and Yahoo used Node.js. Essentially, any

business using Node.js could utilize fewer servers, fewer engineers, and reduced page load times.

Expanding on this, Tiago Brito et al. (2023), in a Study of JavaScript Static Analysis Tools for Vulnerability Detection in Node.js Packages, mentioned that Node.js was a cross-platform, backend runtime environment that executed JavaScript code. It functioned as a web container, handling HTTP requests. Node.js also had an ecosystem of third-party packages managed by the Node Package Manager (npm). At that time, npm contained thousands of packages that developers could readily import for web application development or other packages. The study highlighted Node.js's role as a backend runtime environment that executed JavaScript code.

Basumatary and Agnihotri (2022) examined the advantages and limitations of using Node.js, highlighting that it enabled developers to create server-side applications using JavaScript. This approach simplified the development workflow and allowed for smoother coordination between client- and server-side processes. Its reliance on an event-driven, non-blocking input/output model, combined with asynchronous processing, significantly enhanced system performance. Their study emphasized that Node.js became an essential tool in web development, helping developers transition into full-stack roles without needing to master different programming languages for frontend and backend development. By utilizing a single-threaded model, Node.js reduced the strain on CPU memory typically required for thread management, thereby improving the efficiency of client-side multitasking. Furthermore, the study pointed out that Node.js offered strong scalability and enjoyed substantial community support, with contributions from leading

companies like Google, Facebook, Netflix, and Amazon driving its continuous improvement.

Express JS

Express.js served as the primary web framework for Node.js, specifically designed to build web applications and APIs. It was widely recognized as the standard server framework for Node.js applications. Developing a backend from scratch for a Node.js project was often demanding and time-intensive, so developers leveraged frameworks like Express.js to streamline their processes and concentrate on higher-priority tasks (Codecademy, 2024).

Express.js provided core features that enabled developers to define handlers for various HTTP methods on different URL paths (routes), integrate with view engines for dynamic response generation by embedding data into templates, and configure key aspects of a web application such as server ports and template directories. Furthermore, Express.js supported the use of middleware functions at any stage of request handling, allowing for enhanced customization and functionality. While Express.js maintained a minimalist core, the surrounding community contributed numerous middleware packages addressing a wide range of web development needs, including cookie management, session handling, user authentication, URL parameter processing, POST data handling, and security enhancements (MDN, 2024).

By simplifying the management of routes, requests, and responses, Express.js made it easier to develop scalable and reliable applications. Its rich ecosystem of middleware

appealed to both novice and experienced developers, contributing significantly to its widespread adoption (GeeksforGeeks, 2024).

Moreover, Express.js provided essential features that greatly supported the development of the system in this study. It enabled the definition of request handlers across different HTTP methods and URL paths, seamless integration with view rendering engines for dynamic content delivery, and straightforward configuration of the system's operational settings. These features were critical for building a responsive, web-based platform capable of efficiently managing user interactions and processing data. The ability to incorporate middleware at various points in the request cycle further enhanced the system's reservation and management functionalities.

Postman

Postman software allowed developers to test, document, and share Application Programming Interfaces (APIs). Developers commonly used it to streamline the API testing process, offering an intuitive interface for sending requests, reviewing responses, and troubleshooting problems (Demir, 2024). Postman became an essential tool in API development and testing, widely recognized for its comprehensive feature set that facilitated the creation, testing, and documentation of APIs. As software architecture increasingly relied on APIs, tools like Postman became vital in managing their complexities effectively. The effectiveness of the iSKed system depended on the interconnections among components like user bookings, availability checks, and management tasks, making robust API management crucial. Postman's extensive features

empowered developers to create clear and efficient APIs for these processes. As software increasingly relied on APIs, tools like Postman were essential in navigating complexities. By employing Postman, the development team ensured seamless integration of each functionality in the reservation management system, thus enhancing the user experience for the Barangay Western Bicutan community. Ultimately, Postman improved the development process, facilitated better documentation, and fostered collaboration among team members, ensuring the final system effectively met community needs.

Based on the study entitled API Testing Using Postman Tool by Kore et al. (2022), Postman was an API client designed for developers to simplify creating, sharing, testing, and documenting APIs. It allowed users to build and save simple and complex HTTP/s requests while viewing their responses, making the workflow more efficient and less tedious. Additionally, Postman was a leading API automation and documentation tool initially launched as a Chrome plugin and eventually served over 5 million developers and more than 100,000 enterprises worldwide. Valued at $2 billion, Postman established itself as a key platform for developing enterprise APIs. As the API economy expanded, developers faced new challenges, as traditional methods of manually creating and testing APIs were no longer sufficient for applications that interacted with numerous APIs. Development, testing, and delivery teams had to collaborate effectively to ensure seamless application functionality and maintain a competitive edge.

Postman software was essential for the development of this study. It streamlined the creation and testing of APIs necessary for the reservation management system, allowing developers to establish endpoints for functionalities like facility booking and user account

management. The tool also provided automated testing features to verify the reliability of the APIs and included documentation that detailed the functions of the endpoints, expected inputs, and usage examples to enhance stakeholder understanding. In addition, Postman fostered collaboration among team members by enabling multiple developers to share API collections and work together effectively. It also allowed for the simulation of different user interactions with the system, helping to assess its performance across various scenarios. Using Postman made the development process more efficient and organized, significantly aiding in the successful implementation of the system to serve the community's needs.

Bootstrap

Bootstrap was a comprehensive set of valuable, reusable code fragments written in HTML, CSS, and JavaScri that enabled developers and designers to effectively build fully responsive websites. In essence, Bootstrap minimized the required CSS coding, allowing users to focus more on designing their web pages (Ouellette, 2023).

It also offered design templates based on HTML and CSS for various elements such as typography, forms, buttons, tables, navigation, modals, image carousels, and more. Additionally, it supported JavaScript plugins, enabling users to easily develop responsive designs (Gaikwad, 2019).

Bootstrap brought responsive web design to life by allowing web pages or applications to recognize a visitor's screen size and orientation, automatically adjusting the layout to suit. The mobile-first strategy considered smartphones, tablets, and specialized mobile

applications as the primary tools for employees to complete their tasks. Bootstrap met the design needs of these technologies by incorporating UI components, layouts, JavaScript tools, and a framework for implementation. The software could be accessed either as precompiled files or as source code (Zola, 2022).

Bootstrap was a powerful front-end framework that facilitated the creation of responsive web applications, making it ideal for the SK Youth Program's reservation and management system. It streamlined development, allowing developers to prioritize design while ensuring optimal functionality on mobile devices, which were widely used by the target youth demographic. With pre-built templates and JavaScript plugins, Bootstrap enhanced user engagement for tasks like event registration. Its open-source nature encouraged collaboration within the local community, although developers had to manage customization challenges and ensure local staff were trained for system maintenance. Overall, Bootstrap provided an opportunity to build an accessible and engaging platform for the youth program.

Heroku

Heroku functioned as a Platform-as-a-Service (PaaS), offering a dependable environment for developing, launching, and overseeing applications. According to Nuprene (2020), it was a cloud-based solution known for its accessibility, stability, and user-friendly nature. The platform enabled applications to run in isolated environments while managing critical operational tasks such as container orchestration, load balancing, logging, configuration handling, failover support, and security. Furthermore, it empowered

developers to build, deploy, execute, and scale applications across various programming languages.

Heroku integrated seamlessly with Git, allowing developers to deploy applications with a single command by pushing code to the remote Heroku repository. Other deployment methods included GitHub integration, Dropbox Sync, and the Heroku API, which offered flexibility in building and releasing applications. According to Bajaj et al. (2020), Heroku packaged an application's source code with its dependencies, such as packages, modules, and libraries needed for the runtime environment. It compiled the source code, identified the necessary dependencies, and organized them into a slug structure, which included all resources required to run the application.

Heroku's reliability and user-friendly features were essential for the iSKed system, which needed a dependable platform for managing facility and equipment reservations. Built on dynos, Heroku provided isolated environments that enhanced stability and security, allowing efficient handling of multiple user requests. Additionally, Heroku's packaging of the application's code and dependencies into a slug facilitated seamless deployment and scalability, enabling the iSKed system to adapt to increasing demand in Barangay Western Bicutan. Its integration with Git allowed for rapid updates and new features, ensuring responsiveness to user needs, while various deployment methods supported continuous improvement and innovation.

Render

Render is a well-known cloud-based platform that builds and deploys code and offers resources for hosting applications and services. Various software developers, including frontend, backend, and full-stack developers, use Render to deploy their code. For example, one developer may host their web API on Render, while another might deploy an application that relies on that API (Codecademy, 2025). Render natively supports Node.js/Bun, Python, Ruby, Go, Rust, and Elixir (Render, 2024).

Render can significantly assist the study by making application deployment easier, enabling developers to concentrate on creating features instead of handling infrastructure management. It offers scalable hosting options to accommodate growing needs, supports various programming languages and frameworks, and automatically manages networking tasks like TLS certificate handling, enhancing security. Render is also cost-effective, with free tiers that are beneficial for budget-conscious projects, and it allows for custom domains, improving user accessibility and branding. Additionally, its robust documentation and community support facilitate troubleshooting and best practices during development. Overall, utilizing Render can streamline the development process and lead to a more effective reservation management system.

Visual Studio Code (VS Code)

Visual Studio Code (VS Code) was a free and lightweight source code editor available on different operating systems, such as Windows, macOS, Linux, and Raspberry Pi OS. It had built-in support for JavaScript, TypeScript, and Node.js, and VS Code offered

extensions for other languages, including C++, C#, Java, Python, PHP, and Go. VS Code provided IntelliSense, which enhanced the coding experience of the user (Heller, 2022).

VS Code was packed with features designed to meet the needs of its users. It came with built-in support for web applications, allowing developers to work seamlessly on web projects. Furthermore, VS Code enabled users to open multiple files and folders simultaneously, streamlining the workflow for managing complex projects. Additionally, it had built-in Git integration, allowing users to effortlessly pull from and push to GitHub repositories within the editor (Pedamkar, 2023).

Visual Studio Code allowed users to customize its default configurations according to their preferences. It included modifying the theme to change the visual appearance of the editor, adjusting settings such as the font size for improved readability, and customizing keybindings to set up personalized keyboard shortcuts. These customization options empowered users to tailor their coding environment to suit their needs and workflow preferences (Kramer, 2024).

The current study used Visual Studio Code as the code editor to build web applications since it had built-in support for web development. It also offered features that made programming easier for researchers. Lastly, its extensive library of extensions allowed for enhanced customization and functionality, making it an ideal choice for this project.

Version Control System

A Version Control System (VCS) was fundamental in managing and organizing source code, promoting efficient development practices and maintaining the quality of project

outputs (Deepa et al., 2020). Git, a widely used open-source VCS, played an important role in software development by providing lightweight yet powerful capabilities, especially suited for handling large-scale and complex projects (Ponuthorai & Loeliger, 2023). Research on development practices showed that approximately 93% of professional developers relied on Git (Stack Overflow, 2021), emphasizing its widespread adoption alongside the challenges associated with its use. Although GitHub gained significant popularity within academic environments, some concerns were noted, such as the complexity of mastering version control systems and the additional teaching load placed on educators (Yu-Cheng-Tu et al., 2022). In addition, Git became increasingly integrated into educational settings, particularly in statistics and data science courses, where faculty members who fully adopted Git encouraged students to use Git and GitHub together with RStudio for project work (Beckman et al., 2020).

Figma

Figma was a vector graphics editor and prototyping tool that offered a significant advantage as it was accessible online and a downloadable application. With Figma, users could create designs such as user interfaces for websites, vector graphics, and more. It was highly adaptable and user-friendly, allowing real-time collaboration (Sharma & Tiwari, 2021). It was a versatile and constantly evolving design tool that was open to all users, offering a free plan for beginners. It was also available as web and desktop applications, each with advantages (Staiano, 2022).

According to the study by Olujimi et al. (2022), titled "User-Centered/User Experience (UC/UX) Design Thinking Approach for Designing a University Information Management System," Figma offered vector-based tools for illustration, prototyping, and software development hand-off. It streamlined the creation of uniform interfaces by supporting reusable components, design systems, and style guides. Additionally, designers had access to a growing library of templates, themes, plugins, and UI kits, enhancing their workflow efficiency.

The survey in the study "Exploring the Impact of Linear & Non-Linear Presentation Methods in a Design History Course" by Howell et al. (2023) examined the effectiveness of linear and non-linear presentation methods using Canva and Figma in a Design History course with 18 design students from various disciplines. Students presented multiple times using both tools and were encouraged to explore different presentation styles. When asked which presentation tool to use in next year's course, 50% chose Figma, 11.2% chose Canva, and 38.9% chose both. The results showed a preference for Figma among students for future courses, suggesting that non-linear tools like Figma enhanced learning experiences compared to traditional linear presentation tools like Canva or PowerPoint. Figma had a steeper learning curve than Canva, but its non-linear approach effectively engaged audiences, improved knowledge retention, and enhanced team presentations.

The study aimed to use Figma to enhance design and user experience. Figma was utilized to create a comprehensive UI interface that would guide the website design of the iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan. Its online accessibility and downloadable application

facilitated collaboration and design creation. With vector tools, prototype features, and software development hand-off support, Figma helped create visually appealing and functional systems. Real-time collaboration capabilities aligned with the benefits of team presentations and partner involvement for detailed discussions. By using Figma for consistent interfaces through reusable components, design systems, and style guides, iSKed users enjoyed a seamless experience. Incorporating non-linear design elements based on Figma's capabilities engaged users and improved usability, drawing from studies on enhancing learning experiences.

DBeaver

DBeaver was an open-source database management tool designed for developers and administrators. It supported multiple database management systems (DBMSs) like MySQL, PostgreSQL, MariaDB, SQLite, Oracle, and SQL Server, among others. It is compatible with JDBC and ODBC drivers and can also connect to non-traditional databases like MongoDB and Redis. DBeaver runs on Windows, Linux, macOS, and Solaris as a cross-platform tool, making it versatile for different users. The tool provided essential features to streamline database management. The Database Navigator allowed users to explore database structures easily, while the SQL Editor supported syntax highlighting and autocomplete for efficient query writing. Users could modify records directly within tables, similar to spreadsheet editing. DBeaver also offered a visual representation of database schemas and objects and export options. Additional functionalities included support for stored procedures in SQL Server and XML datatypes

in Oracle and DB2. Moreover, it allowed interface customization, offering theme adjustments for a personalized experience. The study utilized DBeaver as the primary database management tool due to its extensive compatibility with various DBMSs. Its intuitive interface and built-in SQL editor facilitated efficient data handling and query execution. Additionally, its visualization and export features allowed for better organization and analysis of database structures, making it a suitable choice for this research.

Cloudinary

Cloudinary serves as a temporary web-based solution for efficiently handling product images and documents. It streamlines the organization of products by supporting multiple images and files. Currently, the system stores roughly 175,000 images and documents associated with nearly 60,000 products. Built using S3, API Gateway, and Lambda, alongside HTML, JavaScript, and CSS, this internally developed tool will eventually be decommissioned once the new Product Information Management (PIM) system takes over its role in managing product-related assets (Niemi, 2020).

It provides comprehensive multimedia solutions, eliminating the need for complex and costly graphics management tools. Users can confidently upload and securely store unlimited images, benefiting from features like historical revisions and automatic backups. Cloudinary also allows various image modifications, including effects, watermarks, resizing, face detection, and cropping. Images are optimized for any device and delivered quickly via Akamai's global CDN. Additionally, its RESTful APIs and SDKs simplify

image management, while advanced analytics help users assess and enhance image performance (Medical Guide System, 2022).

The iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan parallels the Cloudinary tool in its focus on efficient resource management just as Cloudinary streamlines the handling of product images and documents, iSKed aims to enhance the organization of facility and equipment reservations.

Incorporating principles from Cloudinary, such as user-friendly interfaces, secure data storage, and features like historical revisions, can improve iSKed's functionality and user experience. Additionally, leveraging multimedia capabilities similar to Cloudinary could allow users to upload images of facilities, aiding in informed decision-making.

ISO/IEC 25010

ISO 25010, also known as "Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models," a standard for evaluating software quality, defines various characteristics and sub-characteristics that help assess both the product's quality and usability (Britton, 2021). It outlines two key models: the "quality in use model" and the "product quality model." The former evaluates the user's experience and the outcomes when interacting with a product in specific contexts, while the latter focuses on assessing both the static aspects of the software, such as its code structure, and the dynamic aspects, such as performance. These models work together to provide a detailed, structured approach to evaluating software

quality. The product quality model, specifically, consists of eight primary characteristics and 31 sub-characteristics:

- Functional Suitability: Measured how well a product or system provided the needed functions.

- Functional Completeness: Ensured all specified tasks and user goals were covered by the functions.

- Functional Correctness: Ensured the system provided accurate results with the required precision.

- Functional Appropriateness: Ensured the functions effectively achieved the specified tasks and goals.

1. Reliability: Measured how consistently a system, product, or component performed its functions under certain conditions.

   - Maturity: Indicated how reliably the system met user needs.

   - Availability: Indicated if the system was up and running when needed.

   - Fault Tolerance: Indicated how well the system continued to work despite hardware or software issues.

   - Recoverability: Indicated how well the system could recover data after an interruption or failure.

2. Performance Efficiency: Measured how well the system performed in relation to the resources used.

   - Time Behavior: Measured response times, processing times, and throughput rates while the system was functioning.

- Resource Utilization: Measured the types and amounts of resources the system used while performing.

- Capacity: Measured the maximum limits the system could handle.

3. Usability: Measured how effectively, efficiently, and satisfactorily a product or system could be used to achieve goals.

  - Appropriateness Recognizability: Measured how well users could identify if the product fits their needs.

  - Learnability: Measured how easy it was to learn to use the product.

  - Operability: Measured how easy the product was to operate and control.

  - User Error Protection: Measured how well the system prevented user errors.

  - User Interface Aesthetics: Measured how pleasing the user interface looked.

  - Accessibility: Measured how well the product could be used by people with diverse abilities.

4. Security: Measured how well a product or system protected data and information from security threats.

  - Confidentiality: Ensured data was accessible only to authorized users.

  - Integrity: Ensured data and programs were protected from unauthorized changes.

  - Non-repudiation: Ensured actions or events could be proven to have occurred.

- Accountability: Ensured actions of unauthorized users could be traced.

- Authenticity: Ensured the identity of users and resources could be verified.

5. Compatibility: Measured how well a product, system, or component exchanged information and worked in a shared environment.

    - Co-existence: Ensured the product worked efficiently in a shared environment without affecting others.

    - Interoperability: Ensured multiple systems or components could exchange and use information effectively.

6. Maintainability: Measured how easily a product or system could be modified to improve, fix, or adapt to changes.

    - Modularity: Ensured parts of the system could be changed with minimal impact on others.

    - Reusability: Ensured components could be used in multiple systems.

    - Analyzability: Ensured changes could be effectively assessed and issues diagnosed.

    - Modifiability: Ensured changes could be made without causing new problems or reducing quality.

    - Testability: Ensured the system could be effectively tested to meet criteria.

7. Portability: Measured how well a system, product, or component could be moved to different environments.

    - Adaptability: Ensured the product could adapt to different hardware, software, or usage environments.

- Installability: Ensured the product could be easily installed and uninstalled.

- Replaceability: Ensured the product could effectively replace a similar product.

Britton (2021) stated that software quality was important as it reflected how well a software program functioned and whether it met necessary requirements such as security and maintainability. By assessing software quality, various critical aspects were evaluated:

- Testability: How easy it was to test the software to ensure it worked correctly.

- Understandability: Whether the software was easy to understand.

- Modifiability: Can changes or improvements could be made to the software without causing issues?

Unicsoft (2023) highlighted that poor software qualities were often obvious, while good qualities might not be immediately apparent to end users. Despite this, extra time and effort were required to meet ISO/IEC 25010 standards. Adhering to the ISO/IEC 25010 product quality framework ensured that a software product met user expectations. When developed in accordance with these standards, the software typically delivered four main outcomes:

- An effective and efficient product: High-quality software allows users to complete their tasks quickly and easily.

- User peace of mind: A secure and reliable product ensures users can use the software without fear or hesitation.

- High user satisfaction: A well-designed product provides a positive experience, leading to high levels of user satisfaction.

- A product that did what it was supposed to, period: The product performed exactly as expected, meeting users' needs and requirements.

Effective software quality management requires assessing products against objective criteria. The ISO/IEC 25010 standard offered a structured framework for evaluating software quality, guiding developers to create products with high functionality and security.

Synthesis of Review of Related Literature

The existing literature underscores the inefficiencies of manual reservation systems, such as those used in Western Bicutan's Sangguniang Kabataan (SK). Research by Omnify (2020) and Aldisa (2023) emphasizes how automated reservation platforms streamline processes, minimize scheduling conflicts, and improve user satisfaction. For example, Aldisa's study on hotel reservation systems illustrates the value of simplified workflows and real-time availability, which are directly relevant to iSKed's development. Similarly, Castillo et al. (2024) and Palangdao et al. (2023) highlight the importance of youth profiling to personalize services and boost engagement, a critical consideration for SK programs.

Technological advancements play a vital role in modernizing reservation systems. The utilization of JavaScript, as highlighted by Dou (2020) and Akshay et al. (2019), facilitates dynamic and interactive functionalities, such as real-time data updates and responsive user interactions. ReactJS, discussed by Sousa et al. (2020) and Rawat (2020), enhances the user interface by creating modular, responsive components that optimize the reservation

experience across various devices. Complementing these tools, CSS ensures visually cohesive and user-friendly designs, as noted by Kuparinen (2023). These technologies collectively enable iSKed to provide a seamless and engaging user experience.

Design and backend infrastructure are equally significant. According to Sharma and Tiwari (2021), Figma offers a collaborative platform for developing intuitive UI prototypes that align with the project's user-centric goals. On the backend, PostgreSQL, as detailed by Juba and Volkov (2019), ensures data reliability and scalability, which is critical for securely managing large volumes of reservation data. SMTP protocols enhance communication by enabling automated notifications, confirmations, and feedback mechanisms, as explained by Harshwardhan and Divekar (2019). Deployment platforms such as Heroku and Render further simplify the hosting and scaling of applications, providing robust cloud-based solutions that ensure system reliability and accessibility. These tools create a cohesive infrastructure to support iSKed's functionalities.

To conclude, the literature highlights the broader impacts of adopting reservation systems. Studies such as those by Permatasari et al. (2023) and Marinova (2023) demonstrate how analytics-driven platforms optimize resource allocation and improve decision-making. Responsive design principles, emphasized by Fischer and Schofield (2021), ensure accessibility across devices, enhancing inclusivity. iSKed aligns with global efficiency, usability, and security standards by integrating GitHub for version control. Moreover, the system's adherence to the ISO/IEC 25010 quality framework ensures compliance with software quality standards. By addressing characteristics such as functionality, reliability, usability, and security and leveraging deployment platforms like

Heroku and Render for enhanced performance and scalability, iSKed is designed to meet user expectations, provide a high-quality user experience, and maintain long-term system performance.

Review of Related Studies

The rapid technological advancements significantly enhanced the efficiency and accessibility of systems designed to aid local government entities, particularly barangays in the Philippines (Ballaran et al., 2023). These barangays were crucial as they served as the foundational units for planning and implementing community projects, embodying the first level of governance where development initiatives took root. However, it was paradoxical that, despite their pivotal role, barangays often lacked comprehensive information for effective planning and policy execution (Carpio, 2020). This gap hindered their ability to develop and deploy effective programs that met the community's needs.

Recent research titled "A Risk Assessment of a Barangay Management System Portal: A Case Study" highlighted the situation of the over 42,000 barangays in the Philippines. As the smallest political units, they were indispensable for executing various development programs, projects, and policies. While a few barangays had started embracing technology by using computers and electronic devices for official transactions, many still relied predominantly on manual processes. This reliance on tradition created inefficiencies and limited their ability to harness technological benefits fully.

The contrast between the potential of technology and the reality of barangay operations underscored the urgent need for training and resources to assist these local units in transitioning to more modern, efficient management systems that could better serve their communities. By addressing these challenges, barangays could play a more effective role in local governance and community development, ultimately leading to improved resident outcomes.

In the research study "Information Management System with Project Monitoring for Barangay" authored by Balilia et al. (2023), the Barangay Information Management System (BIMS) was proposed as a Windows-based software designed to automate traditional record-keeping processes within barangays. It aimed to securely store barangay records, files, and session documentation while enabling the efficient processing of pre-filled document requests for legal certificates. BIMS also included a Project Module for tracking barangay projects, featuring a dashboard that displayed a comprehensive list of projects, their statuses, budget information, and monthly project counts. This system was intended to assist barangay officials in meeting client needs by improving the organization and retrieval of records and documentation. As countries globally increasingly focused on e-governance to enhance public administration and service quality, BIMS was particularly relevant for barangay offices like Barangay Ugac Sur. It aimed to improve operational efficiency by managing and storing extensive records, facilitating document requests, and enhancing transparency regarding barangay projects among verified employees.

Senaris (2023), in the study *"Implementation of Barangay Management System: An Extension Service of CvSU-Tanza Campus,"* examined how a barangay management

system improved service delivery for both officials and residents. The system streamlined transactions such as posting announcements, handling citizen complaints, and issuing certifications. It was developed using web technologies, including HTML, CSS, JavaScript, and PHP.

In the study by Himo et al. (2022), the development of a Community Request Queue Management System for local government units (LGUs) employed a structured approach through the software development life cycle (SDLC) to create an efficient tool for managing citizen interactions. Utilizing technologies like HTML for the user interface, Java for backend processes, PHP for web development, and MySQL for database management, this system enabled users to submit inquiries, file complaints, and request documents through a streamlined online platform. Key features included a user-friendly interface that simplified navigation for citizens, an administrative module for government staff to monitor and process requests, and functionalities that enhanced communication between residents and officials. The system's effectiveness was evaluated based on six criteria: functionality, usability, efficiency, portability, dependability, and security. It had to effectively perform its intended tasks, be easy to use, process requests promptly, be accessible on various devices, operate reliably without frequent downtimes, and protect sensitive user data.

The barangay management system improved service delivery for officials and citizens but did not adequately address youth engagement through the Sangguniang Kabataan (SK). While it facilitated transactions like announcements and complaints, it lacked features that prioritized youth involvement in governance. The reliance on manual record-keeping

within the Western Bicutan SK revealed inefficiencies that automated systems could resolve, yet there was limited research on their impact on youth participation. Additionally, a comprehensive framework was needed to integrate management systems with youth-oriented initiatives, overcoming access barriers and fostering youth ownership.

In line with the development of iSKed, the system conducted by Barzaga et al. (2020), the eReserba, addressed the inefficiencies of the current manual reservation system by leveraging real-time data for an immediate reassessment of processes. The study also featured a reviewable interface that enhanced usability for users while eliminating common issues associated with manual documentation. Though this study focused on room and equipment reservations, the system did not integrate a dynamic service creation module that allowed for adding and updating various equipment. Moreover, the study primarily addressed room reservations and lacked an in-depth analysis of user patterns. By addressing these potential gaps, the research could significantly inform and enhance the current study, iSKed.

By recognizing the limitations of the aforementioned related studies, the proponents could prioritize the development of features that enhanced usability, such as dynamic service integration, ensuring the system better met user needs.

Conceptual Model of the Study

Figure 1.

Conceptual Model of the Study

INPUT	PROCESS	OUTPUT
KNOWLEDGE REQUIREMENTS a. Knowledge in Reservation System b. Web Development c. Knowledge for Heroku and Render d. Database Management e. Calendar and Scheduling Algorithm f. Report Generation and Analytics g. ISO/IEC 25010 **SOFTWARE QUIREMENTS** a. Visual Studio Code b. Github c. Figma d. PostgreSQL e. JavaScript, ExpressJS, ReactJs, CSS , Bootstrap, NodeJs **HARDWARE REQUIREMENTS** a. Specification: 1 TB hard disk drive, 8 GB RAM b. Processor: Minimum speed 2.30 GHz, quad-core processor c. Operating system: 64-bit architecture	**System Design** **Development: Agile – Scrum Methodology** • Sprint Planning • Daily Scrum • Sprint Review • Sprint Retrospective **Operating and Testing Procedure**	**iSKed: DEVELOPMENT OF FACILITIES AND EQUIPMENT RESERVATION MANAGEMENT SYSTEM FOR BARANGAY WESTERN BICUTAN**

EVALUATION: ISO25010

The study employed a visual presentation, an Input-Process-Output (IPO) diagram, presented in Figure 1, to depict the study's central concept. This diagram illustrates the overall flow of the study.

Input

The first essential requirement in developing the system was the knowledge requirement, which provided the foundation for the researchers' ability to create the web application. This includes the knowledge of the developer in processing, report generation, and knowledge and skills in development, all of which were critical to ensuring the process of the project.

The second requirement was the software requirements, which helped the proponents understand the concept of a software requirement, as it served as an indicator of how the developers created the system. These were the applications and extensions that were essential in creating and running the project. This included the Windows 10 operating system, PostgreSQL, JavaScript, Node.js, Express JS, ReactJS, CSS, and Bootstrap.

Finally, the hardware requirements were the tools used in developing the system, which included the electronic devices necessary in project development, which were vital in accessing the system. This also consists of the computer desktop and laptop that have a minimum of 1 TB of hard disk drive, 8 gigabytes of RAM or random-access memory, a minimum processor of 2.30 GHz, and a 64-bit operating system. These specifications ensured the stability and performance of the development environment.

Process

The process was part of the development phase, and as the web-based system was in development, the recommended approach for the software development model applied in the study was the agile scrum methodology. It included tasks such as design, sprint planning, daily scrum, sprint review, and sprint retrospectives.

System Design. In this phase, the system design was the overall system architecture, such as the database structure and the systems' diagrams. Wireframes and mockups were also developed for the user interface to ensure usability. Technical requirements for front-end and back-end development were also defined.

Sprint Planning. In this phase of the agile scrum approach, the researchers were tasked to set daily meetings and plans and present necessary information and progress in terms of indicating the needs and required adaptation and application for the project.

Daily Scrum. This was the process of developing the project each day and tracking the versions and modifications of the project.

Sprint Review. During this phase, the project underwent testing as it was necessary to check and review the actual output if it was to the client's liking and demand.

Sprint Retrospective. It was a phase of repeating the same cycle over time to furnish and finalize the project.

Operating and Testing Procedure. The system was initially operated to ensure functionality, followed by various testing phases, including functionality, usability, performance, security, compatibility, and user acceptance testing (UAT) to verify that all

features met user needs performance standards, with necessary refinements and bug fixes applied before the final deployment.

Output

The present study proposed the development of iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan in Taguig City.

Evaluation

The evaluation included the need to meet the demands or required output of the clients. The ISO 25010 was used to evaluate the systems' applicability, effectiveness, compatibility, usability, dependability, security, maintenance, and portability.

Operational Definition of Terms

The following terminology was defined to ensure a clear understanding of this study:

Add. Refers to the process of adding data in different sub-concepts, such as the schedule for reservation and equipment.

Admin Dashboard. Refers to the page where the admin can track their key performance indicators.

Adult Youth. Refers to the youth in Western Bicutan aged 25 and above who were still involved in youth-focused programs and activities.

Appoint. Referred to the process of making a reservation.

Child Youth. Refers to individuals within the younger age group of the youth category, typically ages 15- to 17 years old.

Client. Refers to the admin.

Client login. Refers to the page on the website where the admin can log into their account.

Core Youth. Refers to the youth in Western Bicutan who were actively engaged in youth programs or activities, usually aged between 18- to 24-year-old.

Customer. Refers to the user and the people who made reservations and used the web-based reservation system.

Database. Refers to the collection of data wherein all the information and data were saved in the web-based reservation system.

Delete. Refers to the process of removing data in different sub-concepts, such as the schedule for reservation, customer profile, and equipment.

Equipment Reservation. Refers to the equipment reservation through SK Bicutan's borrowing program.

Generate Report. Refers to the process of creating an analysis report for all the processes and transactions dealt with by the admin and users, mainly for tracking finance.

Gym Reservation. Refers to the process of reserving the gymnasium for use as part of the SK Bicutan's reservation program.

Input. Refers to the process of inputting or adding data necessary from the users.

Inventory. Refers to the list or stock of available equipment for reservation in the SK Bicutan's borrowing program.

Login. Referred to the page and interface for logging into the account and checking profile credentials in the database.

Manage Websites. Refers to the process of managing and creating changes to the user's account in the reservation system.

Modify. The process of creating changes by the admin in different sub-concepts.

Personal Information. Refers to the information of the users and their credentials.

Program Description. Refers to the description and important details regarding the programs that were available for customers.

Profile. Refers to the credentials of the users or admin.

Register. Refers to the process of creating an account.

Report. An analysis report for all of the processes and transactions dealt with by the admin and users, mainly for tracking finance.

Reservation. Refers to the processed schedule of the customer.

SK Council Members. Refers to the members of the Sangguniang Kabataan (SK) in Western Bicutan.

SKWB Officials. Refers to the admin who controls the website and has the responsibility to configure, handle inquiries, and manage transactions.

System Developers. Refers to the proponents or researchers of this study who developed the system.

System Maintenance and Monitoring. The central hub for all work-related tasks.

Update. The process of creating updates and significant changes in different sub-concepts.

User Management. The process of managing the customer profile for significant purposes.

WB Youths. Referred to individuals who were residents of Western Bicutan above 16 years old and could access programs from the website iSKed.

Website Facility Management. The page where the subject of the facility received configuration and performed CRUD operations.

Website Information. The information is displayed to the customers, and the administrator modifies the information.

Website User. Refers to the youth who used the web-based reservation system.

Chapter 3

METHODOLOGY

This section detailed the study's research methodology, which included sections on project design, project development, operations, testing procedures, and evaluation procedures.

Project Design
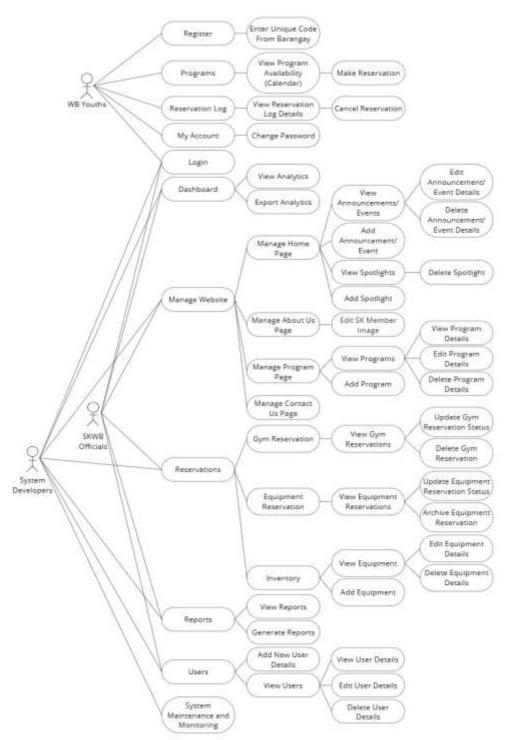
The project involved developing iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan, Taguig City, with the goals of organizing trends program management and streamlining gym reservations to meet overdemand, as well as simplifying the reservation process and improving user experience. The system sought to replace the current stages, which were carried out manually and inefficiently, with a web-based interface. It provided guidance on how to achieve the study's goals through a quick and successful reservation process for barangay residents, SK members, and barangay officials.

Use Case Diagram

The use case diagram in Figure 2 highlighted how various interactions worked within the website of iSKed of Western Bicutan's SK Youth. These interactions were performed by three main actors: WB Youths, SKWB Officials, and System Developers.

Figure 2

Use Case Diagram

The use case diagram in Figure 2 highlighted how various interactions worked within the iSKed website of Western Bicutan's SK Youth. These interactions were performed by three main actors: WB Youths, SKWB Officials, and System Developers.

WB Youths were the end-users of the iSKed website and had access to several key features within the system. These key features included Register, Login, Program Reservation, Reservation Log, and Account Management. First, in the "Register" module, youths activated their account on the website by inputting the unique code provided by the barangay and setting up their desired username and password. The "Program Reservation" module allowed the youths to make a reservation for a specific program. In the "Reservation Log" module, youths viewed all their reservations and canceled current reservations made through the website. The "My Account" module allowed them to change their password.

SKWB Officials had access to various functionalities, including Login, Data Analytics, Website Management, Reservations Management, Reports Management, and User Management. The "Data Analytics" allowed them to view and export into PDF format the information about the number of users, the number of gym and equipment reservations monthly, and yearly rating distribution. The "Manage Website" module enabled them to update details through its sub-modules, such as "Manage Home Page," "Manage About Us Page," "Manage Program Page," and "Manage Contact Us Page." The Manage Home and Program Pages allowed them to add, edit, or delete announcements, events, spotlights, and programs. The "Reservations" module allowed them to view reservations, set the reservation status, and set the status of borrowed equipment through its sub-modules, such

as "Gym Reservation," "Equipment Reservation," and "Inventory." In the "Reports" module, they viewed all reports and generated them in PDF format. The "User Management" module enabled them to view and manage user details, including deleting specific youth account details if necessary. Additionally, this module allowed them to add new user details to the website.
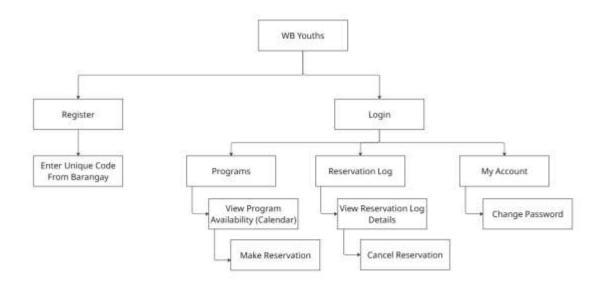
Lastly, System Developers accessed all of the SKWB Officials' modules to monitor website functionality. The System Developers' main role was to maintain the website by fixing bugs when they occurred and performing regular maintenance. This ensured that the website ran smoothly.

Module Hierarchy Chart

The module hierarchy chart for Western Bicutan's iSKed website represented the functionalities clearly and straightforwardly. Each box in the chart represented a module, and some of the modules were connected to their sub-modules. It illustrated the hierarchy and flow of activities within the system, providing a visual representation of how different components and features were organized and interconnected.

Figure 3.A

Hierarchy Chart for WB Youths Module:

Each box shown in the above figure represented a module, some of which were linked to their sub-modules. It showed the system's hierarchy and flow of activities, as well as the organization and interconnection of various components and characteristics. Once the current youth's information was recorded, they created a username and password. After logging in, youth could access further modules such as "Program Reservation," "Reservation Log," and "My Account.

Youths who made reservations for SKWB officials' specific programs using the "Program Reservation" module allowed them to look at available schedules on the calendar. The "Reservation Log" displayed the list of the history of any reservations made through the website by the youth. The third module, "My Account," allowed youths to change their account password.

This hierarchy chart illustrated the links and flow of the WB Youths module and its sub-modules, highlighting the options available to users.

Figure 3. B

Hierarchy Chart for SKWB Officials
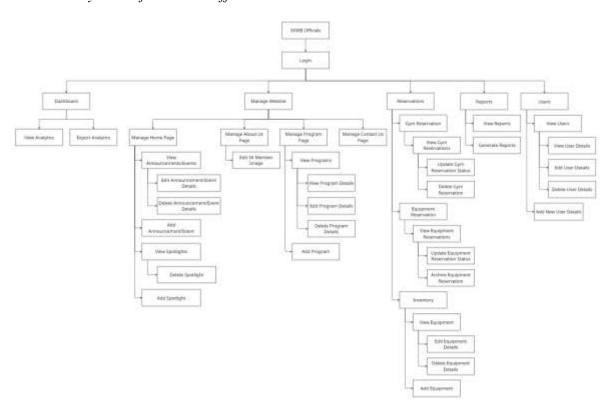
Figure 3. B presents the hierarchy diagram of the SKWB system, illustrating the connection between modules and their respective sub-modules, outlining the structure and flow of activities within the system.

In this system, SKWB officials are assigned to the "Login" module, which requires authentication before gaining access to any other modules. Once logged in, administrators are granted access to various modules, each handling distinct aspects of system management: "Dashboard," "Manage Website," "Reservations," "Reports," and "Users."
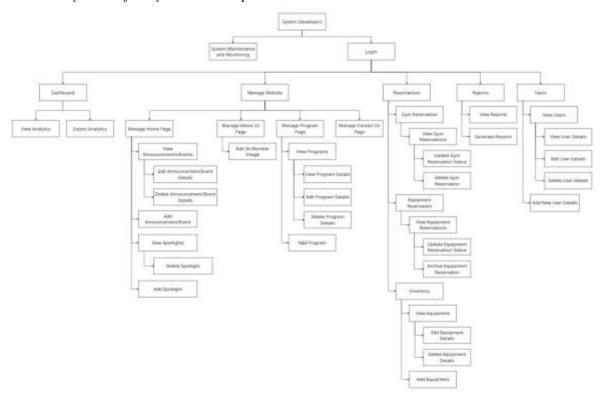
Upon logging in, the administrator is directed to the "Dashboard" module, where they can view data analytics and export the information in PDF format. Within the "Manage Website" module, four sub-modules are available for the administrator: "Manage Home Page," "Manage About Us Page," "Manage Program Page," and "Manage Contact Us Page." These sub-modules allow the admin to add, update, or delete content, depending on the specific page being managed.

The "Reservations" module includes three sub-modules: "Gym Reservation," "Equipment Reservation," and "Inventory." In "Gym Reservation" and "Equipment Reservation," the admin can view user reservations, approve or reject them, and track the status of borrowed items (whether they have been received, returned, or not returned). Within "Inventory," administrators can update the details of equipment and add new items, which are reflected in the "Equipment Reservation" module.

The "Reports" module enables the admin to view and generate various data reports, which can be exported as PDFs. In the "Users" module, administrators can manage user accounts by viewing, editing, adding, and archiving user information as necessary.

The hierarchy diagram provides a detailed visual representation of the system's structure, highlighting the relationships between modules and their sub-modules, and clarifying their respective roles and responsibilities within the system.

Figure 3.C

Hierarchy Chart for System Developers Modules

The diagram Figure 3. C illustrates that System Developers had access to the SKWB Officials modules. It also showed that "System Maintenance and Monitoring" was their responsibility for maintaining the website.
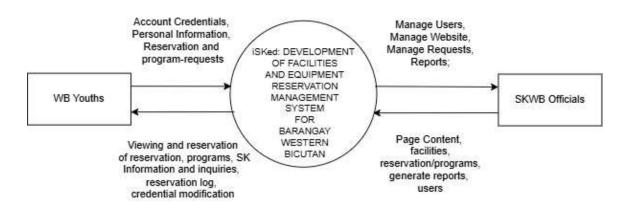
The System Developers' access to the SKWB Officials modules ensured that the website functioned properly. It also allowed them to identify errors that needed improvement. Monitoring and scheduling maintenance on the iSKed system ensured that the website operated smoothly and effectively.

Data Flow Diagram

The data flow diagram (DFD) for Western Bicutan's iSKed website visualized how information moved through the system. It represented numerous operations and the flow of data between them. Each level of the DFD described a different component of the system, ranging from high-level interactions with users to specific procedures that occurred behind the scenes. This diagram showed how data was processed, saved, and transported, providing a clear picture of the system's functionality and communication across its components.

Figure 4. A

Data Flow Diagram – Level 0 (Context Diagram)
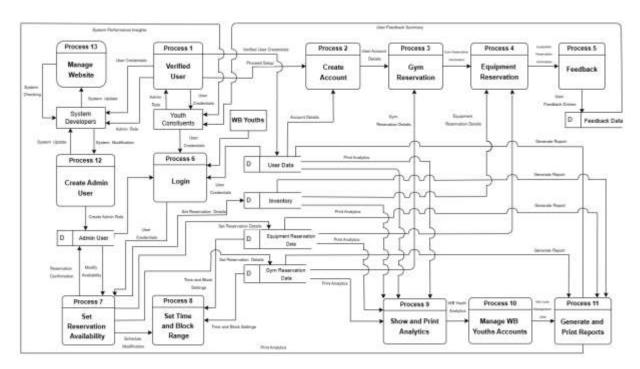
The figure above illustrates the Data Flow Diagram (DFD) Level 0, the Context Level Diagram. This figure demonstrated the data flow between external entities and the entire system, interacting with two key entities: the WB Youths, which referred to the users of the system, and the SKWB Officials, which referred to the admin. WB Youths accessed the system to view the programs offered by the SK Western Bicutan. Users can view and

inquire about programs at the SK Barangay. In the user entity, account credentials, personal information, reservation requests, and program inquiries were input into the system. The main outputs and accessible pages for users included browsing or viewing reservation logs, printing SK information and inquiries, and modifying credentials. The SKWB Officials oversaw the management of these reservations, ensuring smooth operations. In the SKWB Officials' entity, user, website, request, and report management were inputted into the system. The system output included page content, announcements, programs, reservations, reports, and user information.

Figure 4. B

Top Level Data Flow Diagram

The figure shows the Data Flow Diagram Level 1. While Level 0 displayed the data interaction between external entities and the system, Level 1 provided more details about the flow of data within the system. This level still showed the same external entities and data flows as Level 0 but with additional information on the system's subsystems and module processes.

The process was divided among three primary roles: WB Youths, SKWB Officials, and System Developers. The WB Youths access the system using account credentials and interact with the system to check SK Western Bicutan services, including programs that primarily contain gym and equipment reservations. Users will be able to book their reservations. On the other hand, WB Youths can browse for SK Western Bicutan contact

details, News and Events, relevant information, Reservation Logs, and Profile Modification. The SKWB Officials managed the entire functionality of the programs, such as posting programs, updating the home page, and managing contact and about us sections. The admin was also responsible for incoming requests from users, ensuring that the website remained functional for a new term of SK officials. The Users are able to fetch the summary forecast of the Users' information through the DataBase, and Create Reports if necessary. The System Developers ensured smooth operations through website modification, continuous system maintenance and monitoring, adding new features, updating existing features, removing features, and fixing bugs. Each entity interacted with the system by exchanging necessary data, facilitating a seamless reservation process, and ensuring efficient management of events and equipment.

Entity Relationship Diagram (ERD)

The Entity Relationship Diagram (ERD) for the iSKed website of Western Bicutan illustrated the relationships between the key entities within the system. A rectangle represented each entity, while the relationships between them were depicted as lines connecting the entities. The diagram highlights how different components were linked to each other, showing the flow of information and the dependencies among entities. This ERD provided a clear and structured view of the system's database design, ensuring that all data point and their relationships were well-defined and easily understood.

Figure 5

iSKed Entity Relationship Diagram
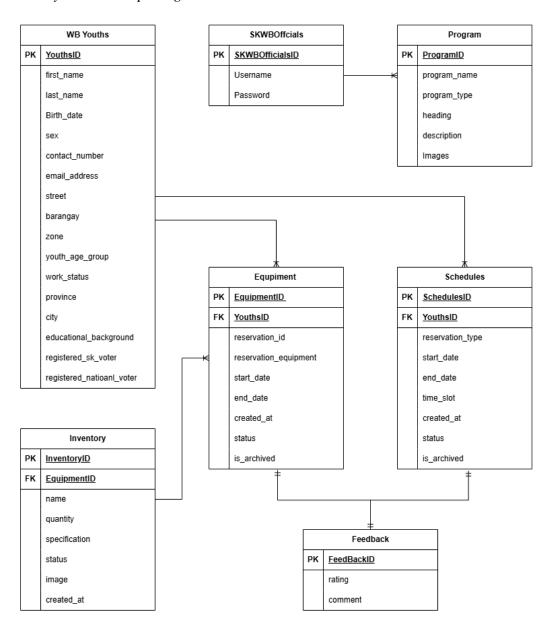
Figure 5 shows ERD for a system meant to manage programs, highlighting key entities and their relationships. The "SKWB Officials" entity was at the heart of the system and was in charge of program oversight and management.

The "Program" entity was linked to the "SKWB Officials" entity, indicating that SKWB Officials may construct several programs that help promote and manage. Additionally, a major program selection leads to the creation of a related entity.

The "Schedules" and "Equipment" entities represent the output of WB Youths' reservations. The "Schedules" entity specifically records all gym reservations within the Western Bicutan community, tracking when and where WB Youths make their bookings. Similarly, the "Equipment" entity manages reservations of equipment, ensuring availability and tracking its use. This entity is connected to the "Inventory" entity, which maintains a record of all equipment available for reservation.

The "Feedback" entity collects and stores feedback from participants regarding programs and equipment. It includes ratings and comments, which provide valuable insights for improving future programs and equipment management.

The "WB Youths" entity represents individuals within the Western Bicutan community. Each user had a unique profile and could make several reservations at different places. The system's design enabled the effective management of reservations, equipment, and associated WB Youth and SKWB Officials information, resulting in an excellent tool for managing iSKed.

System Flow Chart

Figure 6.A

WB Youths System Flow Chart
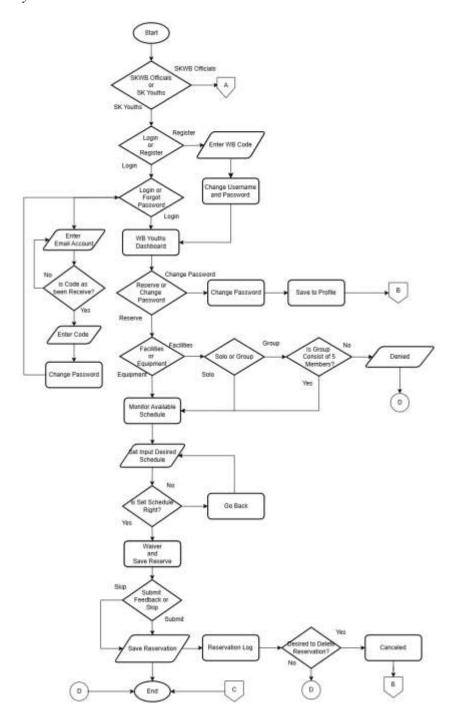
Figure 6. A presents the system flow chart for SK Youths. The process started with the user either logging in or registering. New users needed to obtain a code from the barangay hall or via email containing their identification details. Once logged in, the user was directed to the SK Youths Dashboard, where they could change their password. Also, if WB Youths forgot their password, they can change it by entering their registered email account. WB Youths could reserve any available schedule in gym facilities or borrow equipment.

If a user chooses to book a gym reservation, they need to provide certain information, such as the number of people and solo or group preferences. WB Youths then entered their desired schedule. If the schedule was unavailable, they could monitor it through the calendar. Once the contract was confirmed, a feedback option appeared. Providing feedback was optional but helped SKWB Officials gain insights for improvements. WB Youths could navigate through their reservation log to view their schedules and past histories, as well as cancel bookings if needed, thereby completing the process.

Figure 6.B

SKWB Officials System Flow Chart
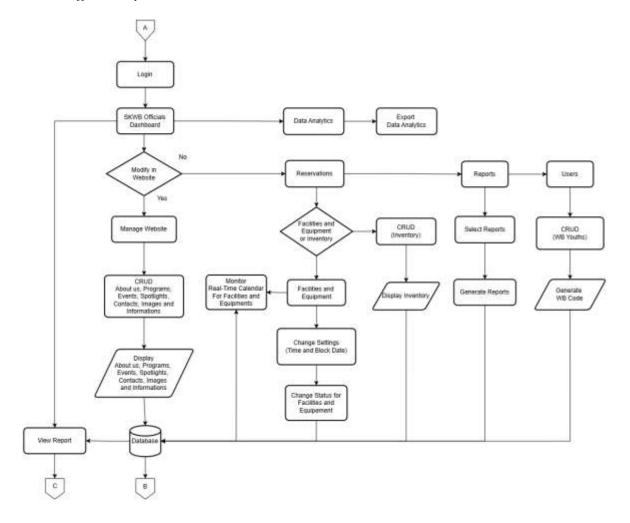
Figure 6. B presents the system flow chart for SKWB Officials within iSKed. The flow chart included the SKWB Officials' role in managing the system. The process began with SKWB Officials logging in. Once logged in, they were directed to the Dashboard, where they could manage content, track records, and view analytics. The "Modify" option within the Dashboard allows SKWB Officials to make necessary changes, such as in the "Manage Website" section, where they could change the website content, such as adding, updating, viewing, or deleting information in sections like About Us, Programs, Events, Spotlights,

Contacts, Images and Other Information. The "Reservations" option enabled SKWB Officials to monitor calendars for facilities and equipment.

Additionally, SKWB Officials can modify schedules and block dates through Time and Date Settings, which can appear on SK Youths calendars. After that, SKWB Officials can modify the status of SK Youths Reservation by selecting Approve or Disapprove for facilities and Return or Not Returned for equipment. SKWB Officials can manage equipment content by adding, updating, and deleting it. Additionally, they can track "Inventory," such as their quantity and status. The "Report" feature allowed SKWB Officials to generate comprehensive reports from the data provided by WB Youths accessing the website. The "Users" option allows SKWB Officials to manage all WB Youths by manually inputting their information to create a WB Youth profile. SKWB Officials can add, view, edit, and archive WB Youths. After creating a WB Youth profile, a code is generated for WB Youth Registration. All information was saved through the database, and SKWB Officials could view their specific reports.

Wireframe

Figure 7
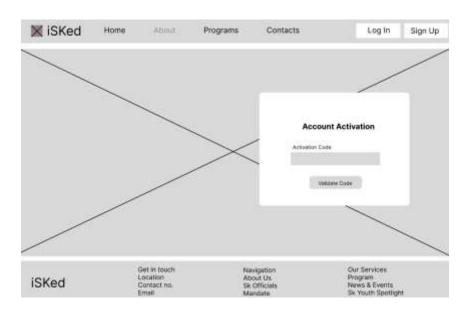
Admin- Adding User Wireframe

The figure above shows a basic wireframe where the admin could add a youth user by entering their personal information into the form fields.

Figure 8

User- Activation Code Wireframe

The figure above shows where the user entered the activation code provided by the system, verifying their account and enabling access to the platform's features and services.

Figure 9
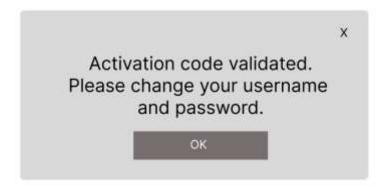
User – Invalid Code Wireframe

The figure above showed a wireframe alert that appeared when a user entered an invalid activation code, prompting them with a message and an option to acknowledge the error by clicking "OK."

Figure 10
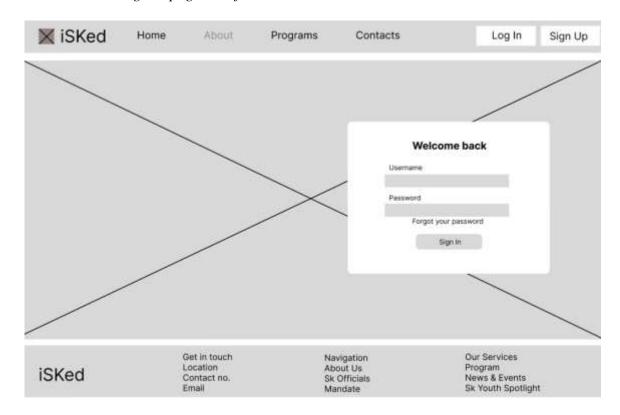
User– Account Activation Successful Wireframe

The figure above shows a wireframe notification confirming that the activation code was valid and instructing the user to change their username and password.

Figure 11
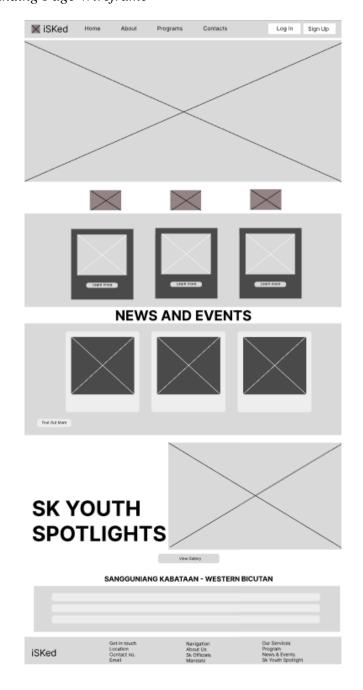
Admin/ User – Sign in page Wireframe

The figure above shows a wireframe of the sign-in page where both admin and users could log in by entering their credentials.

Figure 12
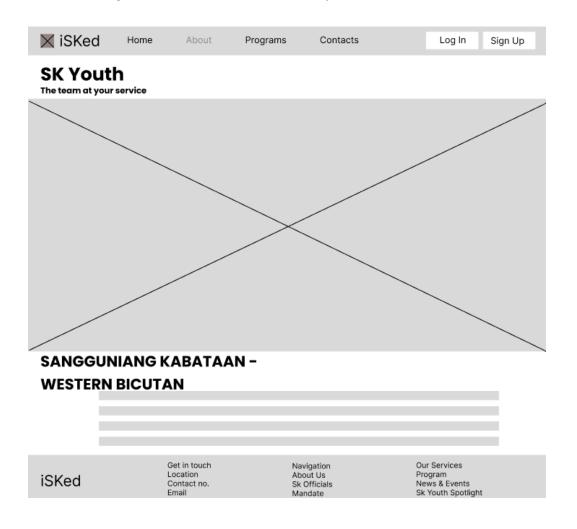
Admin/User - Landing Page Wireframe

The figure above presents a wireframe of the landing page featuring the system's name, navigation bar, sign-in and sign-up buttons, program highlights, news and events, SK Youth spotlights, and contact details.

Figure 13

User About Us Page – View About Us Details Wireframe
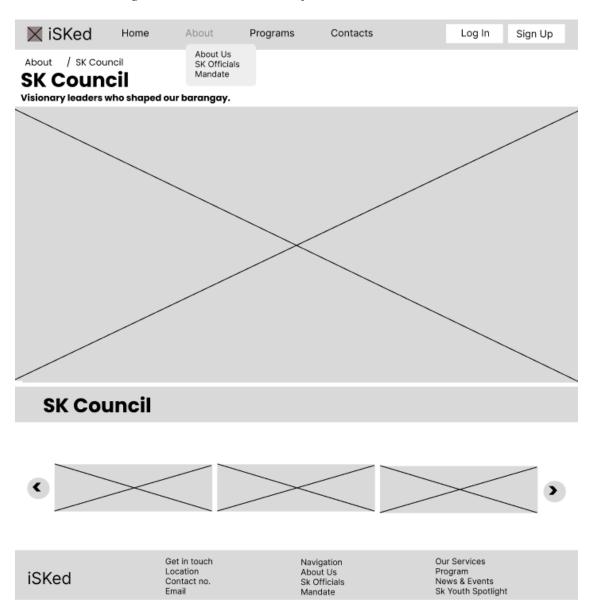
The figure above illustrates the wireframe of the "About Us" page, which featured a prominent header image, the title of the Sangguniang Kabataan of Western Bicutan, and

designated text sections intended to present detailed information about the organization's background and services.

Figure 14
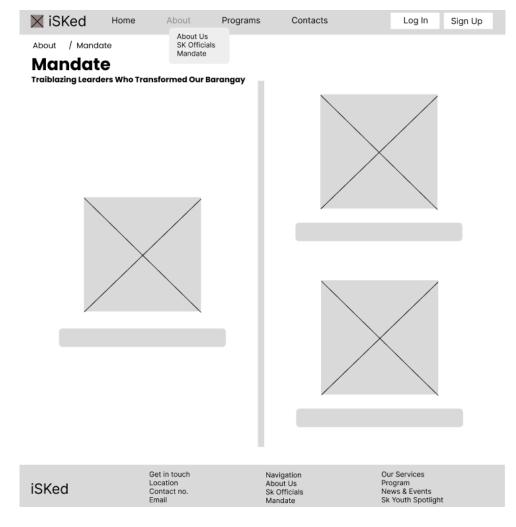
User About Us Page – View Sk Council Wireframe

The figure above shows the wireframe of the "About Us – SK Council" page, which includes a main image section, a short description of the council, and a horizontal image slider used to display photos of SK Council members in an organized layout.

Figure 15

User About Us Page –Mandate Wireframe
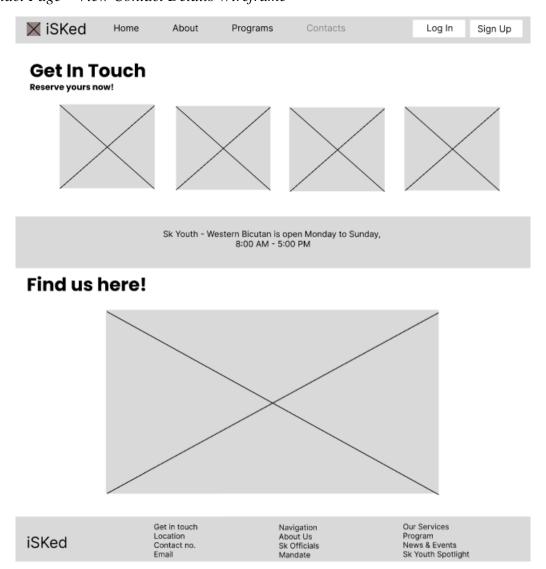
The figure above displays the wireframe of the "Mandate" section under the "About Us" page, incorporating multiple visual and textual placeholders arranged in a two-column

format to convey the mission, vision, and objectives of the Sangguniang Kabataan effectively.

Figure 16
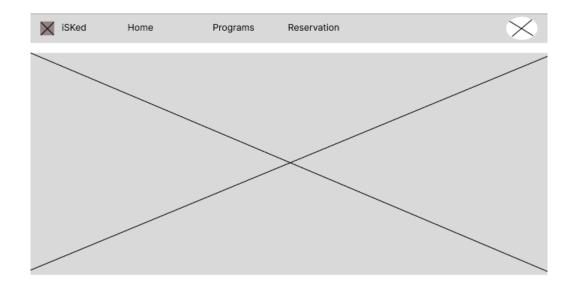
Contact Page – View Contact Details Wireframe

The figure above shows the wireframe layout of the Contact Page, featuring labeled sections for contact options, visual placeholders for icons or images, office hours displayed

in a horizontal banner, and a large map section designed to guide users to the SK Office

location.
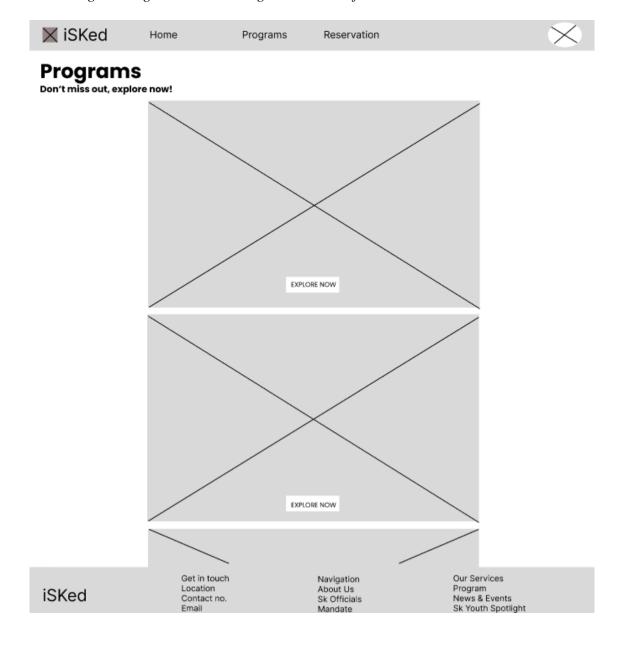
Figure 17

User - Dashboard Wireframe

The figure above shows the wireframe of the user dashboard, which includes a top

navigation bar with labeled links, a profile icon on the upper right, and a large central area

intended for a banner or cover photo representing the user's homepage.

Figure 18
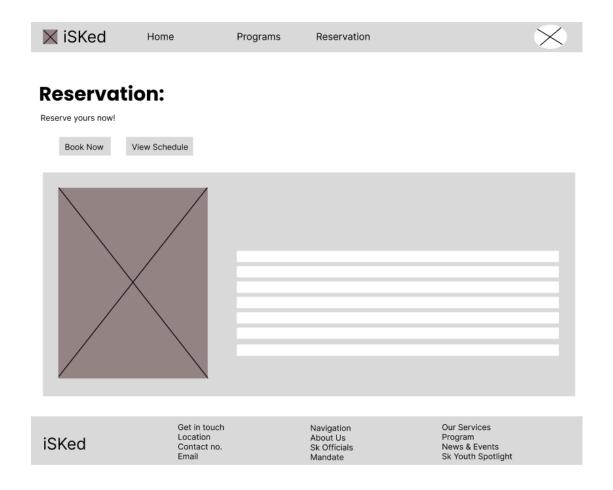
User Program Page – View All Program List Wireframe

The figure above shows the wireframe of the program listing page, which displayed program images arranged vertically with "Explore Now" buttons below each, allowing users to view and explore the available SK programs.

Figure 19

User Program Page – View Program Detail Wireframe

The figure above presented the wireframe of the program detail page, which included a program title, action buttons for booking or viewing schedules, a visual section for an image, and a structured text area for program descriptions and details.

Figure 20

User – Option for Number of Participants Wireframe

The figure above showed a wireframe pop-up that allowed users to select the number of participants by choosing between the "Solo" or "Group" options.

Figure 21

User – Facility Scheduling Wireframe
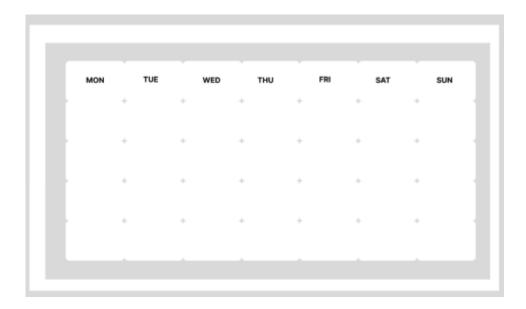
The figure above presents the wireframe of the facility scheduling page, which included dropdown menus, date selectors, and a time selector designed to help users filter and apply preferred reservation details.

Figure 22
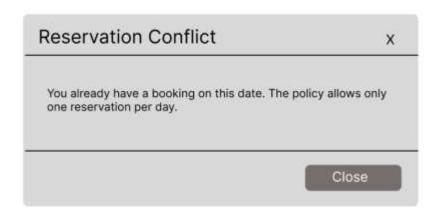
User – Facility Scheduling Calendar View Wireframe

The figure above displays the wireframe of the calendar view, showing the weekly layout from Monday to Sunday with selectable date slots for scheduling facility reservations.

Figure 23
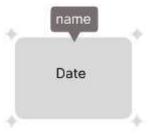
User – Reservation Conflict Wireframe

The figure above shows a wireframe alert that notified users of a reservation conflict, stating that only one booking per day was allowed and included a button to close the message.

Figure 24

User – Calendar View Occupied Wireframe

The figure above shows a wireframe of an occupied calendar slot, where the selected date displayed a label indicating the name of the participant or reservation holder.

Figure 25

User Reservation – Facility Schedule Details Wireframe

The figure above presents the wireframe of the reservation details page, which includes form fields for entering reservation information, such as the program or facility, date, and time, followed by a button next.
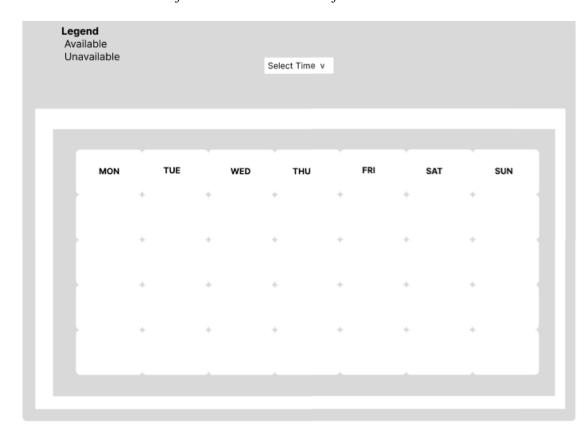
Figure 26

User Reservation – Reservation Confirmation Wireframe

The figure above shows the wireframe of the reservation confirmation page, which provides a summary layout separating reservation details and participant information for users to review before proceeding to waiver.

Figure 27
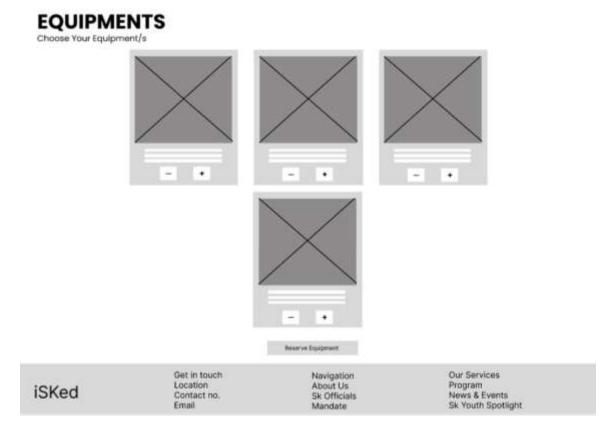
User Reservation – View facilities schedules Wireframe

The figure above displays the wireframe of the weekly facility schedule, showing selectable date slots arranged from Monday to Sunday, along with a legend to indicate available and unavailable slots.

Figure 28

User Reservation –Equipment List Wireframe

The figure above shows the wireframe of the equipment list page, which included boxes for each item displaying an image, name, availability, and quantity controls, along with a reservation button below.
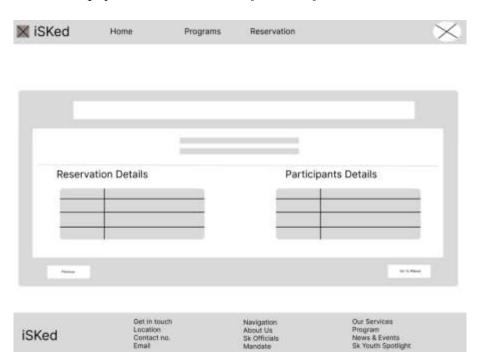
Figure 29

User – No more quantities available notice Wireframe

The figure above shows a wireframe pop-up message that alerted users when they tried to reserve more items than the available quantity, with a close button to dismiss the notice.

Figure 30
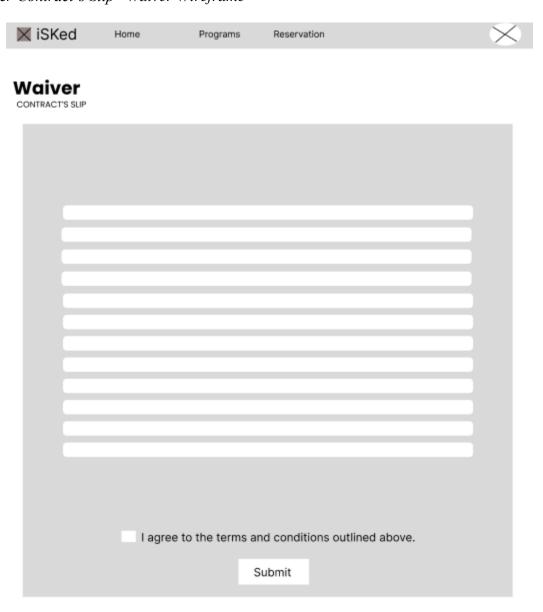
User Reservation –Equipment Review and Confirm Wireframe

The figure above presents the wireframe of the final review page for equipment reservation, summarizing both reservation and participant details side by side before proceeding to the waiver.
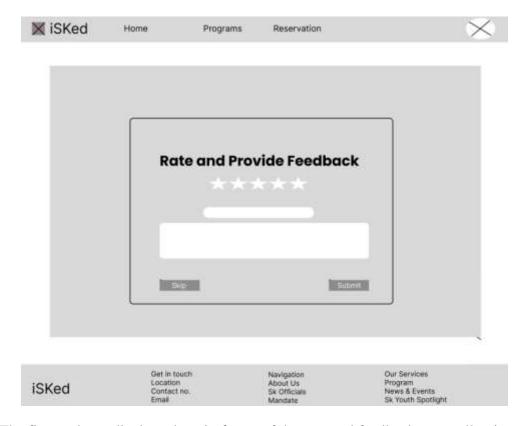
Figure 31

User Contract's Slip –Waiver Wireframe

The figure above shows the waiver page, which users accessed after confirming their reservation, displaying the terms and conditions, including required waivers or agreements to proceed.

Figure 32

User – Rate and Feedback Wireframe

The figure above displays the wireframe of the rate and feedback page, allowing users to rate their experience with stars, enter comments in a text box, and choose to either skip or submit the feedback.

Figure 33

User Home –Reservation Log Wireframe
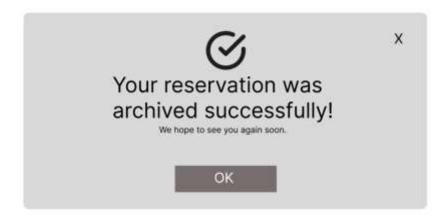
The figure above shows the wireframe of the reservation log page, which lists past reservations in a table format and includes details such as program, dates, time slot, and status, along with an option to delete each entry.

Figure 34

User– Archived Reservation Wireframe

The figure above presents a wireframe pop-up confirming a successful reservation archive, with a message and an "OK" button to close the notification.

Figure 35

User– Reservation Log Option Wireframe

The figure above displays a wireframe dropdown menu within the reservation log, providing filtering options for equipment and facility reservations.

Figure 36

User– Reservation Log Cancellation Wireframe
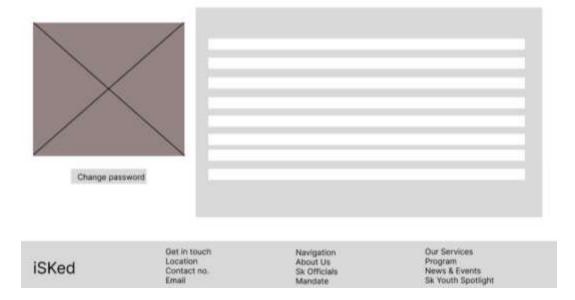
The figure above presented the Cancel Reservation framework, showing reservation details, list of Cancellation reasons, a confimation checkbox, and a Confirm and Cancel button.

Figure 37

User Profile –View Personal Details Wireframe

The figure above displays the wireframe of the user profile page, featuring a profile image placeholder, personal details in text fields, and a button for updating the user's password.

Figure 38

User – Change Password Wireframe

The figure above shows a wireframe pop-up for changing a user's password, which included input fields for the current password, new password, and confirmation, along with options to close or submit the update.

Figure 39

User – Password Updated Wireframe

The figure above presents a wireframe message confirming that the user's password was successfully updated, with a close button to dismiss the notification.

Figure 40
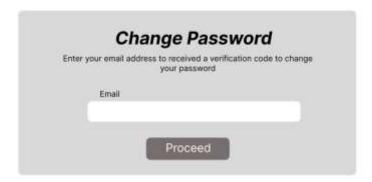
User – Confirm Logout Wireframe

The figure above shows a wireframe confirmation dialog for logout, asking the user to confirm their action by selecting either "Yes" or "No."

Figure 41
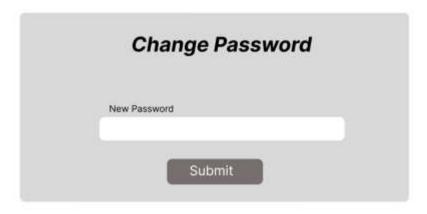
User –Forgot Password Wireframe

The figure above shows the wireframe for the "Forgot Password" page, where users were asked to enter their email address to receive a verification code for password reset.

Figure 42

User –Forgot Password (New Password) Wireframe
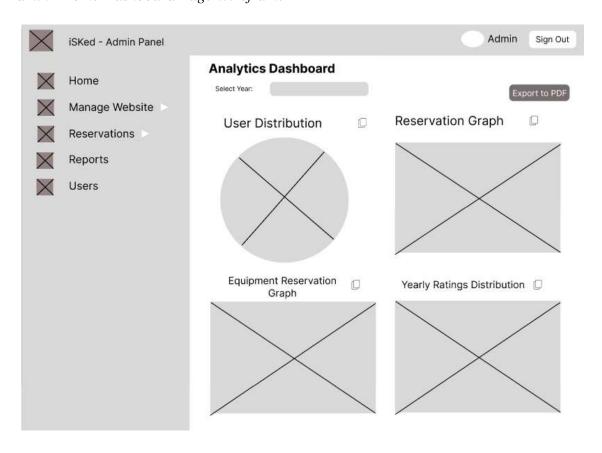
The figure above displays the wireframe for setting a new password, containing a single input field and a submit button to complete the password reset process.

Admin Wireframe

Figure 43
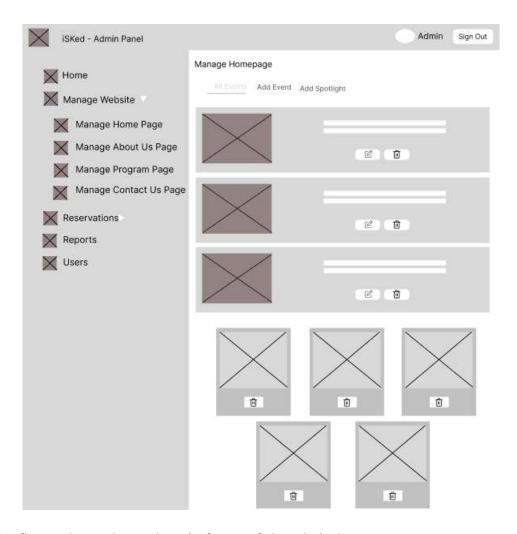
Admin –Home Dashboard Page Wireframe

The figure above shows the wireframe of the admin dashboard, which included graphs for user distribution, reservation trends, and equipment usage, providing an overview of user activity and system performance.

Figure 44

Admin Manage Home Page –All Event Details Wireframe

The figure above shows the wireframe of the admin homepage management section, displaying a list of editable event blocks and a spotlight, each with options to update or delete information.

Figure 45
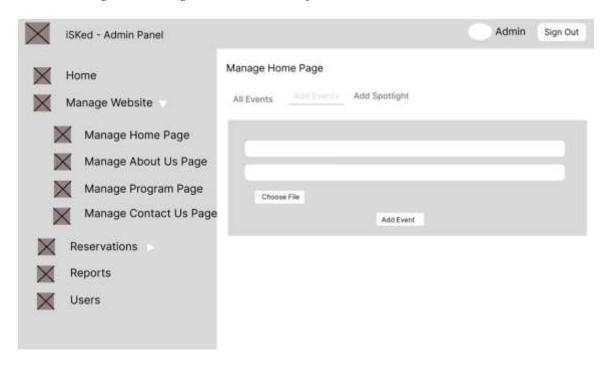
Admin – Manage Homepage Edit Event Wireframe

The figure above displays a wireframe pop-up for editing an event, containing fields

for the event name and description, along with buttons to cancel or save the changes.

Figure 46

Admin Manage Home Page –Add Event Wireframe

The figure above shows the wireframe of the admin page for adding events, which included input fields, a file upload option, and a button to submit the new event.
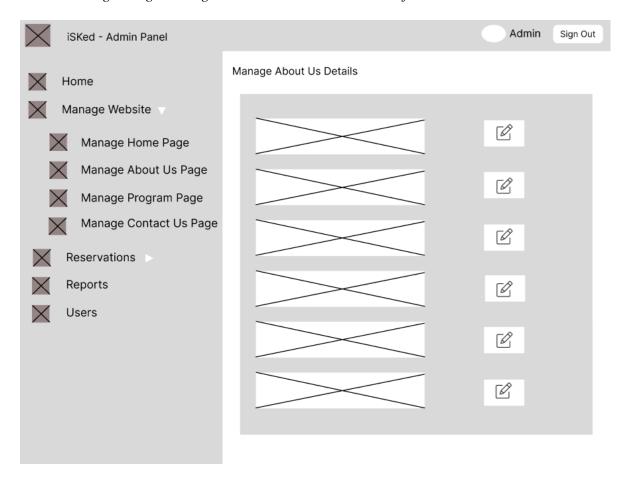
Figure 47

Admin Manage About Us Page –Edit About Us Wireframe
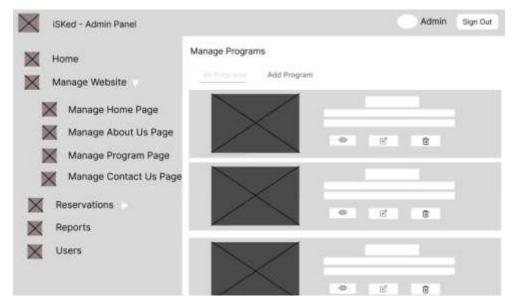
The figure above displays the wireframe of the admin interface for editing the "About Us" page, providing fields for adding SK Council information and a button to update the content.

Figure 48

Admin Manage Program Page – SK Counsil Members Wireframe

The figure above shows the wireframe for managing SK Council member entries, with a vertical list of image placeholders and corresponding edit buttons for each entry.

Figure 49

Admin Manage Program Page –All Programs Wireframe

The figure above presents the wireframe of the admin's program management page, featuring a list of programs with options to view, update, or delete each one.

Figure 50

Admin Manage Program Page – Edit Program Wireframe

The figure above displays the wireframe of the admin interface for editing a program, including fields for the name, description, type, images, and amenities, along with save and cancel buttons.

Figure 51
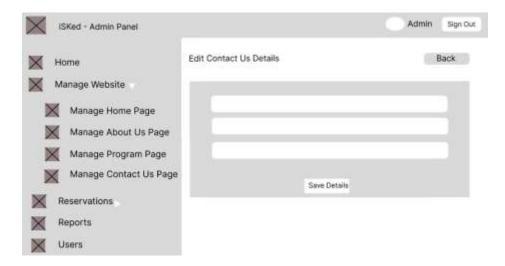
Admin Manage Program Page –Add Program Wireframe

The figure above shows the wireframe of the admin interface for adding a new program, which included input fields for program details and a file upload option, followed by a submission button.

Figure 52
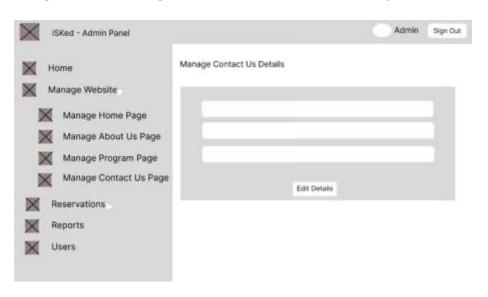
Admin Manage Contact Us Page –Edit Contact Details Wireframe

The figure above displays the wireframe for viewing contact information on the "Manage Contact Us" page, including text fields and an "Edit Details" button for updates.

Figure 53

Admin Manage Contact Us Page –Edit Contact Save Details Wireframe

The figure above shows the wireframe for editing contact details, with editable fields and a button to save the updated information.

Figure 54

Admin Reservation Page –Inventory Equipment List Wireframe

The figure above presents the wireframe of the equipment inventory list, displaying items in a table format with image placeholders and action buttons for each entry.

Figure 55

Admin –Add Inventory Equipment Wireframe

The figure above shows the wireframe for adding an equipment item to the inventory, with form fields for item details and a file upload button.

Figure 56
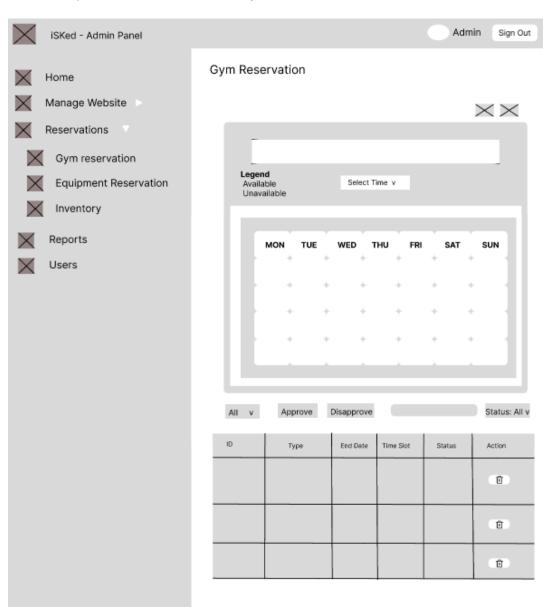
Admin –Inventory Edit Equipment Wireframe

The figure above displays the wireframe for editing an inventory item, similar to the add form, with fields for editing information and saving changes.

Figure 57

Admin – Gym Reservation Panel Wireframe

The figure above shows the wireframe of the gym reservation panel, combining a weekly calendar view and a reservation log table with action buttons to approve or disapprove requests and a drop-down button for status.

Figure 58
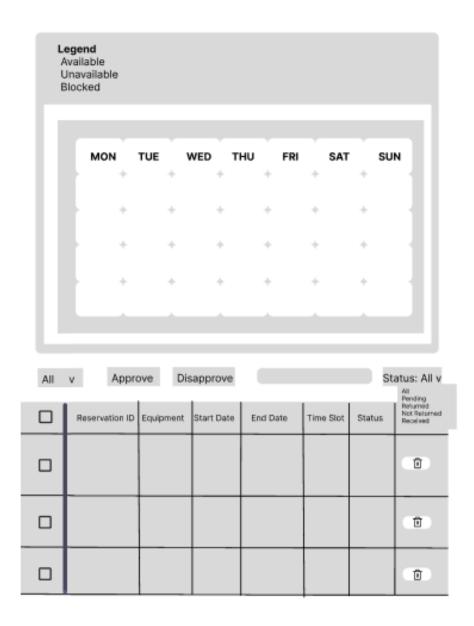
Admin – Customize Time Gap and Block Dates Wireframe

The figure above presents two wireframes for customizing the reservation system: one for adjusting the time gap between bookings and another for blocking specific date ranges.

Figure 59

Admin – Equipment Reservation Panel Wireframe

Equipment Reservation

Legend
Available
Unavailable
Blocked

	MON	TUE	WED	THU	FRI	SAT	SUN

All ∨	Approve	Disapprove		Status: All ∨

All
Pending
Returned
Not Returned
Received

☐	Reservation ID	Equipment	Start Date	End Date	Time Slot	Status	
☐							🗑
☐							🗑
☐							🗑
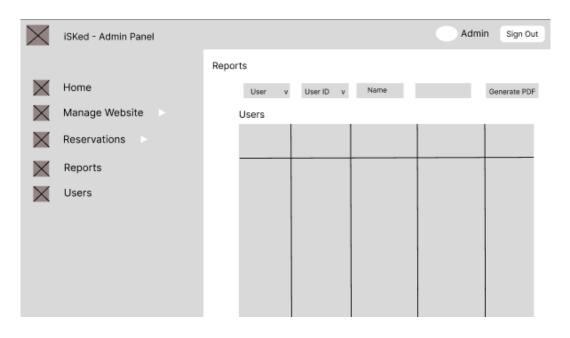
The figure above displays the wireframe of the equipment reservation panel, featuring a calendar and a reservation table where admins could approve, disapprove, or filter equipment bookings.

Figure 60

Admin Reports Page –View Reports Wireframe

The figure above shows the wireframe of the user report page with an option to generate a PDF report.

Figure 61
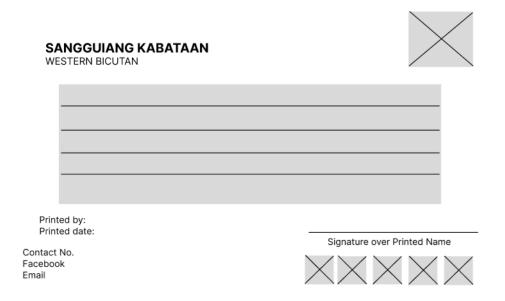
Admin – Reports Dropdown Button Wireframe

The figure above shows the wireframe of the reports dropdown button, allowing admins to choose between report categories such as users, equipment reservations, schedules, and inventory.
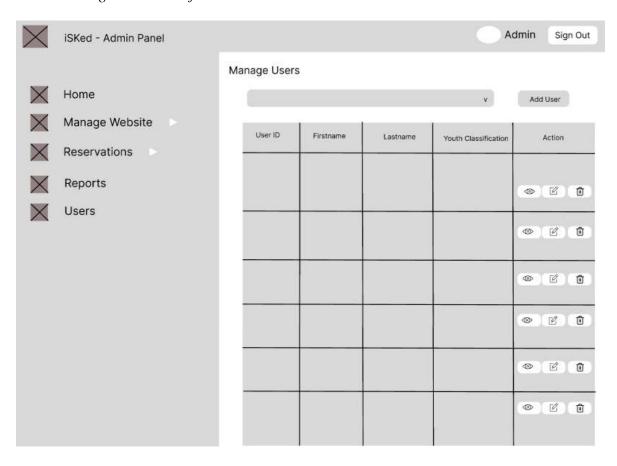
Figure 62

Admin – Reservation Reports PDF Wireframe

The figure above displays the wireframe of the reservation report PDF layout, which included formatted fields for summary data, printed name, contact info, and a signature section.

Figure 63

Admin Manage Users Wireframe

The figure above shows the wireframe of the admin user management page, presenting a searchable user table with options to view, edit, or delete each user.
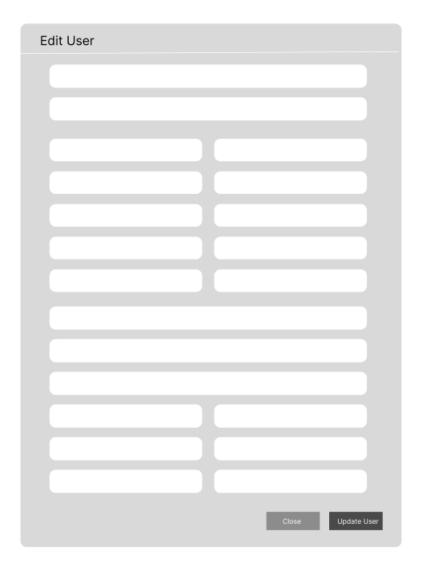
Figure 64

Admin Manage Users- User Details Wireframe

The figure above presents a wireframe modal displaying a user's personal details, including classification, contact information, and the generated code.

Figure 65

Admin Manage Users- Edit User Details Wireframe

The figure above displays the wireframe of the edit user wireframe, providing form fields for updating user information and buttons to either close the form or save the changes.
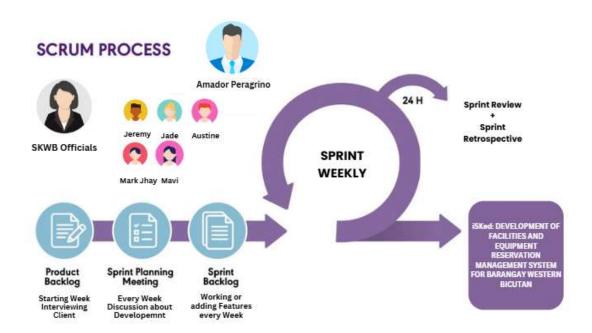
Project Development

This section addressed the methods used to develop the web application in compliance with the design specifications.

The system was developed using Agile Scrum methodology, an approach known for its iterative and collaborative nature. The proponents chose this methodology because it was a framework specifically designed for effective collaboration on complex projects. Agile Scrum Methodology relied on incremental development (Peek, 2023), means the system was built and delivered in stages. This ensured that the final output or product remained relevant and valuable to users by incorporating their needs and responding to any changes that arose during development.

Figure 66

Agile Scrum Methodology Framework

By utilizing Agile Scrum, as shown in Figure 66, the development process became more dynamic and responsive. Below are the core components of Agile Scrum:

Phases:

Sprint Planning Meeting. The iSKed sprints began with this phase, wherein the team defined a goal, selected user stories from the backlogs, and created a development plan for the upcoming sprint.

Daily Scrum Meeting. In this phase, the team kept things moving by quickly sharing updates on what they had accomplished, what they would work on on the daily basis, and any roadblocks encountered.

Sprint Review. At the conclusion of the sprint, the team presented the completed user stories to their capstone adviser and collected feedback. This process ensured that the project met expectations and aligned with the specified requirements.

Sprint Retrospective. Following the sprint review, the team reflected on the sprint, discussing successes, challenges, and areas for improvement in future sprints.

Roles:

Product Owner. The product owner's role fell to the client, wherein these individuals prioritized features within the product backlog and held the final decision on accepting the completed system.

Development Team. The proponents of this system were responsible for delivering the requested and committed product increments.

Scrum Master. These were the facilitators of the team, guaranteeing that the team complied with the Scrum practice, eliminating impediments to the team, and supporting the team throughout their sprints.

Tools:
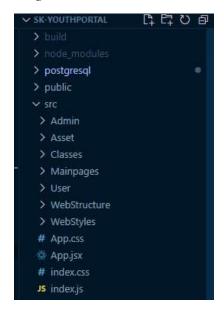
Product Backlog. A structured and prioritized collection of features, enhancements, and system requirements maintained by the product owner, serving as the foundation for development efforts.

Sprint Backlog. A well-defined and time-bound list that focused on a specific development cycle, consisting of prioritized user stories selected from the product backlog. These were the tasks that the development team pledged to complete within the sprint. The sprint backlog also provided a breakdown of the necessary steps to successfully implement each user story.

Project Setup

Figure 67

Folder Configuration According to Features

This section aimed to provide the readers with an overview of the web application development process. The project's purpose was to assist the researchers in methodically and carefully achieving their strategic objectives.

In order to make file names more legible and to let developers know what to expect, the researchers named files and folders using PascalCase. PascalCase was frequently used to name variables, functions, classes, interfaces, and types. The "admin/Categories/index.js" and "import/Admin/Categories" were the two instances of PascalCase file names. Additionally, as PascalCase was the most widely used way in the React community, it was also used for variable names. The folder structure consisted of different folders based on feature, where all of the files associated with that feature were organized.

The following is the description of the project setup:

- **build**. Contained the build artifacts of the project, which were the compiled and minified files ready for deployment.

- **node_modules**. Housed all the dependencies and packages installed via npm (Node Package Manager) that were required for the project to run.

- **public**. Typically contained static assets that were publicly accessible, such as HTML files and other resources, and also stored images.

- **src**. The source folder is where the main application code is resided. It contained several subfolders and files:

    1. **Admin**: Contained code related to the administrative functionalities of the application.

    2. **Asset**: Stored various assets used in the project, such as images, fonts, or other media files.

- **Classes**. Contained classes used in the project, such as Navbar for both Users and Admins Footers.

- **Mainpages**. Included the main pages or components of the application, such as Home, About, Contacts, etc.

- **User**. Contained code related to user functionalities, such as user profiles, reservation logs, and more.

- **WebStructure**. Included are files that define the structure of the web application, such as layout components or routing configurations like AuthContext and Codex.

- **WebStyles**. Contained CSS or other styling files used to style the web application.

  App.css. A CSS file that contained styles for the application.

- **App.jsx**. A JavaScript XML (JSX) file that likely contained the main React component for the application.

- **index.css**. Another CSS file that contained global styles for the application.

- **index.js**. A JavaScript file that likely served as the entry point for the application, where the main rendering logic was defined.

- **.env**.An environment file that contained environment-specific variables, such as API keys, database URLs, and other configuration settings.

Program Coding

The web application was developed using ReactJS and Javascript, with GitHub as the collaborative version control system and PostgreSQL as the database. The web application was developed using the following steps:

1. A folder was created, and in the Windows terminal, a specific folder was entered to implement npx create-react-app.

2. The design was created, and code was implemented for the interface and functions.

3. All necessary libraries were imported into the Visual Studio IDE.

4. Users were able to browse available real-time slots for facility reservations and could navigate and book equipment for their preferred day.

5. Detailed information about each program was presented, including photos, capacity, amenities, and any specific rules or regulations.

6. Once users confirmed their reservation slot, they encountered a waiver and contract.

7. The system stored user information, program and event details, reservation data, and more.

8. It tracked the real-time availability of facility and equipment reservations through a calendar based on existing reservations and prevented double reservations.

9. An administrative dashboard was provided for data analytics, managing programs, reservations, and user accounts.

10. The Admin Module had access to all key features of the web application.

11. All transactions and information were saved in both the users' and admins' transaction history.

Operation and Testing Procedures

To ensure that the system produced the required outcomes, its features and functionalities were tested. These tests were designed to guarantee that every feature and function of the system worked and operated as expected.

Functional Suitability Testing was undertaken to confirm that the web application's functionalities work as intended. The test cases include a set of inputs, an execution precondition, and expected results. A test case form, illustrated in Table 1, was used.

Table 1

Sample Test Case

Test Case ID		ISK-FUN-001	UC Reference	User Registration and Verification
Objective		Test user registration functionality validates the SKWB unique code, and generates login credentials.		
Assumptions/Preconditions		1. The user was a youth resident of Western Bicutan and has verified official records at the barangay office. 2. The user has a valid SKWB unique code retrieved from the barangay office.		
Actions		**Expected Result**	**Actual Result**	
1.Enter the SKWB unique code verification, then click "Validate Code". 2. Change password and username before clicking the "Create Account".		1. The system validates the SKWB unique code correctly. 2. The system confirms the username and password match the confirmation field.	1. The system validated the SKWB unique code correctly, and a username and password appeared, which needed to be changed before signing in or logging in. 2. Once the system confirmed the new username and password, a success message box appeared.	
Status	PASSED	**Severity**		**Priority**	

The table above outlined a test scenario for evaluating the system's user registration and verification functionality. The test case ID, "ISK-FUN-001," was associated with a specific use case. The major goal of this test case was to ensure that the system correctly handled user registration and verification. The requirements for this test case included the user

being a youth resident in Western Bicutan, having confirmed official records at the barangay office, and possessing a valid SKWB unique code given by the barangay office.

The following actions were performed during testing:

1. The user entered the SKWB's unique code for verification and then clicked the "Validate Code" button.

2. After changing the username and password, the user pressed the "Create Account" button.

The expected outcome was that the system would successfully validate the SKWB unique code and ensure that the entered username and password matched their respective confirmation fields. In this instance, the system validated the SKWB unique code correctly, and the username and password were displayed, requiring the user to update them before signing in or logging in.

The test case status was marked as "PASSED," indicating that the test ran successfully and met the expected results. While the severity and priority fields were not filled out in this particular example, they are typically used to assess the impact and urgency of any issues identified during testing. This structured approach ensured that the web application's functionality was rigorously tested and validated.

Table 2

Classification of Error Severity

Severity	Description
Critical	The problem signifies that the process has been completely stopped and cannot continue until it is resolved.
Major	The problem causes the system to crash. Nevertheless, some system components are still functional.
Minor	The problem does not result in any significant system failure.

The table classified errors based on their severity. The severity level was divided into three categories: critical, major, and minor, each with a separate description. This table was used to assess the magnitude and severity of system failures. By categorizing errors into different severity levels, stakeholders may better prioritize and deploy resources to resolve the issues. Resolving issues in each category increased the system's overall reliability and performance.

Table 3

Classification of Error Priority Levels

Priority	Description

High	The problem needs to be resolved as soon as possible since it significantly affects the application.
Medium	The problem should be fixed throughout the normal course of development.
Low	The problem must be resolved when a more crucial feature is taken care of.

The table provided a detailed error priority level classification. The error priority was divided into three categories: High, Medium, and Low, each with its corresponding description. This table served as a guide for prioritizing and allocating resources to address issues based on their impact and urgency. By assigning the appropriate priority level to each error, development teams could efficiently manage their workflow, ensuring that critical errors were addressed promptly while less significant problems were handled at the appropriate stages of the development cycle.

Functionality and Reliability Testing

Test cases were gathered and presented in Table 4, which summarized the functionality test cases for each use case. Table 5 provides a summary of the reliability test cases, outlining the procedures and expected outcomes to ensure system performance and dependability.

Table 4

Overall Summary of Functionality Test Cases

Use Case	No. of Test Cases
Registered Youth	
Unregistered User	
Admin/Client	
Total	

The table categorized the test cases based on user roles, which included "Registered Youth," "Unregistered User," and "Admin/Client. "The "Total" row displayed the total number of test cases across all user roles. This table served as a placeholder for capturing and organizing test case information efficiently.

Table 5

Reliability Test Cases Summary

Test Case ID	Objectives

The test case ID uniquely identified each test case, ensuring efficient tracking and organization. Each test case outlined its objectives and intended purpose, particularly emphasizing the reliability aspects being evaluated or validated.

Table 6

Testing Procedure for Functionality Suitability

Modules	Steps to be taken	Expected Output
1. Sign up	1. From the landing page of the website, click sign up from the navigation bar. 2. Enter the provided unique code in the input field. 3. Click the "Validate Code" button to proceed. 4. Set up password and username. 5. Click the "Create Account" button.	1. The user was directed to the sign-up page and was prompted to input their unique code. 2. The system verified the code and either allowed the next step or displayed an error for an invalid or missing code. 3. The system accepted valid username and password input and once signed up, the user was directed to the system's dashboard.
2. Sign in	1. From the landing page of the website, click the "Log in" button from the navigation bar. 2. Enter existing user details. 3. Click the sign-in button.	1. The user who input the correct details was redirected to the home page. 2. An error message was displayed to the user who inputted incorrect details and stayed on the sign-in page.
3. Facility or Equipment Reservation	1. Click the desired equipment or facility for reservation. 2. Select a booking option, either "solo" or "group" for facility reservation. This does not apply to equipment reservations, but	1. The user clicked on the desired equipment or facility for reservation. 2. The reservation type for the facility specified whether the person making the reservation

	choose a quantity for this reservation. 3. Select a specific time and date of reservation from the calendar. 4. Click the "Apply Dates" button. 5. After confirming the details, click "Go to Waiver".	was alone or accompanied by others. As for the equipment, the user chooses its quantity. 3. The user selected a specific date and time for the reservation from the calendar. 4. The user clicked the "Apply Dates" button to confirm the selected time and date. 5. After reviewing and confirming the reservation details, the user clicked the "Go to Waiver" button to proceed to the next step. 6. Once the user clicked the checkbox, it was redirected to the reservation log.
4. Manage Profile	1. After signing in, click the profile button on the right side of the navigation bar. 2. Clicked "Profile". 3. User may change their password by clicking the "Change password."	1. By clicking the "Profile," a user was directed to the profile page wherein they can view their credentials. 2. If the users decided to change their password, they were fronted by the modal to set their new password. 3. After changing the account password, the account stayed on the "Profile" page.
5. Manage Home Page (Admin)	1. The "Manage Website" link was clicked from the navigation bar. 2. "Manage Home Page" was clicked under the "Manage Website" link. 3. The default page, "Manage Details," allowed the admin to edit information on the "Home" page. 4. "All Announcements/Events" was clicked at the top of the page to view all announcements/events and spotlights.	1. When the "Edit Details" button was clicked, and there were modified details, the "Save Details" button saved the changes on the website. 2. After the "Add" button was clicked, the new announcement/event was instantly presented on the website. 3. Clicking the "Delete" button for the specific announcement/event

	5. The admin viewed the details of existing announcements/events on the right side of the image of an announcement/event. 6. The admin chose to "Edit Details" or "Delete" the announcement/event. 7. "Add New" was clicked from the top of the page to add a new announcement/event. 8. The details of the new announcement/event were filled in the provided form. 9. The "Add" button was clicked. 10. "Add Spotlight" was clicked from the top of the page to add a new spotlight. 11. The details of the new spotlight were filled in the provided form. 12. The "Add Spotlight" button was clicked. 13. The "Remove Spotlight Image" button at the lower content part of the "All Announcements/Events" was clicked to remove a spotlight.	automatically removes it from the website. 4. After the "Add Spotlight" button was clicked, the new spotlight was instantly presented on the website. 5. Clicking the "Remove Spotlight Image" button for the specific spotlight container automatically removes it from the website.
6. Manage About Us Page (Admin)	1. "Manage About Us Page" was clicked under the "Manage Website" link. 2. After clicking, the admin viewed all the details displayed on the "About Us" page. 3. The admin edited the current details by clicking the "Edit Details" button. 4. After modifying the details, the "Save Details" button was clicked. 5. To replace the SK member image in the SK Council Members section at the lower part of the content, the "Edit" button on the	1. By clicking the "Save Details" button after modifying the existing details, the information was instantly updated on the website. 2. By clicking the "Save Member" button after replacing the SK Council Member image, the update was instantly reflected on the website.

	right side of the selected image was clicked. 6. The "Save Member" button was clicked to save the changes.	
7. Manage Program Page (Admin)	1. "Manage Program Page" was clicked under the "Manage Website" link. 2. The default landing page displayed after clicking was "All Programs," which showed all existing programs from the website. 3. A "View Details" button was available for the selected program to view all of its details. 4. After clicking "View Details," the admin had the option to click two buttons: "Edit Details" and "Delete Program." 5. "Add Program" was clicked from the top of the page to add a new program. 6. After clicking, the admin entered all the details about the new program into the form displayed on the website. 7. The "Add Program" button was clicked to save the new program.	1. When the "Edit Details" button was clicked, clicking the "Save Details" button for the modified program updated the details on the website automatically. 2. After the "Add Program" button was clicked, the new program instantly appeared on the website. 3. Clicking the "Delete Program" button for the selected program instantly removed it from the website.
8. Manage Contact Us Page (Admin)	1. "Manage Contact Us Page" was clicked under the "Manage Website" link. 2. After clicking, the admin viewed all the details displayed on the "Contact Us" page. 3. The admin edited the current details by clicking the "Edit Details" button. 4. The "Save Changes" button was clicked to save the updates.	1. By clicking the "Save Details" button, the updated details were automatically reflected on the website.
9. Reservations (Admin)	1. The "Reports" link was clicked from the navigation bar.	1. When the "Edit" button was clicked, clicking the "Save

	2. The "Gym Reservation" link was clicked under the "Reports" link. 3. The day in the calendar with a number was clicked to view details about the reservation. 4. A check was placed on the checkbox in the Gym Reservation table to select a list of reservations for the Gym. 5. The "Approved" button was clicked to approve the reservation, and the "Disapproved" button was clicked to disapprove the reservation. 6. The "Equipment Reservation" link was clicked under the "Reports" link. 7. The day in the calendar with a number was clicked to view details about the reservation. 8. A check was placed on the checkbox in the Equipment Reservation table to select a list of reservations for Equipment. 9. The "Approved" button was clicked to approve the reservation, and the "Disapproved" button was clicked to disapprove the reservation. 10. If the equipment was returned, the "Returned" button was clicked on the specific equipment reservation status; the "Not Returned" button was clicked when the equipment was not yet returned. 11. The "Inventory" link was clicked under the "Reports" link.	Changes" button updated the equipment details on the website automatically. 2. After the "Add Item" button was clicked, the new equipment instantly appeared on the website. 3. Clicking the "Delete" button for the selected equipment instantly removed it from the website.

	12. "Add Equipment" was clicked to add new equipment to the list. 13. The provided form for the new equipment was filled out. 14. The "Add Item" button was clicked. 15. The "Edit" button was clicked on specific equipment to edit its details. 16. The provided form for editing the current details of equipment was filled out. 17. The "Save Changes" button was clicked. 18. The "Delete" button was clicked on specific equipment to delete its details from the list.	
10. Reports (Admin)	1. The "Reports" link was clicked from the navigation bar. 2. After clicking, all the reports based on the category were displayed. 3. The reports were classified into four categories: Users, Equipment Reservations, Schedules, and Inventory, by selecting the option at the top part of the page. 4. The reports were also viewable by date, monthly, or annually by selecting the option at the top part of the page. 5. To generate reports, the Admin Name text box located at the top part of the page was filled out. 6. The "Generate Report" button was clicked and located at the top of the page.	1. Clicking the "Generate Report" button instantly downloaded a PDF file related to the reports of the selected category and date.
11. Users (Admin)	1. The "Users" link was clicked from the navigation bar.	1. When the "Update User" button was clicked, all the

	2. After clicking, all the existing users were displayed. 3. The users were classified into three categories, Child Youth, Core Youth, and Adult Youth, by selecting the option at the top part of the page. 4. The "View" button of a specific user was clicked to view all of the user's details. 5. The "Edit" button of a specific user was clicked to edit the user's details. 6. The "Update User" button was clicked to save the changes. 7. The "Delete" button of a specific user was clicked to delete the user's details from the system.	changes edited by the admin were saved. 2. By clicking the "Delete User" button, the user's account was no longer accessible on the website.

Functionality Testing was conducted to evaluate the system's behavior under defined environmental conditions, as well as its effectiveness over a specified period of time. The table above presented a compilation of test cases for various modules and functionalities of the system application. Each row represented a distinct test case, with the columns documenting the relevant information, such as the test ID, the procedures to be followed, and the expected outcome of each test case. The table covered aspects such as user sign-up and sign-in, equipment or facility reservations, user profile management, and administrative tasks for the admin. Organizing and documenting the test cases in a table made it easier to manage and track the testing process, ensuring that each feature was thoroughly tested and validated according to the defined processes and expected results.

Table 7

Testing Execution Summary

Test Execution	Expected Results	Active Result Cycle 1	Cycle 2
No. of Test Cases Executed	100%		
Results of Test Cases			
Passed	100%		
Failed	0%		
No. of Test Cases Not Executed	0%		

The table summarized the test execution, showing that all test cases had been completed with no tests left unexecuted. The expected results column was empty and needed to be filled with the expected outcomes for each test case. The test results indicated that all tests had passed. This table served as a status report, affirming the successful and thorough completion of the testing process.

Evaluation Procedure of the Study

The core evaluation was guided by the principles outlined in the ISO/IEC 25010 standard to provide a structured framework for assessing software quality. The evaluation also used stratified random sampling to select participants based on the specific criteria relevant to the research objectives. The system was demonstrated, and the respondents received a survey in return.

The evaluation procedure was conducted to assess the acceptability of the proposed system as follows:

1. Thirty (30) youth participants, including the SK members from Western Bicutan, Taguig City, and IT professionals and IT students, were selected. These participants had varying levels of technical proficiency and represented different age groups. They were also involved in SK activities to ensure a comprehensive evaluation. The selection included individuals who typically use a range of devices for accessibility testing, such as mobile phones, tablets, and computers.

2. Explained and demonstrated how the system works.

3. The system was made available to the sample participants.

4. Participants then complete individual evaluations using provided sheets with a 4-point Likert Scale, where four (4) is the highest and one (1) is the lowest.

5. The evaluation sheet was then processed, and the data was tabulated to determine the mean rating, ensuring accurate and efficient processing of results.

6. After, the adjectival ratings for the mean ratings were also interpreted using the Likert Scale shown in Table 8 below.

Table 8

Four-point Likert Scale

Scale	Descriptive Rating
4	Highly Acceptable
3	Very Acceptable
2	Fairly Acceptable
1	Not Acceptable

The table above showed that the evaluation utilized a 4-point acceptability scale to assess various criteria related to the iSKed responsive web application. This scale assigned higher numerical values to indicate a greater level of success in meeting those criteria. To ensure clear communication and a consistent evaluation process, the scale incorporated descriptive ratings alongside the numerical scores. These descriptive labels provided a qualitative understanding of what each numerical value represents. For instance, a score of "4" translated to "Highly Acceptable," signifying that the criterion is met exceptionally well. Conversely, a score of "1" translated to "Not Acceptable," indicating the criterion failed to meet essential requirements. Highly rated criteria highlighted areas of strength, while lower ratings pinpointed aspects that required improvement.

Table 9

The Range of Mean Ratings and the Equivalent Descriptive Rating

Scale	Descriptive Rating
3.26 - 4.00	Highly Acceptable
2.51 - 3.25	Very Acceptable
1.76 - 2.50	Fairly Acceptable
1.00 - 1.75	Not Acceptable

Each row in the table represented a numerical range, with the corresponding column offering a descriptive rating. This scale was utilized to evaluate or grade something based on its level of acceptability, with each rating tied to a particular numerical range. For instance, a value between 3.26 and 4.0 was deemed "Highly Acceptable." Similarly, values in other ranges were categorized as "Very Acceptable," "Fairly Acceptable," or "Not Acceptable," according to the given descriptors.

Chapter 4

RESULT AND DISCUSSION

This chapter presents the findings from the system evaluation and provides an analysis of the project. It encompasses the system's description, structural design, key capabilities, identified limitations, and a detailed discussion of the project evaluation results.

Project Description

iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan is designed to streamline the reservation and management processes for facilities and equipment. The system includes key features such as a dynamic service creation module for adding and updating services, an automated scheduling system to prevent double bookings, and an inventory monitoring module to track equipment availability and maintenance. Additionally, it generates comprehensive reports on reservation trends and utilization metrics, enabling informed resource planning and effective community event organization.

To align with SK-specific policies, iSKed incorporates approval processes for facility and equipment reservations, ensuring proper governance and accountability. User registration requires generating a secure, unique activation code based on SKWB records, which users obtain from the barangay office to activate their accounts. The system supports multiple user roles, enhancing its ability to manage and optimize resources effectively.
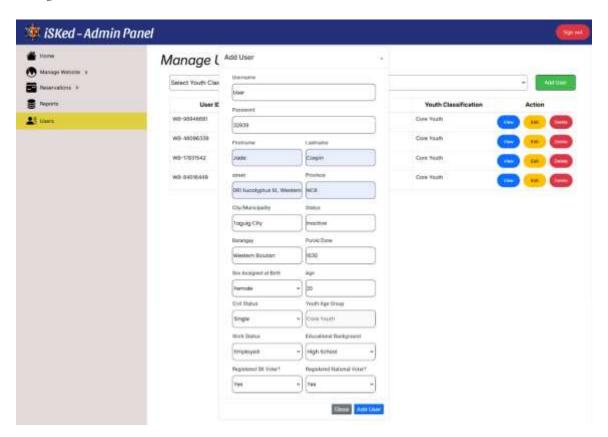
The application employs ReactJS, CSS, and Bootstrap for front-end development, while the back end is built using ExpressJS, NodeJS, and Postman. PostgreSQL serves as the database management system, ensuring reliable and scalable data handling. Visual Studio Code facilitates development as the integrated development environment (IDE) and GitHub for version control. Being a web-based platform, iSKed requires an active internet connection to operate seamlessly.

By combining modern technologies and adhering to SK governance policies, iSKed aims to provide a user-friendly, efficient, and transparent platform to address the reservation needs of Barangay Western Bicutan.

Project Structure

This section presents an overview of the project through detailed screenshots of the web application. These visuals highlight the user interface, showcasing key elements, features, and interactions within the system. The screenshots provide readers with a clear understanding of the design choices, functionality, and workflow of the project.

Figure 68

Adding Youth User

The figure above illustrates the addition of verified youth users where the admin has overall control. This contains essential information such as name, date of birth, status, and educational background.

Figure 69

User – Account activation Interface

This figure illustrates the process users underwent before accessing the website. They were required to obtain an account activation code, which was provided by the barangay or SK officials/administrators. This step ensured that only authorized users could proceed to access the platform, enhancing security and maintaining the integrity of the reservation system.

Figure 70
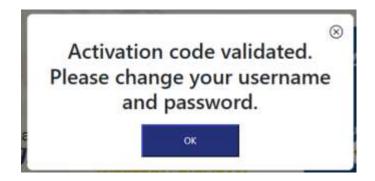
User – Invalid Code

The 'Invalid Activation Code' notification appeared when the user entered an incorrect or invalid activation code during the account activation process. This message alerted the user that the code provided did not match the system records, prompting them to recheck the code or request a new one to complete the activation process.

Figure 71

User – Account Activation Successful

The 'Activation Code Validated' page confirmed that the user had successfully entered a valid activation code. On this page, the user was prompted to set up their account by creating a new username and password to complete the activation process and gain access to the system.

Figure 72

Admin / User – Sign-in page Interface

Figure 72 illustrates the Sign-In Page of the admin panel, providing administrators with a secure and user-friendly interface to log into their accounts. This page required administrators to enter their username and password, ensuring that only authorized personnel could access the system and protecting sensitive data from unauthorized access.

Figure 73

User – Landing page Interface
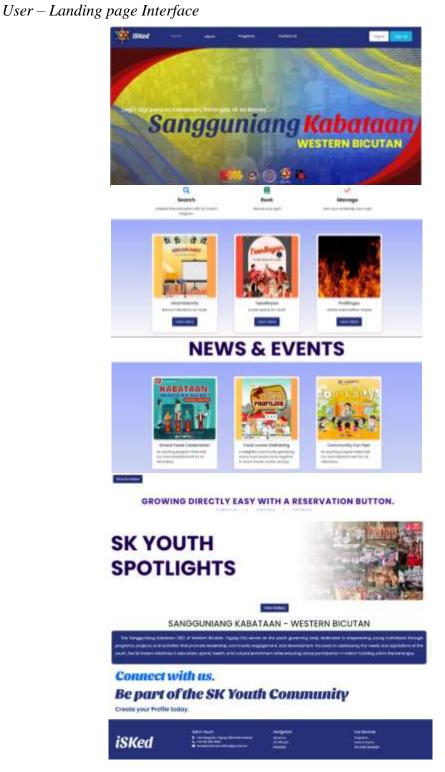
Figure 73 represents the landing page that users first encounter upon visiting the Sangguniang Kabataan (SK) Western Bicutan website to book facilities and equipment. The design prioritizes a user-friendly interface with clear navigation options and prominently displays key features such as available services and a quick-access booking system at the top of the page.

As users scroll down, additional elements come into view, including a dedicated section for news and spotlights to keep users informed about updates and ongoing activities. Further down, the page provides essential details such as the barangay's location, contact information, and operating hours, ensuring users have all the necessary information for their reservations and inquiries. This thoughtfully designed landing page serves as the primary gateway, enabling users to efficiently navigate the platform and fulfill their reservation needs.

Figure 74
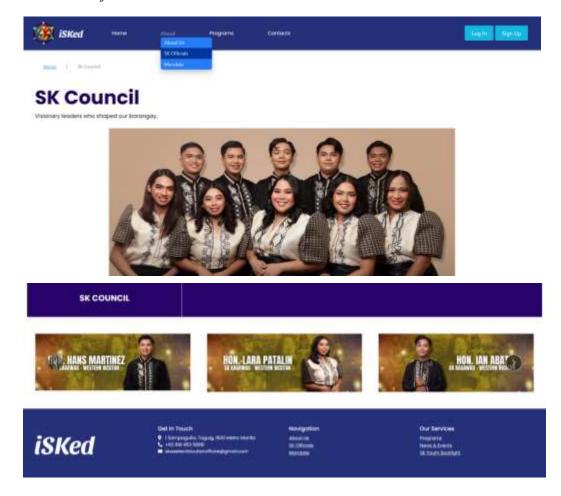
User – About Us Page Interface

This figure shows the 'About Us' page that users could view. This page displayed details such as pictures of the SK officials, their mandate, and information about the organization. It was designed to give users a clear understanding of the SK's purpose and the people behind its operations.

Figure 75

SK Council Interface

The above image was from the dropdown option from the "About" navigation bar. This was where the complete pictures with the names of the council were shown.

Figure 76
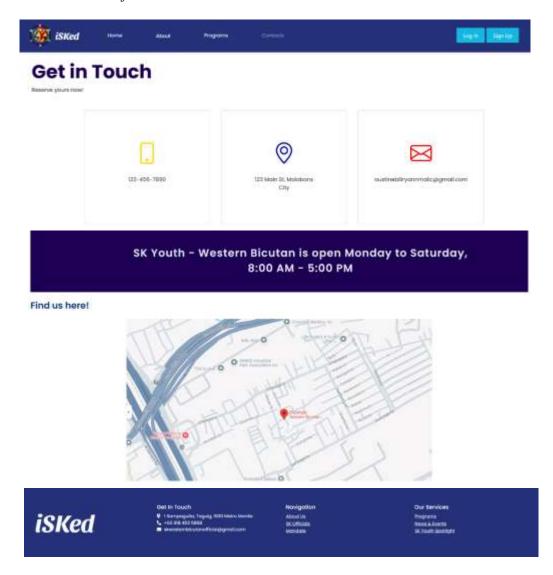
SK Mandate Interface

The above illustration was the mandate of the SK of Western Bicutan. This included their mission, vision, and objectives.

Figure 77

Contact Details Interface

The figure above presents the contact details interface where the contact information of the SKWB was shown. This includes the opening and closing hours, along with their pinned location.

Figure 78

User – Homepage Interface

Figure 78 shows the homepage that users saw after they logged into the website. The homepage was the main screen where users could see their profile, available facilities and equipment for booking, recent announcements, and quick links to start reservations or check schedules. It was designed to be simple and easy to use so that users could quickly find what they needed.

Figure 79

User – Program List Page Interface

Figure 79 shows the page where users could see a list of programs offered by SK Western Bicutan. Each program had a button labeled 'Explore Now,' which users could click to view more details about that specific program.

Figure 80

User – Reservation Program Description Interface

This figure shows the page that appeared after clicking the 'Explore Now' button. It displayed the program's description along with two buttons: 'Book Now' and 'View Schedule.' These options allowed the user to decide whether to proceed with booking or to check the program's schedule first.

Figure 81
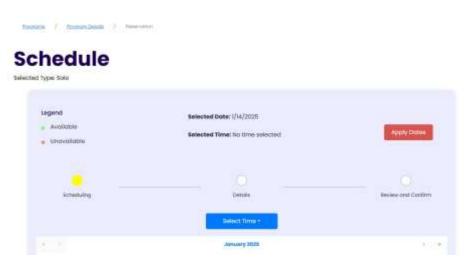
User – Option for Number of Participants

This figure shows the page that appeared after the user decided to book. The user was required to choose whether they were booked as an individual (solo) or as part of a group, with a maximum of 5 people. This step was necessary to proceed to the next part of the booking process.

Figure 82

User – Facility Scheduling Interface

This figure shows the page where the user could see the date and time they selected for their booking, along with an 'Apply Dates' button. This allowed them to confirm their choices before proceeding.

Figure 83

User – Facility Scheduling Calendar View Interface

Figure 83 displays the calendar page, where users can choose the date and time for their booking. Dates marked in red were unavailable, while green dates were available. To book for a single day, the user simply needed to double-click the specific date, such as January 14. For booking a week, the user could click the start date (e.g., January 14) and then select the end date (e.g., January 18) to complete the selection.

Figure 84
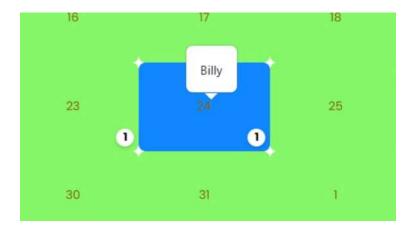
User–Reservation Conflict

This notice appeared when the user attempted to book a date and time that had already been reserved. It informed the user that only one reservation per day was allowed, and the selected time slot was no longer available.

Figure 85
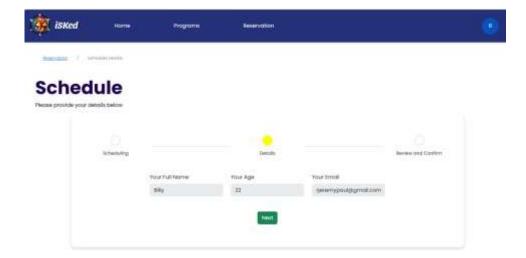
User – Calendar View Occupied

This feature allowed users or youth to see the details of bookings on specific dates. By viewing the calendar, users could identify which dates were already reserved and who made the reservation, promoting transparency and helping users plan their bookings accordingly.
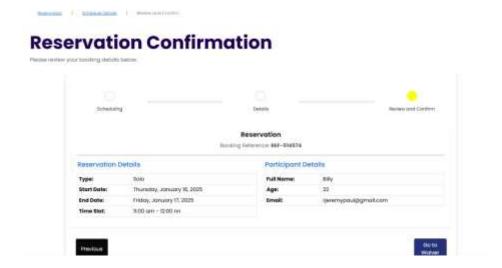
Figure 86

User – Facility Schedule Details Interface

This figure shows the page that appeared after the user selected a date and time. It displayed the user's details, such as their full name, age, and email. To proceed to the next step, the user would click the 'Next' button.

Figure 87

User – Facility Reservation Confirmation Interface

This figure shows the page that appeared after scheduling and entering the details. It displayed the review and confirmation section, where the user could see their reference number and all the reservation details. The user could review everything to ensure the information was correct before confirming the reservation.

Figure 88
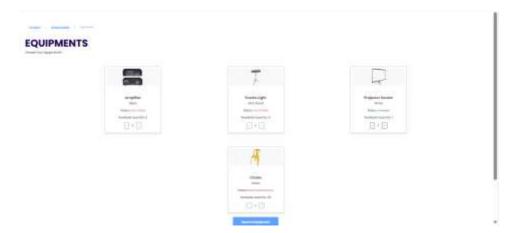
User – View facilities schedules Interface

The 'View Facilities Schedules' page appears when the user selects the 'View Schedule' option. It simply displays a button labeled 'Select Time,' which allows the user to see the details for that specific day. The process then follows the same steps as in the previous figures, guiding the user through selecting available times and confirming the reservation details.

Figure 89

User – Equipment List Interface

This figure shows the page where users can see a list of the equipment available at Barangay Western Bicutan. For each equipment item, the page displayed the number of available units and their status, such as whether it was "Available," "Out of Stock," and "Under Maintenance."

Figure 90

User – Equipment List Double Booking Interface
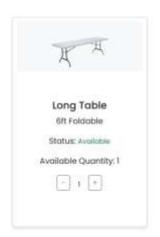
Figure 90 shows that users could also book more than one type of equipment in a single reservation. For example, they could book both a projector and a table at the same time in one booking."
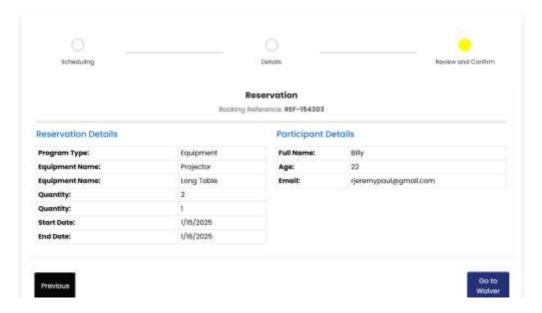
Figure 91

User – No more quantities available notice

This figure shows the notice that appeared when the user attempted to reserve more equipment than was available. For example, if only two chairs were available but the user tried to book 5, the notice would display: 'You cannot reserve more than the available quantity of this equipment.
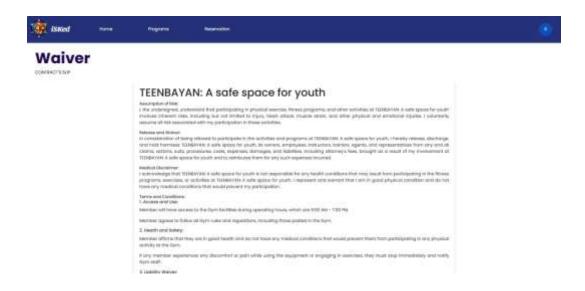
Figure 92

User – Equipment reviews and Confirm Interface

This figure shows the review and confirmation page for equipment reservations. Unlike the facility reservation, this page displayed the name and quantity of each piece of equipment the user was reserving. It allowed the user to review the details of their equipment reservation before finalizing it.

Figure 93

User – Waiver Interface
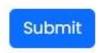
Next is the waiver, where the user needs to read the terms and conditions of the program they are booking. This ensures that the user is fully informed and aware of the guidelines and policies before proceeding with the booking.

Figure 94

User – Agree to Terms and Conditions
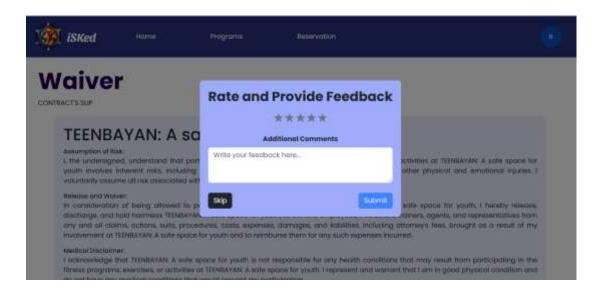
After reading and understanding the waiver, the user can check the checkbox to indicate their agreement with the terms and conditions. Once the checkbox is selected, the user can click the 'Submit' button to proceed, and their booking will be completed.

Figure 95
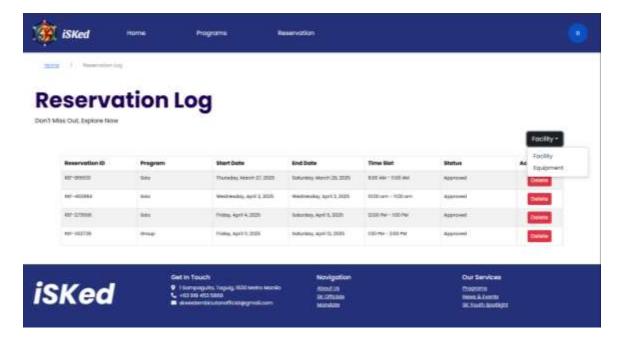
User – Rate and Feedback Interface

The figure above shows a rate and feedback that will appear on every successful reservation. This was designed not to be a compulsory one, so the user or the youth could rate their experience using a 5-star system, with five stars being the highest rating. The feedback area also allowed users to provide their comments.
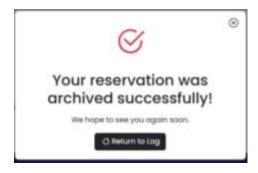
Figure 96

User – Reservation Log Interface

This figure displays the user's history of reservations. It provided a detailed record of past equipment reservations, including the reservation date, time, and equipment details. This allowed users to review their previous bookings and track their usage history effectively.

Figure 97

User – Archived Reservation

This figure shows the reservation log for the user's equipment bookings. It displayed the history of their reservations, including the reservation ID, equipment name, start and end dates, status, and actions. The user could delete any reservation from their log by clicking the 'Delete' button next to the item they wanted to remove. This figure displayed the confirmation message that appeared after the user successfully deleted a reservation from their log. It confirmed that the reservation had been removed.

Figure 98

User – Reservation Log Option
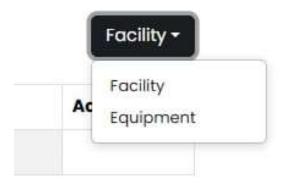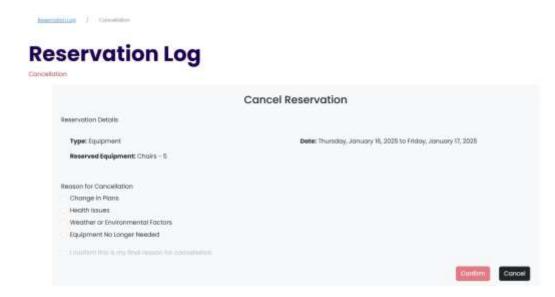
Figure 98 shows that in the reservation log, the user could choose which type of reservation log they wanted to view, whether it was for facilities or equipment. The facility log was similar to the equipment log shown in Figure #, displaying the history of the user's reservations with details such as reservation ID, facility name, start and end dates, status, and action options.

Figure 99
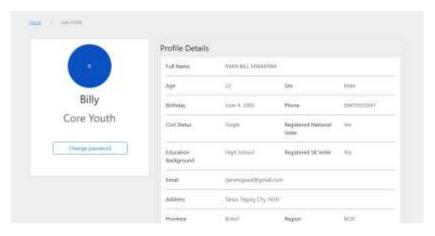
User – Reservation Log Cancellation

Under the Reservation Log, there was a cancellation section where the user could cancel their booking. The user was required to provide a reason for the cancellation before clicking the 'Confirm' button to proceed with the cancellation process.

Figure 100
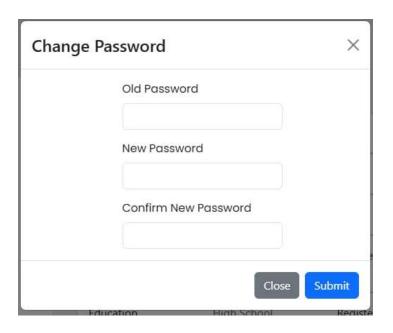
User – Profile Details Interface

In the User Profile details, the user could view their personal information, including their name, age, birthday, civil status, education background, email, province, and whether they were a registered national voter. The user could also change their password by clicking the 'Change Password' button.

Figure 101

User – Change Password

When the user attempted to change their password, the system required them to input their old password first. After providing the old password, the user was prompted to enter a new password. To ensure accuracy, the system also asked the user to re-enter the new password in a "Confirm New Password" field to verify that it matched the initially entered new password. Once all the fields were correctly filled out and the passwords matched, the user was able to click the "Submit" button to complete the process.

Figure 102

User – Password Updated

This figure shows the confirmation message that appeared once the user correctly input their old password and successfully changed it to a new one. It confirmed that the password had been updated successfully.

Figure 103
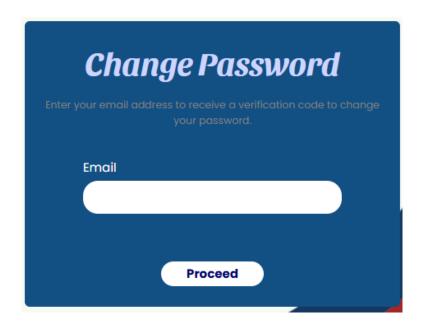
User – Confirm Logout

The 'Confirm Logout' page appeared when the user initiated the logout process. It asked the user to confirm whether they wanted to log out of the system, ensuring that the user intentionally intended to end their session. The user could choose to either confirm the logout or cancel to return to their previous session.

Figure 104
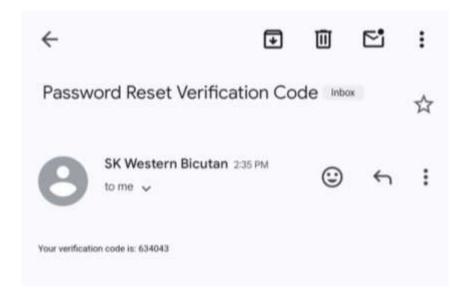
User- Forgot Password

The 'Forgot Password' feature allowed users to recover access to their accounts if they could not remember their password. By clicking the 'Forgot Password' link on the login page, users were prompted to provide their registered email address.

Figure 105

User- Forgot Password Email Code

The Forgot Password Email Code sent a unique code to the user's registered email address when they requested a password reset. The user needed to input this code into the provided field on the system to verify their identity. Once the code was successfully validated, the user could proceed to reset and create a new password securely.

Figure 106

User- Forgot Password (New Password)

This page allows the user to enter the code sent to their registered email to verify their identity. Once the code was entered and validated, the user could create a new password to regain access to their account.

Figure 107
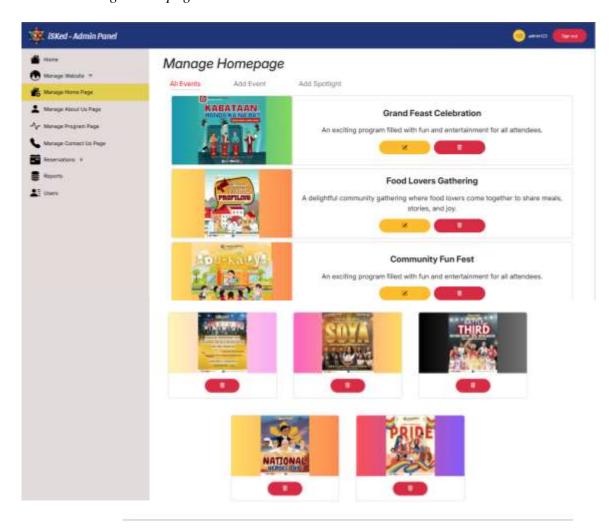
Admin – Homepage / Dashboard Interface

Figure 107 represents the Admin Dashboard, serving as the central hub of the system. It featured an analytics section with a user distribution chart showing active and inactive users, equipment and gym reservation graphs highlighting resource usage trends, and a yearly rating distribution summarizing user satisfaction. These insights helped administrators optimize scheduling, monitor engagement, and improve overall system performance.

Figure 108
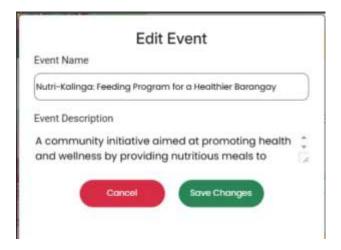
Admin – Manage Homepage All Events

This figure illustrates the All Events section, where administrators could view a comprehensive list of existing events. In this section, they had the option to edit or delete events as needed, providing a streamlined way to manage event listings effectively.

Figure 109
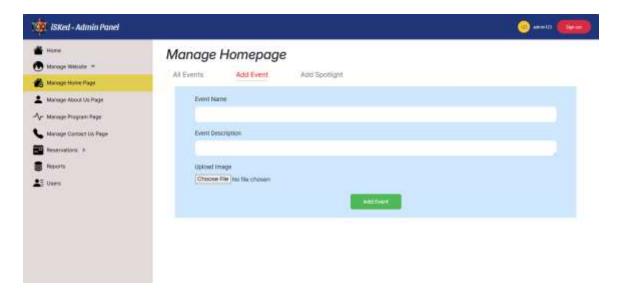
Admin – Manage Homepage Edit Event

This figure shows the editing interface for event details. Administrators could modify event information and either save their changes or cancel the edits. This flexibility ensured that updates could be made efficiently while allowing for corrections if necessary.

Figure 110
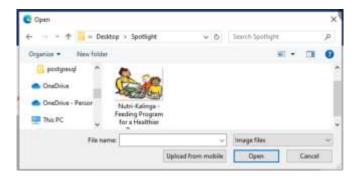
Admin – Manage Homepage Add Event

In the Add Event page, administrators were required to provide details such as the event name and event description and upload an image. To upload an image, the admin selected a file from their device. After entering and reviewing all the information, the admin clicked "Add Event" to save the event.

Figure 111
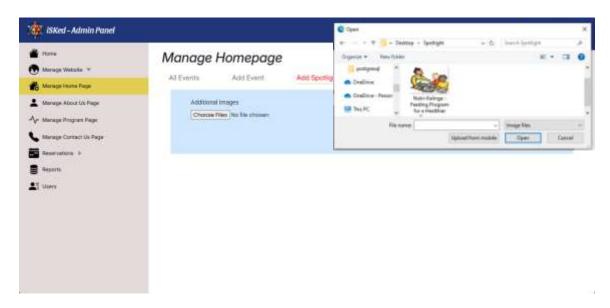
Admin – Manage Homepage Add Photo

This feature allows the admin to upload photos for the event. The admin could browse their device to select the desired file to upload, ensuring flexibility and control over the images displayed on the website.

Figure 112

Admin – Manage Homepage Add Spotlight

In the Add section, the process was similar to the Add Event. Administrators selected an image to upload, and once the details were reviewed, they clicked "Add Spotlight" to save the changes.

Figure 113

Admin – Manage About Us Details Interface
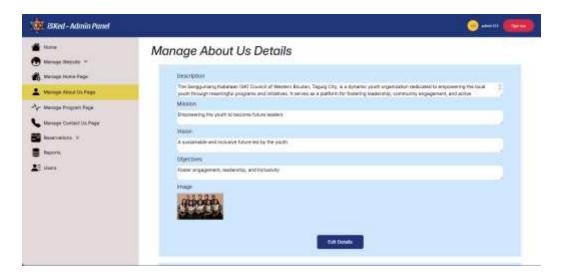
Figure 113 shows the "Manage About Us Details" section of the admin panel. It allowed the administrator to edit the About Us page and update details such as Description, Mandate, Mission, Vision, and Objectives. The administrator could also upload a new image for the About Us section. These details could be edited by clicking "Edit Details," and a new image could be uploaded as well. The "Save Details" button saved the changes.

Figure 114

Admin – SK Council Members Interface
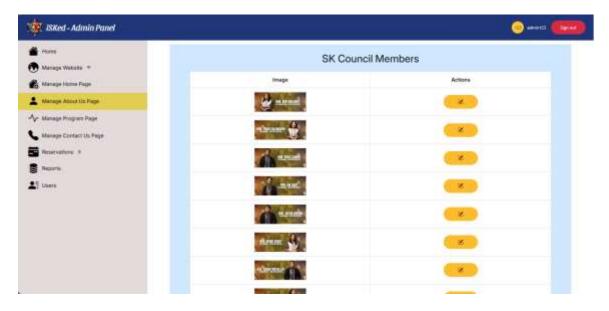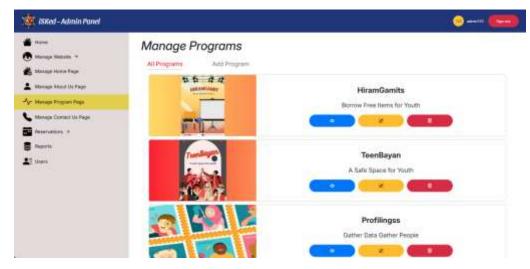
Figure 114 shows the 'SK Council Members' section of the admin panel. It displayed a list of SK council members along with their images. The administrator could edit the details of each member by clicking the 'Edit' button.

Figure 115

Admin – Manage All Program Page Interface

In Figure 115, the 'Manage Programs' section featured the 'All Programs' list, where administrators could see all the available programs. This section provided options to view, edit, or delete any listed program.

Figure 116

Admin – All Program Interface

This figure illustrates the program details displayed when the 'View Detail' button was clicked. It presented the program's description, allowing administrators to review its content. Also, an edit button was displayed for archived data.

Figure 117

Admin – Manage Program Page Edit Program
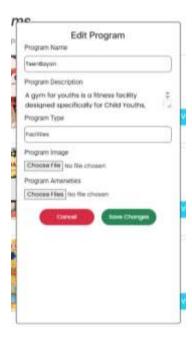
Figure 117 showcases the interface visible when the 'Edit Details' button was selected. Administrators could update the program name, description, image, and amenities through this view. After adjusting, they had the choice to either click 'Cancel' to discard the changes or 'Save Changes' to apply the updates, ensuring precise management of program information.

Figure 118

Admin – Manage Program Page Add Program
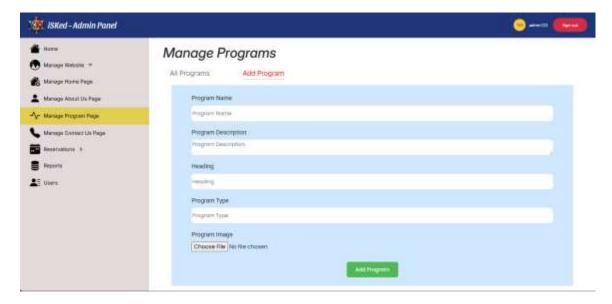
Figure 118 shows the "Add Program" section under the "Manage Program" page. In this section, the admin was required to fill in the program name, description, heading, program type, program image, and amenity image. After entering all the details, the admin could click the "Add Program" button to save the new program.

Figure 119

Admin – Manage Contact Details Edit Details
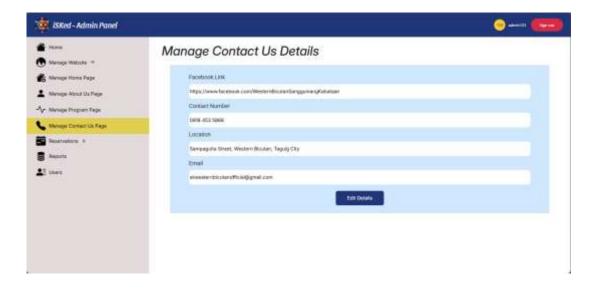
Figure 119 shows the "Manage Contact Us" section, where the admin can edit the contact number, location, and email address.

Figure 120

Admin – Manage Contact Details Save Details

Figure 120 displays the "Save Details" button after the information was filled in.

Figure 121

Admin – Inventory Equipment List Interface
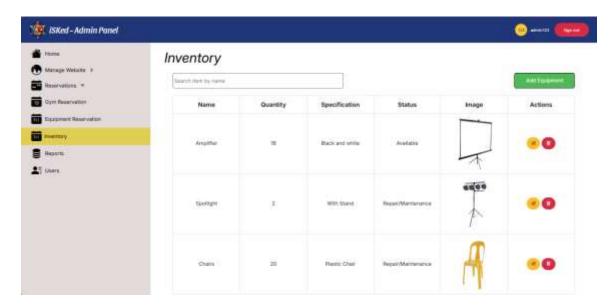
Figure 121 showcases the Inventory page, displaying a list of equipment items available for reservation. Administrators could add new items or edit existing ones using the "Add Equipment" or "Edit" buttons. Each item included details such as name, quantity, specifications, status, image, and available actions.

Figure 122
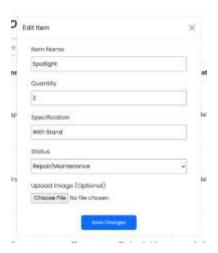
Admin – Add Inventory Equipment

The "Add Item" pop-up appeared when the "Add Equipment" button was clicked. It allowed administrators to add new equipment to the inventory by entering the name, quantity, specifications, and status, and uploading an image. Clicking "Add Item" saved the new entry.

Figure 123
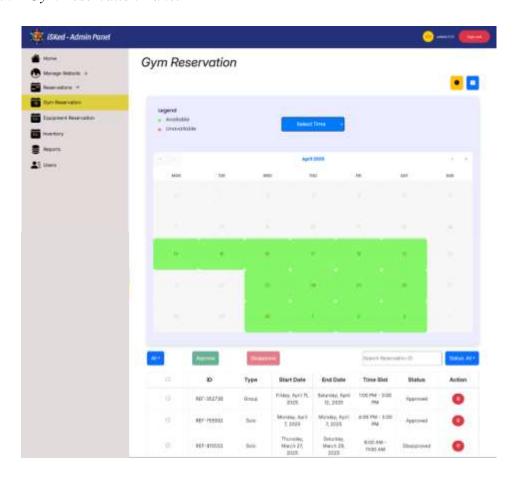
Admin – Inventory Edit Item

When an admin clicks the "Edit" button, a pop-up window appears, allowing modifications to the selected equipment's details, such as name, quantity, specifications, status, and image. Clicking "Save Changes" updated the inventory record.

Figure 124
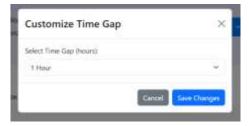
Admin – Gym Reservation Panel

The Admin Gym Reservation panel displayed a list of reservation requests submitted by users. Each reservation entry included the reservation ID, type (Group or Solo), start and end dates, time slot, status (Approved or Disapproved), and a delete action button. The

admin had the option to filter reservations by type or status, search for specific entries using the reservation ID, and perform bulk actions such as approving or disapproving selected reservations. Before the reservations appeared in the table, the admin accessed a calendar interface where available and unavailable dates were visually indicated using a color legend (green for available, red for unavailable). The "Select Time" dropdown allowed the admin to choose a time slot for a specific date. Once selected, the reservation was processed and reflected in the table below for approval or disapproval.

Figure 125

Admin – Gym Reservation Time Customization

The above figure shows where the admin can customize the time gap for reservation.

Figure 126

Admin – Gym Reservation Date Customization
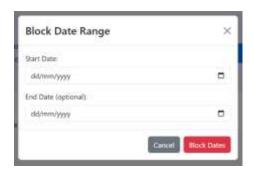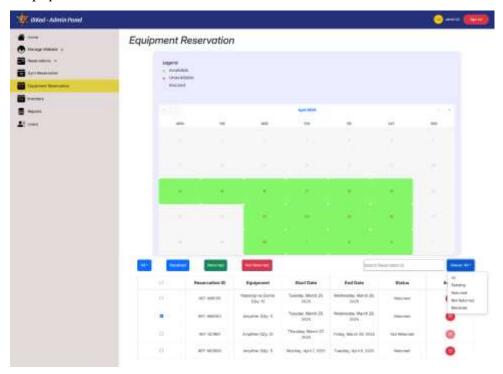
Figure 126 shows where the admin can adjust the block range as a date customization.

Figure 127

Admin – Equipment Reservation Panel

The Admin-Equipment Reservation Panel was designed to allow administrators to manage and monitor equipment reservations efficiently. It featured a calendar view where available dates were highlighted in green, while unavailable and blocked dates were indicated according to the legend. Admins had the ability to filter reservations using quick status buttons such as "All," "Pending," "Received," "Returned," and "Not Returned." A dropdown menu and a search bar were also provided to help locate specific reservation IDs easily. A table below the calendar displayed the reservation details, including the Reservation ID, equipment name and quantity, start and end dates, and the current status.

Admins were able to perform actions such as deleting a reservation using the available action buttons.

Figure 128.A

Admin – Report Interface
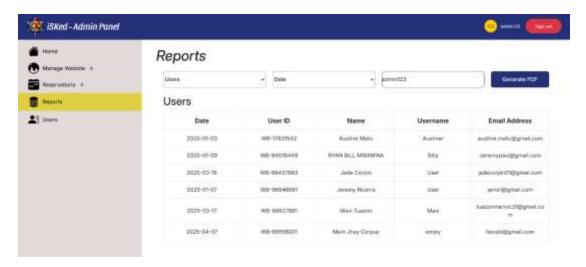
Figure 128. A depicts the Reservation Reports page, where administrators can generate PDF reports on-reservation activities. User, date, and admin name could filter reports. Clicking "Generate PDF" created the report.

Figure 128. B

Admin – Reports Download
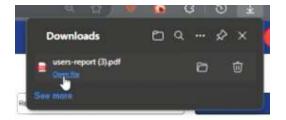
Figure 128. B showss a download notification pop-u, indicatingthat the PDFiss ready for viewing.

Figure 129

Admin – Reports Dropdown Button

This feature allows the admin to select the type of report they want to view from a dropdown menu. The options included reports on users, inventory, equipment, or facilities, providing flexibility and easy access to specific data as needed.

Figure 130

Admin – Reservation Reports PDF

This figure displays a downloaded PDF report from the Reservation Reports page. The report contained user information, including User ID, name, username, and email address. It also indicated who generated or printed the report and included a designated section for a manually written signature over the printed name.

Figure 131

Admin – Equipment Reservations Report

The figure above was the downloaded file report of the reservation of equipment. It displayed the date, user ID, start and end date, the reserved equipment, and its status, such as "Pending," "Not Returned," and "Returned." The printed date and the name of the admin who generated the report were also indicated.

Figure 132

Admin – Facility Reservation Report

Figure 132 contained information about the youth or users who took a slot in the facility reservation. It has six columns, including the user ID, the start and end date for that reservation, the timeslot, and its different statuses: Disapproved and Approved. The name of the admin who generated this report was also displayed along with its print date and a space for their signature.

Figure 133

Admin – Inventory Report

From the equipment reservation, the inventory report was the printed document that summarized the equipment's status, quantity, and item ID.

Figure 134

Admin – Manage Users View Interface

This figure shows the Manage Users page, where a list of users was displayed. The page also featured a dropdown menu, allowing the admin to select and filter users based on youth classifications: Child Youth, Core Youth, and Adult Youth.

Figure 135

Admin – Manage Users View User Details

Figure 135 displays the screen that appeared when the 'View' button was clicked, providing a detailed overview of a user's information, including their unique generated code.

Figure 136

Admin – Manage Users Edit User
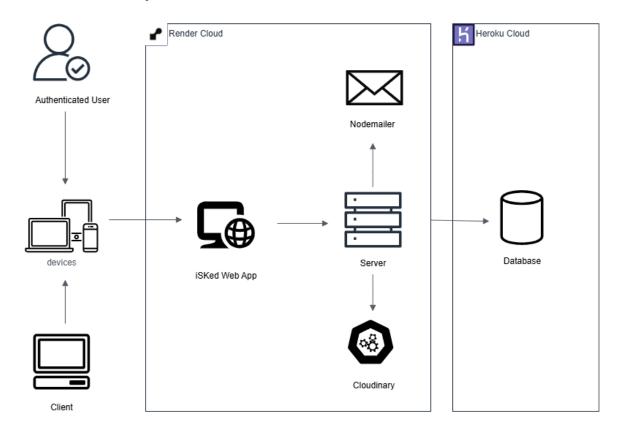
Figure 136 shows the screen for editing a user's details, allowing administrators to modify and update user information as necessary. View, delete, and manage user accounts.

Cloud Architecture

Figure 137

Cloud Architecture for iSKed

The diagram depicts a cloud-based web application architecture for a responsive reservation system, iSKed Web App. It represents the interaction between various components that enable access to the application through the Internet. Users and Clients interact with the system on any device and then navigate to the iSKed Web App, which is hosted on Render.

The iSKed Web App, built with React.js, served as the user interface and handled all user interactions. It communicated with the server hosted on Render. The server, developed with Node.js, processes actions such as logging in, storing, updating, and retrieving

information. It managed the core business logic, processed API requests, and facilitated communication with other components, including the Heroku database.

The server requested APIs such as Nodemailer, which sent emails to users, and Cloudinary, which stored all images, ensuring efficient delivery of these resources to the iSKed Web App. Moreover, the server interacts with a PostgreSQL database hosted on Heroku. This database securely stores all application data, including user information, reservations, and other critical records.

Project Test Results

This section outlines the data collected during the system's testing, focusing on its functionality and reliability. It also presents and analyzes the findings from the project evaluation, providing insights into the system's performance and effectiveness.

Functionality Test Results

This section provides a detailed assessment of the system's features and performance during the testing. It evaluates how well the system's functions align with its intended objectives or purpose and user requirements.

Table 10

Overall Summary of Functionality Test Cases

Use Case	No. of Test Cases
Registered Youth	18
Unregistered User	4
Admin/Client	15
Total	**37**

The functionality of the system was thoroughly tested using various test case involving key user roles. Table 10 provides a summary of the test cases executed for each actor, including Registered Youth, Unregistered Users, and Admin/Client Role. A total of 37 test cases were conducted to ensure the system operates effectively for all user types.

Table 11

Functionality Test Execution Summary

Test Execution	Expected Results	Active Result	
		Cycle 1	Cycle 2
No. of Test Cases Executed	100%	100%	100%

Results of Test Cases

Passed	100%	83.78%	100%
Failed	0%	16.22%	0%
No. of Test Cases Not Executed	0%	0%	0%

The table above summarizes the results of the functionality test execution over two cycles. In cycle 1, 83.78% of the test cases passed, while 16.22% failed. The failed cases were addressed and re-tested in Cycle 2, where 100% of the test cases passed, and no test cases were left unexecuted. This process highlights the effectiveness of the corrective actions taken between cycles to ensure the system's full functionality.

Reliability Test Results

This section evaluates the system's stability and consistency under different conditions, assessing its long-term performance and resilience. It simulates real-world scenarios to identify potential issues that could impact reliability. The evaluation includes data on system uptime, response times, error rates, and recovery from failures. The results demonstrate the system's ability to sustain optimal performance and handle various

stressors, ensuring a seamless user experience. Any identified issues or inconsistencies are documented to guide future enhancements.

Table 12

Reliability Test Execution Summary

Use Case	No. of Test Cases
ISK-REL-001	To verify if the system allows users to successfully perform a standard reservation workflow repeatedly under normal conditions, ensuring that the system can handle multiple requests without errors.
ISK-REL-002	To verify that the system performs efficiently and remains stable under varying levels of user load, such as single-user and multiple users accessing the system simultaneously.
ISK-REL-003	To verify that the system handles errors during reservation operations without crashing and provides error messages to the user.
ISK-REL-004	To verify that the system reliably sends and verifies email addresses without failure and under varying conditions.
ISK-REL-005	To verify that the system remains smooth and responsive during various user interactions, including navigating

	between pages and submitting requests under different load conditions.
ISK-REL-006	To ensure that the system reliably handles conflicts or double bookings, displaying availability status correctly.
ISK-REL-007	To verify that the booking data remains accurate and consistent during and after the booking process.
ISK-REL-008	To verify that the system allows admins to approve or disapprove facility reservations and accurately reflects this in the user interface.
ISK-REL-009	To verify that the system accurately allows the admin to mark equipment reservations as "Returned" or "Not Returned," and reflects this status correctly in the system.
ISK-REL-010	To verify that the system can handle multiple facility and equipment approvals and status without performance degradation or data loss.
ISK-REL-011	To verify the website's responsiveness, ensure it adjusts seamlessly across different screen sizes without loss of functionality or usability.

Table 12 summarizes the reliability tests conducted to asses they system's stability and performance under different conditions. Each test case is designed to evaluate specific aspects of the system's ability to handle various user interactions and system loads without errors. The tests range from ensuring the system's responsiveness and efficiency under different user loads to verifying its ability to handle errors, data consistency, and conflicts.

The goal of these tests is to ensure the system remains stable, efficient, and responsive, providing a seamless user experience even under varying conditions.

Table 13

Reliability Test Execution Summary

Test Execution	Expected Results	Active Result	
		Cycle 1	Cycle 2
No. of Test Cases Executed	100%	100%	100%
Results of Test Cases			
Passed	100%	90.91%	100%
Failed	0%	9.09%	0%
No. of Test Cases Not Executed	0%	0%	0%

Table 13 summarizes the execution results of the reliability tests over two cycles. In Cycle 1, all test cases were executed, but only 90.91% passed, with 9.09% failed. In Cycle

2, after addressing the issues, the system achieved 100% passing for all test cases, with no failures or unexecuted tests.

Project Capabilities and Limitations

The developed system offers the following capabilities:

1. The system allows the user to view and book facility and equipment reservations using an interactive calendar, minimizing manual scheduling conflicts.

2. Enables administrators to add, update, or remove some data such as highlighted news and events, and even their everyday spotlights to cater to community needs.

3. Track usage trends, facility utilization rates, and user behavior, providing insights through detailed analytics and reports.

4. Incorporates approval workflows to ensure all reservations align with SKWB policies and regulations.

5. Generates detailed reports on resource availability, usage trends, and maintenance requirements to assist administrators in strategic decision-making.

6. Implements a unique code-based account registration for youth residents, verified manually via SKWB records, ensuring secure and authentic access.

7. Tracks the availability and maintenance status of equipment to prevent overuse or scheduling conflicts.

8. Provides real-time information on reservation availability, system updates, and announcements, ensuring users stay informed

9. Optimized for desktop and mobile devices, ensuring accessibility for a broader user base.

10. Incorporated a feedback feature, enabling users to provide reviews with their experience in using the website.

11. Facilitated secure password recovery through an email verification process, allowing users to reset their password in case of forgotten credentials.

While the system offers a range of useful features, certain limitations may impact its overall performance and user experience. These limitations include:

1. The registration process requires manual retrieval and verification of the SKWB unique code from the barangay office, which may cause delays for users.

2. The system heavily relies on an internet connection and offers no offline capabilities for reservations or updates.

3. Only youth residents, ages 15 to 30 years old, of Western Bicutan with verified SKWB codes can register, limiting accessibility for external users or visitors.

4. The system does not integrate live support or chat features.

5. The system is tailored to the specific policies of SKWB, and any significant changes to regulations may require reconfiguration or redevelopment.

6. Equipment tracking relies on manual updates, and discrepancies may occur if physical inventory is not synchronized with the system.

7. The approval and return status for the equipment and gym reservation relies on manual reservation of the administration, which may lead to delays, inefficiencies,

or errors in the process. This dependency could also result in miscommunication and inaccurate records, particularly during periods of high demand.

8. Users cannot extensively personalize their profiles or details beyond the default settings provided by the system.

Project Evaluation

The developed web application was demonstrated to three (5) IT professionals, five (5) IT students, nine SK officials (9), and eleven (11) registered youth. A total of thirty (30) respondents from this group of people evaluated the system in terms of functional suitability and reliability in accordance with ISO/IEC 25020, yielding an overall weighted mean rating of (). The overall weighted mean rating falls within the scale range of 3.26-4.0, which is described as Highly Acceptable.

Table 14

Functional Suitability Evaluation Result

Criteria	Weighted Mean	Descriptive Rating
Functional Completeness	3.83	Highly Acceptable
Functional Correctness	3.7	Highly Acceptable

Functional Appropriateness	3.8	Highly Acceptable
Overall Weighted Mean	**3.77**	**Highly Acceptable**

The table shows the results of the Functional Suitability Evaluation based on three criteria: Functional Completeness, Functional Correctness, and Functional Appropriateness. Each criterion was assessed using a Weighted Mean and a Descriptive Rating. The results show that Functional Completeness received a weighted mean of 3.83, which is rated as "Highly Acceptable," meaning the system includes all the necessary functions. Functional Correctness scored 3.7, also rated as "Highly Acceptable," showing that the system provides accurate and reliable outputs.

Functional Appropriateness had a weighted mean of 3.8, which is also "Highly Acceptable," indicating that the system helps users complete their tasks effectively. Overall, the system achieved an Overall Weighted Mean of 3.77, with a "Highly Acceptable" rating, proving that its functionality is well-suited to the users' needs.

Table 15

Reliability Suitability Evaluation Result

Criteria	Weighted Mean	Descriptive Rating
Maturity	3.64	Highly Acceptable

Availability	3.70	Highly Acceptable
Fault Tolerance	3.71	Highly Acceptable
Recoverability	3.73	Highly Acceptable
Overall Weighted Mean	**3.69**	**Highly Acceptable**

The table presents the evaluation result for Reliability Suitability, which determines the dependability of the system based on the extracted response of the respondents. The aspects are divided into four concepts: Maturity, Availability, Fault Tolerance, and Recoverability. The calculated weighted mean and Descriptive Ratings assess these concepts. These ratings are measured based on the weighted mean calculated. Not Acceptable is the lowest value, with a scale of 1.00 - 1.75. Fairly Acceptable scale of 1.76 - 2.50, Very Acceptable scale of 2.51 - 3.25, and the highest, Highly Acceptable with a scale of 3.26 - 4.0; Maturity scored of 3.64 with a Descriptive Rating of Highly-Acceptable. Availability reached the score of 3.70 with the same Descriptive Rating of Highly-Acceptable, Fault Tolerance earned 3.71, denoted also as Highly-Acceptable, and lastly is Recoverability which earned a weighted mean of 3.73, the same Descriptive Rating as the rest of the critical factors. These Results of the weighted mean are extracted as a general weighted mean, computing the average weighted mean and denoted as 3.69, Highly Acceptable.

Table 16

Software Evaluation Results Summary

Criteria	Weighted Mean	Descriptive Rating
Functionality	3.77	Highly Acceptable
Reliability	3.96	Highly Acceptable
Overall Weighted Mean	**3.73**	**Highly Acceptable**

The Software Evaluation Results Summary presents the evaluation based on the criteria of Functionality and Reliability. The overall evaluation of the system indicates that it performs at a high level of acceptance in both aspects. It highlights that the system achieves a "Highly Acceptable" rating for both Functionality (3.77) and Reliability (3.69), with an overall weighted mean of 3.73 and a final descriptive rating of "Highly Acceptable". These results demonstrate the system's ability to effectively meet user needs while maintaining stability, consistency, and robust performance under various conditions.

Chapter 5

SUMMARY OF FINDINGS, CONCLUSIONS AND RECOMMENDATIONS

This chapter provides the summary of findings, conclusions, and the corresponding recommendations drawn from the outcomes of the system's testing and evaluation.

Summary of Findings

The study successfully developed the "iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan," which automates the manual reservation process for the Sangguniang Kabataan (SK) programs. The system provides a responsive online platform that streamlines operations, improving both user experience and administrative efficiency. The system offers a dynamic service creation module, allowing the addition and updating of services and community programs. It automates gym scheduling, optimizes the allocation of facilities and equipment, and displays appointment calendars for better organization.

The evaluation of the study is conducted to assess its acceptability and performance in terms of Functionality and Reliability. The evaluation follows the ISO/IEC standard for software quality, with 30 respondents, including youth participants, SK members, IT professionals, and IT students. These respondents, selected through stratified random sampling, have varying levels of technical proficiency and are asked to evaluate the system using surveys based on a 4-point Likert scale.

For the Functionality test, 37 test cases are performed based on different user roles, including registered youth, unregistered users, and admin/clients. In Cycle 1, 83.78% of the test cases pass, while 16.22% fail. After addressing the issues from Cycle 1, Cycle 2 achieved a perfect score, with 100% of the test cases passing. This indicates that the system's core functions, such as user registration, login, and reservation processes, operate as intended for all user types.

Regarding Reliability, the system is tested under varying conditions, including user load, error handling, and data consistency. In Cycle 1, 90.91% of the reliability test cases pass, with 9.09% failing. After improvements are made, Cycle 2 results in 100% of the test cases passing, confirming the system's ability to handle various scenarios without errors or performance issues.

In terms of evaluation results, the functionality of the system receives a weighted mean of 3.77, rated as "**Highly Acceptable**" for Functional Completeness, Functional Correctness, and Functional Appropriateness. Meanwhile, the system's Reliability is rated with a weighted mean of 3.69, which is also "**Highly Acceptable**" for Maturity, Availability, Fault Tolerance, and Recoverability. Overall, these results indicate that the system is both functionally efficient and reliable, making it a valuable tool for managing reservations for the Sangguniang Kabataan (SK) of Western Bicutan, Taguig City. The system provides a seamless and effective solution for facility and equipment reservations, ensuring stability, efficiency, and ease of use for all stakeholders involved.

Conclusions

The following are the conclusions drawn from the findings of the study:

1. The "iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan" is successful, as it:

   a. Provides a dynamic service creation module, allowing the addition and updating of services and community programs.

   b. Automates gym scheduling and optimizes the allocation of facilities and equipment by displaying appointment calendars.

   c. Tracks facility and equipment usage, service popularity, reservation trends, user satisfaction, and no-show rates through analytics tools.

   d. Integrates approval processes to ensure compliance with SK's policies for reservations.

   e. Assesses the system's impact on resource management and community event organization.

   f. Generates reports for resource forecasting and maintenance planning.

   g. Creates unique codes for youth registration, ensuring secure access with manual verification from the barangay office.

   h. Implements an inventory system to track the availability and maintenance of equipment.

   i. Integrates a feedback system within the user dashboard to improve service delivery and align with government standards.

j. Allows secure password management through email verification, enabling users to update their credentials.

2. The iSKed system is successfully developed using (a) Figma for user interface design, (b) GitHub for version control and collaboration, (c) ReactJs, CSS, and Bootstrap for front-end development, (d) ExpressJs, JavaScript, NodeJS, and Postman for back-end development, (e) PostgreSQL as the database management system, (f) Visual Studio Code as the integrated development environment (IDE), and (g) Heroku for hosting the application.

3. The iSKed system is test and performs well in terms of its functionality and reliability.

4. The system is evaluated as Highly Acceptable in terms of functionality and reliability using ISO/IEC 25010.

Recommendations

To further optimize its implementation and support future improvements, the following recommendations are hereby presented based on the work accomplished during this project and the conclusions drawn:

1. In the group reservation section, a verification feature should be incorporated that cross-references group members with the youth registry, ensuring all participants are registered youth members. Implement a filter to easily select eligible members, and automatic alerts should be included for any discrepancies in member details.

2. Integrate email notification features to alert youth or users about successful reservations, updates on group member verification, and reminders for upcoming bookings, ensuring that users will be kept informed throughout the reservation process.

3. Introduce a more advanced search feature that allows users to filter and sort available facilities based on their preferences.

4. Enable a direct messaging system that allows users to communicate with administrators or other group members regarding their reservations, clarifications, or any issues related to the facility or equipment. This feature would streamline communication and foster better coordination within the system.

5. Incorporate a Calendar API that automatically syncs Philippine public holidays into the reservation system. This feature will help users identify unavailable dates due to holidays and assist administrators in planning facility maintenance or special events around these dates, ensuring a seamless booking experience.

6. Introduce reservation editing features allowing users to modify their reservations, such as adjusting the date, time, or number of attendees, before a specified deadline. This feature would provide flexibility and improve user satisfaction by accommodating changes without requiring cancellation and rebooking.

7. Develop a feature that allows administrators to view and manage user feedback. This functionality promotes transparency, strengthens accountability, and offers valuable insights to support the ongoing enhancement of the system and its services.

8. Implement an automatic delete management wherein it automatically deletes uploaded spotlight, images, or highlights every month to ensure the section remains updated and relevant. Utilize APIs such as tasks scheduling API Google Cloud Tasks to automate this process efficiently while reducing manual intervention.

REFERENCES

A study on user interface and user experience designs. (n.d.). https://www.wjrr.org/a-study-on-user-interface-and-user-experience-designs-and-its-tools

Acharya, K., (2024). Online Bus reservation system project report. Retrieved from: https://d197for5662m48.cloudfront.net/documents/publicationstatus/200624/preprint_pdf/bb22bcb956a6bbca763f738043f26cc3.pdf

Alao, O. D., Priscilla, E. A., Amanze, R. C., Kuyoro, S. O., & Adebayo, A. O. (2022). User-Centered/User Experience UC/UX Design Thinking Approach for designing a university Information Management system. Ingénierie Des Systèmes D'information/Ingénierie Des Systèmes D'Information, 27(4), 577–590. https://doi.org/10.18280/isi.270407

Aldisa, R., T., (2024). *Hotel Reservation System Using Website-Based Rapid Application Development (RAD) Method.* "International Journal of Society Reviews (INJOSER) Vol. 2No. 3, ". Retrieved from: https://injoser.joln.org/index.php/123/article/view/111/139

B. Venkat, S. Indla, Y. Puranik, P. G. (2021). "Review on React JS," vol. 5, no. 4, pp. 1137–1139

Bajaj, D., Bharti, U., Goel, A., & Gupta, S. (2020). PaaS providers and their offerings. International Journal of Scientific & Technology Research, 9(01), 4009-4017.

Baker, M., & Hsu, C. (2021). The Role of Technology in Modern Hospitality Management. Journal of Hospitality and Tourism Research, 45(2), 224-240.

Balilia, C. J., Pagulayan, L., Acoba, J. A., & Mebaña, J. (2023). Information Management System with Project Monitoring For Barangay. *Journal of Engineering, Architecture, and Informatics (JEAI)*, *3*. http://urdc.usl.edu.ph/journals/jeai/papers/vol3/vol%203%20series%20 2023-51-60.pdf

Bargaza, P.C,A., German, J., Binoya,G., Bucao, S., Ibe, Cyrine.,Yap,Dave. (March,2020). eReserba Cardinal: An Intergrated Roo Reservation System for Higher Education Institutions. https://dl.acm.org/doi/10.1145/3664968.3664982

Basumatary, B., & Agnihotri, N. (2022, May 30). Benefits and challenges of using NodeJS. https://acspublisher.com/journals/index.php/ijircst/article/view/10433

Beckman, M., D., Mine, C., K., R., Horton, N., J., Rundel, C., W., Sullivan, A., J., Tacket, M. (2021). *Implementing Version Control With Git and GitHub as a Learning Objective in Statistics and Data Science Courses*. "Journal of Statistics and Data

Science   Education,   Volume   29.”   Retrieved   from: https://www.tandfonline.com/doi/full/10.1080/10691898.2020.1848485 Bharadwaj, A., Kaur, R., & Gupta, R. (2020). Impact of Appointment Scheduling Systems on User Satisfaction and Administrative Efficiency. Journal of Service Research, 23(3), 345-360. doi:10.1177/1094670520910391

Bharadwaj, A., Kaur, R., & Gupta, R. (2023). Managerial Challenges in Implementing Online Reservation Systems in Small Businesses: A Qualitative Study. International   Journal   of   Hospitality   Management,   101,   102143. doi:10.1016/j.ijhm.2022.102143

Blancaflor, E. B., Alvarez, D. a. C., Magno, F. D. C., Purificacion, P. M. G., Requiño, A. J. M., & Yao, J. J. M. (2024). *A Risk Assessment of a Barangay Management System Portal: A Case Study*. https://doi.org/10.1109/icctech61708.2024.00015

BOOKAZOR - an online appointment booking system. (2019, March 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/8899460

Britton,   J.   (2021,   May   6).   What   is   ISO   25010?   *Perforce   Software*. https://www.perforce.com/blog/qac/what-is-iso-25010

Castillo, Mae A.Z, Cruz, Dylan C., Lapaz, Joshuah L., Vallespin, Mc R., Marcha, Karylle G., Padillo, John   V.P., Vegiga, Frinzylle. (2024). *Assessing Demographics, Perceptions, and Participation: A Study on Sangguniang Kabataan (SK) Effectiveness   and   Improvement   Suggestions.* https://www.researchgate.net/publication/380404326_Assessing_Demographics_Pe rceptions_and_Participation_A_Study_on_Sangguniang_Kabataan_SK_Effectiven ess_and_Improvement_Suggestions

Codecademy.   (2024).   *What   is   Express.js?*   Codecademy. https://www.codecademy.com/article/what-is-express-js

DBeaver.   (n.d.).   About   DBeaver.   -   Ian.   (2018,   March   15).   Retrieved   from https://dbeaver.io/about/

Deepa, N., Prabadevi., B., Kritikha, L., & Deepa., B. (2020). *An Analysis on Version Control Systems. 2020 International Conference on Emerging Trends in Information Technology and Engineering (c-ETITE).* IEEE. Retrieved from: https://doi.org/10.1109/ic-ETITE47903.2020.9077781

Demir, D. (2024, November 1). *What is Postman (A Tutorial for Beginners)*. Apidog Blog. https://apidog.com/blog/what-is-postman/

Designing and Prototyping Interfaces with Figma. (n.d.). Google Books. https://books.google.com.ph/books?hl=en&lr=&id=GOBeEAAAQBAJ&oi=fnd&pg=PP1&dq=figma+design+tool&ots=el8vnO6V6N&sig=nkDuF59b915J78Awi7KC4uAv-_Q&redir_esc=y#v=onepage&q&f=false

Deploying with Render | Codecademy. (2025). Codecademy. *https://www.codecademy.com/learn/deploying-with-render*

Eunifridah, P. N. &. M. M. &. S. L. &. W. H. &. S. (2021). Challenges in the investment of ICT infrastructure at secondary school in Zambia, in promoting quality education delivery. ideas.repec.org. https://ideas.repec.org/a/bcp/journl/v5y2021i7p269-279.html

Express/Node introduction - Learn web development | MDN. (2024, July 25). MDN Web Docs. https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/Introduction#introducing_express

Faridah et al. (2023). Social media marketing practices: Opportunities and challenges for MSMEs. ResearchGate. https://www.researchgate.net/publication/375882744_Social_media_marketing_practices_opportunities_and_challenges_for_MSMEs

Fariz, M.S.L., Anggraini, D. (2022). Modern Front End Web Architectures with React.JS

Flores, L., Mendoza, R., Yap, J. (2021). Advancing Youth Governance in the Philippines: A Narrative of the Sangguniang Kabataan and its Road to Reform

GeeksforGeeks. (2024, July 15). Express.js tutorial. GeeksforGeeks. https://www.geeksforgeeks.org/express-js/

Ghansham Jadhav, & Flavia Gonsalves. (2020, June 6). Role of Node.js in Modern Web Application Development. IRJET- International Research Journal of Engineering and Technology. https://www.irjet.net/archives/V7/i6/IRJET-V7I61149.pdf

Gillis, A. S. (2023, March 6). SMTP (Simple Mail Transfer Protocol). WhatIs. https://www.techtarget.com/whatis/definition/SMTP-Simple-Mail-Transfer-Protocol

GMA Integrated News. (2023, October 21). What are the duties and responsibilities of your Sangguniang Kabataan leaders? GMA News Online. https://www.gmanetwork.com/news/topstories/specialreports/885854/what-are-the-duties-and-responsibilities-of-your-sangguniang-kabataan-leaders/story/#:~:text=The%20Sangguniang%20Kabataan%20is%20the,their%20skills%20in%20public%20service.

Guan, Y., Wu, B., & Jia, J. (2020). Does online ticket booking system make people better off? An empirical study on railway service. Transportation Research. Part F, Traffic Psychology and Behaviour, 73, 143–154. https://doi.org/10.1016/j.trf.2020.03.014

Habibi, M. R. M., Mohammadabadi, F., Tabesh, H., Vakili-Arki, H., Abu-Hanna, A., & Eslami, S. (2019). Effect of an Online Appointment Scheduling System on Evaluation Metrics of Outpatient Scheduling System: a before-after MulticenterStudy. Journal of Medical Systems, 43(8). https://doi.org/10.1007/s10916-019-1383-5

Harshwardhan, & Divekar. (2019). Secure Email Software using e-SMTP. International Research Journal of Engineering and Technology (IRJET), 06(03), 2395–0056. https://d1wqtxts1xzle7.cloudfront.net/60555053/IRJET

Heller, M. (2022, July 8). What is Visual Studio Code? Microsoft's extensible code editor. InfoWorld. https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html

Himo, A. J., & Gail, A. (2022). Community Request Queue Management System for Local Government Unit of Cabanatuan City. www.academia.edu. https://www.academia.edu/93028450/Community_Request_Queue_Management _System_for_Local_Government_Unit_of_Cabanatuan_City

Heroku vs AppEngine | UpGuard. (2024). https://www.upguard.com/blog/heroku-appengine

Howell, B. F., Hemming, A. L., Kilbourn-Barber, G., & Christensen, S. Y. (2023). EXPLORING THE IMPACT OF LINEAR & NON-LINEAR PRESENTATION METHODS IN A DESIGN HISTORY COURSE. The Design Society. https://doi.org/10.35199/epde.2023.97

IJRASET Publication. (2022). API testing using Postman tool. www.academia.edu. https://www.academia.edu/92861694/API_Testing_Usi ng_Postman_Tool

IMPLEMENTATION OF BARANGAY MANAGEMENT SYSTEM: AN EXTENSION SERVICE OF CVSU-TANZA CAMPUS – International Journal of Research in Education Humanities and Commerce. (n.d.). https://ijrehc.com/vol-4-issue-04/implementation-of-barangay-management-system-an-extension-service-of-cvsu-tanza-campus/

Jagadeeshwaran, M., Jahnavi, D., & Sowjanya Lakshmi, A. (2023). ENHANCING IMAGE TRANSFER SECURITY IN FTP AND SMTP PROTOCOLS. International Research Journal of Modernization in Engineering Technology and Science, 05(08), 2582–5208. https://doi.org/10.56726/IRJMETS43980

Kramer, N. (2024, May 21). Visual Studio Code: Read this before you get started. Daily.dev. https://daily.dev/blog/visual-studio-code-read-this-before-you-get-started

Koh, W. S., & Mohmad Hassim, Y. M. (2021). Hotel Reservation Management System. *Applied Information Technology And Computer Science*, *2*(2), 973-992. https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/5047

Kumar, A., Singh, A., & Gupta, R. (2021). User Experience in Online Appointment Systems: Challenges and Solutions. Journal of Business Research, 130, 735-743. doi:10.1016/j.jbusres.2021.03.038

Kuparinen, S. (2023). Improving Web Performance by Optimizing Cascading Style Sheets (CSS): Literature Review and Empirical Findings. From https://helda.helsinki.fi/server/api/core/bitstreams/694695cf-1bdf-4432-b255-f68d6bdb4b76/content

Learning PostgreSQL 11. (2019). Google Books. https://books.google.com.ph/books?hl=en&lr=&id=ZtOGDwAAQBAJ&oi=fnd&pg=PP1&dq=postgresql+introduction&ots=GGXGLh4fOE&sig=z-UPVEc5H6QAIIo7wtg7WBTYosM&redir_esc=y#v=onepage&q=postgresql%20introduction&f=false

Lee, S. M., & Lee, H. (2023). User Experience and Administrative Burdens in Manual Appointment Systems: A Study on Healthcare Services. International Journal of Health Care Quality Assurance, 36(1), 12-25. doi:10.1108/IJHCQA-05-2022-0147

Li, M. (2019, May 1). The implementation and evaluation of a simple online reservation system for local sports game meet-ups. Carolina Digital Repository. https://cdr.lib.unc.edu/concern/masters_papers/db78tg78v

Loeliger, J., & Ponuthorai, P.,K. (2022). *Version Control with Git.* "O'Reilly Media, Inc."Retrieved From https://books.google.com.ph/books/about/Version_Control_with_Git.html?id=4v8tzwEACAAJ&redir_esc=y

Marinova, K. (2023). KEY DIMENSIONS OF WEB-BASED RESERVATION SYSTEMS IN MODERN TOURISM SECTOR. ResearchGate. https://doi.org/10.5281/zenodo.10400242

Marksman Drysdale Travel: Corporate Travel and Tours | Travel Agency Manila. (n.d). Marksman-Drysdale Travel Inc. | Travel and Tours | Corporate Travel Agency. Retrieved April 6, 2023, from https://marsmantravel.com.ph/

Medical guide system. (2022).
https://www.indianjournals.com/ijor.aspx?target=ijor:ijemr&volume=12&issue=6
&article=025

Murphy, J. (2020). Understanding Reservation Systems: A Comprehensive Guide for
Industry Professionals. Travel and Tourism Management Journal, 12(1), 45-60.

National   Youth   Commission   (n.d).   Sangguniang   Kabataan.
https://nyc.gov.ph/sangguniangkabataan/?fbclid=IwY2xjawFdo1ZleHRuA2FlbQIxM
AABHWJKGKidTd6k9yPYHs2CqhfCTGVvf4CBUlwIqyojlPd9VEZmqL5Q1g3Teg_
aem_T_mXABlg20v-tD3XDO6VYA

Ndia et al., (2019). On the analysis of cascading style sheets. ResearchGate.
https://www.researchgate.net/publication/254008906_On_the_Analysis_of_Casca
ding_Style_Sheets

Neupane, S. (2020). Developing a static website and deploying it to Heroku. Theseus.
https://www.theseus.fi/handle/10024/349682

Nguyen, T., Tran, T., & Pham, H. (2021). The Role of Automated Systems in Reducing
Operational Challenges in Appointment Management. Journal of Business Research,
134, 191-199. doi:10.1016/j.jbusres.2021.06.067

Niemi,   L.   (2020).   Cloud   Solutions   in   Non-software   Company.
https://www.theseus.fi/bitstream/handle/10024/334658/Niemi_Lauri.pdf?sequence=2

Olle Lundmark. (2022, May). Responsive Web Design for Modern Devices.
https://www.diva-portal.org/smash/get/diva2:1673527/FULLTEXT01.pdf

Omnify (2020). *What is an Online Reservation System and why do you need it*. Retrieved
from: https://www.getomnify.com/blog/one-reservation-system

Oscar, O., Kazaara, A. I., Micheal, T., & Kaazara, A. G. (2023). Impacts of farmer's
cooperative societies in marketing of agricultural produce. A case study of Loro
Sub. . . ResearchGate. https://www.researchgate.net/publication/371751953

Ouellette, A. (2023, February 2). What is Bootstrap? An Awesome Beginner's Guide.
CareerFoundry.   https://careerfoundry.com/en/blog/web-development/what-is-
bootstrap-a-beginners-
guide/#:~:text=Bootstrap%20is%20a%20giant%20collection,quickly%20build%2
0fully%20responsive%20websites.

Palangdao, C. D. S., Dumpayan, H. M. C., Perez, G. K., Aquino, A. S. L., & Ramos, C. A.
(2023). YOUTH GOVERNANCE IN THE SELECTED BARANGAYS OF
BAGUIO CITY. *Diversitas Journal, 8*(4). https://doi.org/10.48017/dj.v8i4.2674

Pedamkar, P. (2023, October 17). Visual Studio Code. *EDUCBA*. https://www.educba.com/what-is-visual-studio-code/

Peek, S. (2023, December 20). *What Is Agile Scrum Methodology?* Business News Daily. https://www.businessnewsdaily.com/4987-what-is-agile-scrum-methodology.html

Perdana., R., D., A., K., & Sancoko, S., D., (2023). *Android-Based Reservation System at Dharmarini Health Center.* "International Journal of Computer Applications (0975 – 8887) Volume 185 – No. 43." Retrieved from: https://www.researchgate.net/profile/Sulistyo-Dwi-Sancoko/publication/375850476_Androidbased_Application_of_Reservation_System_at_Dharmarini_Health_Center/links/658af3732468df72d3db2755/Android-based-Application-of-Reservation-System-at-Dharmarini-Health-Center.pdf

Permatasari, H., Purwanto, E., & Triyono, T. (2023). ANALYSIS OF THE ROLE OF RESERVATION SYSTEMS IN IMPROVING CUSTOMER EXPERIENCE IN THE SALON AND SPA INDUSTRY: LITERATURE REVIEW. Proceeding of International Conference on Science, Health, and Technology, 220–228. https://doi.org/10.47701/icohetech.v4i1.3395

PostgreSQL: about. (2024). The PostgreSQL Global Development Group. https://www.postgresql.org/about/

Ramiscal, J. D. (2023, September 6). Focus on education, not just on sports. Inquirer Opinion. https://opinion.inquirer.net/168421/focus-on-education-not-just-on-sports

Rawat, P., Mahamajan, S. N. (2020). ReactJS: A Modern Web Development Framework. At https://ijisrt.com/assets/upload/files/IJISRT20NOV485

Render. (2024). Cloud application platform. Retrieved from https://render.com/

Rosales, Renniel Jayson Jacinto (2023). Assessing Sangguniang Kabataan Programs: A Baseline Study in Developing a Comprehensive Plan for the Development of the Youth in Cuenca, Batangas. Universal Journal of Educational Research 2 (1):47-56.

Schofield, D. (2021). Responsive Redesign and its Effects on Perceived Usefulness. Suny. https://www.academia.edu/45285422/Responsive_Redesign_and_its_Effects_on_Perceived_Usefulness?rhid=28523159139&swp=rr-rw-wc-36339757

Shahu Gaikwad, S. S. G., & Adkar, P. A. (2019). A Review Paper On Bootstrap Framework. *A Review Paper on Bootstrap Framework*, 2(2456–8880), 349–351. https://www.irejournals.com/paper-details/1701173

Shakira, M., & Devis, A. (2024). The Impact Of Online Reservation System On The Profitability Of Uganda's' Hotel Industry; The Case Of Source Of The Nile Hotel Jinja. ResearchGate. https://www.researchgate.net/publication/378480821_The_Impact_Of_Online_Reservation_System_On_The_Profitability_Of_Uganda's'_Hotel_Industry_The_Case_Of_Source_Of_The_Nile_Hotel_Jinja#fullTextFileContent

Siddiqui, M., Khan, M., & Ali, A. (2023). Understanding User Frustration in Online Booking Systems: An Empirical Study. Journal of Service Management, 34(1), 45-63. doi:10.1108/JOSM-05-2022-0190

Singh, G., Saboo, R., & Mathur, R. (2024). Responsive Web Design with HTML and CSS. Responsive Web Design With HTML and CSS, 9(1). https://doi.org/10.48047/resmil.v9i1.22

Sousa, M., Goncalves, A. (2020). Humanportal – A react.js case study. From https://ieeexplore.ieee.org/abstract/document/9141070

Stack Overflow Developer Survey 2021. (2021). Stack Overflow. Retrieved from: https://survey.stackoverflow.co/2021#overview

Study of JavaScript static Analysis Tools for vulnerability detection in Node.js packages. (2023, December 1). IEEE Journals & Magazine | IEEE Xplore. https://ieeexplore.ieee.org/document/10168679?denied=

Svensson, O., Kald, M. P. (2021). React Native and Native Application Development. From https://www.diva-portal.org/smash/get/diva2:1547719/FULLTEXT01.pdf

Tomar, R., & Dangi, S. (2021). JavaScript. https://doi.org/10.1201/9781003122364

Tu, Y., C., Terragni V., Tempero, E., Shakil, A., Meads, A., Giacaman, N., Fowler, A., Blincoe, K. (2022) *Github in the classroom: Lesson learnt*. "ACE '22: Proceedings of the 24th Australasian Computing Education Conference February 2022 Pages 163–172." Retrieved from: https://dl.acm.org/doi/abs/10.1145/3511861.3511879

Unicsoft, & Unicssoft_Dev. (2023, June 2). ISO 25010: Enhancing our software quality Management Process. Unicsoft. https://unicsoft.com/blog/iso-25010-enhancing-our-software-quality-management-process/

Usage statistics of JavaScript as client-side programming language on Websites, May 2024. (2024). https://w3techs.com/technologies/details/cp-javascript

What is Dbeaver? Database.Guide. Retrieved from https://database.guide/what-is-dbeaver/

What is an Online Reservation System and why do you need it? (2020b, April 4). Omnify. https://www.getomnify.com/blog/one-reservation-system

Zhao, Y., Chen, L., & Wang, X. (2022). Appointment Booking Systems: Enhancing User Satisfaction through Improved Availability Features. International Journal of Information Management, 66, 102525. doi:10.1016/j.ijinfomgt.2022.102525

Zola, A. (2022, August 3). *Bootstrap*. WhatIs. https://www.techtarget.com/whatis/definition/bootstrap

Appendix A

ISO/IEC 25010 SOFTWARE EVALUATION FORM

iSKed: Development of Facilities and Equipment Reservation Management System for Barangay Western Bicutan

Name: _____ **Age:** _____

Sex: _____ **Devices Used:** _____

Role: *(Please check the appropriate option)*

☐ IT Professional ☐ SK Officials ☐ Youth Participant

☐ IT student

For Youth Participant, please indicate your frequency of SK Program usage:

☐ Frequent ☐ Occasionally

Evaluation Procedure: This evaluation is conducted to assess the iSKed system's compliance with ISO/IEC 25010 software quality standards. Kindly rate the system based on your experience by using the 4-point Likert scale to rate each criterion. Use the following scale to rate each statement:

4 – Highly Acceptable 3 –Very Acceptable 2 – Fairly Acceptable 1 – Not Acceptable

A. Functional Suitability

Evaluation Criteria	4	3	2	1
Functional Completeness *Does the system provide all the required features?*				
Functional Correctness *Are the features working correctly without errors?*				

Functional Appropriateness *Are the features aligned with user needs?*				

B. Reliability

Evaluation Criteria	4	3	2	1
Maturity *How reliably does the system meet your needs?*				
Availability *Does the system remain functional and available when you need it?*				
Fault Tolerance *How well does the system continue to function despite errors?*				
Recoverability *How well does the system recover data after an interruption or failure?*				

Appendix B

SAMPLE ANSWERED EVALUATION SHEET AND RESULTS SHEET

Email *

marivic.tuazon@tup.edu.ph

Do you consent? *

⦿ I have read and understood the conditions under the Data Privacy Act of 2012 and consent to the collection and use of my personal information for this study.

◯ No

PERSONAL INFORMATION

Please make sure that all responses are accurate and filled out correctly.

Full Name *
Format: First Name, Middle Initial, Last Name

Marivic O. Tuazon

Sex *

Female

Age *

22

Devices Used *

Laptop

Role: *(Please check the appropriate option)* *

- ☐ IT Professional
- ☐ SK Officials
- ☐ Youth Participant
- ☑ IT Student

For Youth Participant, please indicate your frequency of SK Program usage: *

- ☐ Frequent
- ☐ Occasionally
- ☑ Not a Youth Participant

Evaluation Procedure

This
evaluation is conducted to assess the iSKed system's compliance with ISO/IEC
25010 software quality standards. Kindly rate the system based on your
experience by using the 4-point Likert scale to rate each criterion. Use the
following scale to rate each statement:

4 – Highly Acceptable
3 –Very Acceptable
2 – Fairly Acceptable
1 – Not Acceptable

A. Functional Suitability
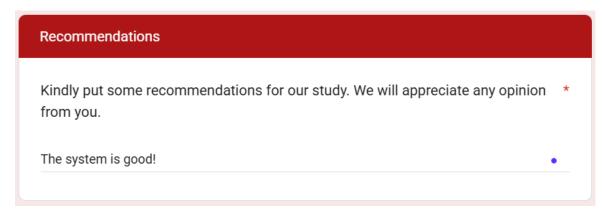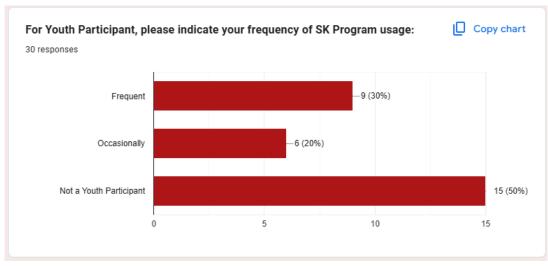
Functional Completeness *

	4 – Highly Acceptable	3 –Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
Does the system provide all the required features?	●	○	○	○

Functional Correctness *

	4 – Highly Acceptable	3 –Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
Are the features working correctly without errors?	○	●	○	○

Functional Appropriateness *

	4 – Highly Acceptable	3 –Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
Are the features aligned with user needs?	●	○	○	○

B. Reliability

Maturity *

	4 – Highly Acceptable	3 –Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
How reliably does the system meet your needs?	●	○	○	○

Availability *

	4 – Highly Acceptable	3 –Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
Does the system remain functional and available when you need it?	○	●	○	○

Fault Tolerance *

	4 – Highly Acceptable	3 –Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
How well does the system continue to function despite errors?	●	○	○	○

Recoverability *

	4 – Highly Acceptable	3 – Very Acceptable	2 – Fairly Acceptable	1 – Not Acceptable
How well does the system recover data after an interruption or failure?	○	●	○	○

Recommendations

Kindly put some recommendations for our study. We will appreciate any opinion *
from you.

The system is good!

Sex

30 responses

Age

30 responses

Devices Used

Copy chart

30 responses

Role: *(Please check the appropriate option)*

Copy chart

30 responses

Role	Count
IT Professional	6 (20%)
SK Officials	8 (26.7%)
Youth Participant	10 (33.3%)
IT Student	6 (20%)

For Youth Participant, please indicate your frequency of SK Program usage:

Copy chart

30 responses

Frequency	Count
Frequent	9 (30%)
Occasionally	6 (20%)
Not a Youth Participant	15 (50%)

Evaluation Procedure

A. Functional Suitability

Functional Completeness

Copy chart

Legend: 4 – Highly Acceptable, 3 – Very Acceptable, 2 – Fairly Acceptable, 1 – Not Acceptable

Does the system provide all the required features?

Functional Correctness

Copy chart

Legend: 4 – Highly Acceptable, 3 – Very Acceptable, 2 – Fairly Acceptable, 1 – Not Acceptable

Are the features working correctly without errors?

Functional Appropriateness

Copy chart

Legend: 4 – Highly Acceptable, 3 – Very Acceptable, 2 – Fairly Acceptable, 1 – Not Acceptable

Are the features aligned with user needs?

B. Reliability

Maturity

How reliably does the system meet your needs?

Availability

Does the system remain functional and available when you need it?

Fault Tolerance

How well does the system continue to function despite errors?

Recoverability

Copy chart

How well does the system recover data after an interruption or failure?

Legend: 4 – Highly Acceptable, 3 – Very Acceptable, 2 – Fairly Acceptable, 1 – Not Acceptable

Recommendations

Kindly put some recommendations for our study. We will appreciate any opinion from you.

30 responses

The design and functions are great specially the features but need to enhance

It met the needs for reserving, just need enhancements.

(Goodluck ninyo, maka graduate ramong tanan mehehehe. Goodluck jadeng, I hartchu)

Please ensure that your iSKed System performs at its best, wherever it's needed.

If you have an onion and cut it, does that make it an opinion?

The iSKed system demonstrates a well-thought-out design, ensuring user-friendly navigation

good job :)

The system is user-friendly and convenient to use.

Love the yugi-oh refrence.

Recommendations

Kindly put some recommendations for our study. We will appreciate any opinion from you.

30 responses

I greatly appreciate the report generation feature. Lalo na ngayon na nag ma-manual lang kami sa report :>

Love the website, it would be better if only it is not focused on two services.

The system is user-friendly and well-designed, with features that are very helpful for the SK. It works well overall, and with a few small improvements, it could be even better. I also appreciate the students' behavior during our meetings they were all kind and professional.

Goodluck Guys!

the system and study is connected to each other. The design is fine.

The system meets our expectations, and several features are working well!
Thank you guys!

The system has potential, was able to choose available time schedule.

Recommendations

Kindly put some recommendations for our study. We will appreciate any opinion from you.

30 responses

It is a good way

Improve the view details UI :>

nice ang galing good job napaka galing nyo salamat sa lahat

Always think on some ways to further improve the whole system.

Hoping for improvement in the future as needed.

It's a great idea to generate unique codes for the youth who want to create accounts, ensuring safety against fake registrations.

the study/system is great but the system is need to enhance

Recommendations

Kindly put some recommendations for our study. We will appreciate any opinion from you.

30 responses

.

Thank you for creating the system for us, the SK officials, and the residents of Western Bicutan.

Aside from star rating, a comments or suggestion feedback would be much better.

Thank you for making a system like this for our community

I like the system it has good functionality overall, though there are some areas that still need improvement.

Hindi masakit sa eyes yung UI, but hoping sa ibang pages magkaron din ng designs. Overall the system is good and makakatulong sa community at makaka-save ng time para sa kania. Goodluck and soon maiimprove nyo pa yan!

Maganda yung system pero may i-improve pa and mga need ayusin like yung mga pop up notification and recom na sana lagyan ng comment box para sa mga user na gagamit

Recommendations

Kindly put some recommendations for our study. We will appreciate any opinion from you.

30 responses

Thank you for making a system like this for our community

I like the system it has good functionality overall, though there are some areas that still need improvement.

Hindi masakit sa eyes yung UI, but hoping sa ibang pages magkaron din ng designs. Overall the system is good and makakatulong sa community at makaka-save ng time para sa kania. Goodluck and soon maiimprove nyo pa yan!

Maganda yung system pero may i-improve pa and mga need ayusin like yung mga pop up notification and recom na sana lagyan ng comment box para sa mga user na gagamit

The iSKed: Development of Facilities and Equipment Reservation Management System is a transformative initiative for Barangay Western Bicutan. This project not only streamlines the reservation process but also promotes transparency, efficiency, and accessibility in managing our resources, and most importantly, to help us provide high quality of service to the youth. We deeply appreciate the system developers for choosing our barangay as their main respondent for this project. I have no recommendations, as the project already aligns perfectly with our goals and expectations. We look forward to seeing the positive impact this project will bring to our barangay's services and operations.

Appendix C

SUMMARY OF RESPONDENT' EVALUATION

Respondents	Functional Suitability			Reliability			
	Functional Completeness	Functional Correctness	Functional Appropriateness	Maturity	Availability	Fault Tolerance	Recoverability
1	4	4	4	4	4	4	4
2	4	4	4	4	4	4	4
3	3	3	3	3	3	3	3
4	4	4	4	3	3	3	3
5	4	4	4	4	4	4	4
6	4	4	4	4	4	4	4
7	3	3	3	4	4	4	3
8	4	4	4	3	3	4	3
9	4	4	4	4	4	4	4
10	3	3	4	3	4	3	3
11	3	2	2	2	2	2	2
12	4	4	4	4	3	3	4
13	4	3	3	3	3	3	3
14	4	4	4	4	4	4	4
15	4	4	4	4	4	4	4
16	4	4	4	4	4	4	4
17	4	4	4	4	4	4	4
18	4	4	4	4	4	4	4
19	4	4	4	4	4	4	4
20	4	4	4	4	4	4	4
21	4	4	4	4	4	4	4
22	4	4	4	4	4	4	4
23	4	4	4	4	4	4	4
24	4	4	4	4	3	3	3
25	4	4	4	3	3	2	3
26	4	3	4	3	4	4	4
27	3	3	4	3	3	4	3
28	3	3	3	3	3	3	3
29	4	4	4	4	4	4	4
30	4	4	4	4	4	4	4
WEIGHTED MEAN	3.83	3.7	3.8	3.64	3.70	3.71	3.73

Appendix D

TEST CASES FOR FUNCTIONALITY

Test Case ID		ISK-FUN-001	UC Reference	User Registration and Verification, Youth User
Objective		This test case is executed to check the functionality of registration, such as validation of SKWB unique code and generation of login credentials.		
Assumptions/Preconditions		1. The user has verified records at the barangay, identified as a resident of Western Bicutan and part of the youth. 2. The user obtained the automatically generated unique code of the system from the barangay.		
Actions		**Expected Result**	**Actual Result**	
1. Enter the unique code to the designated location of SKWB unique code verification, then click the "Validate Code" button. 2. Enter the desired username and password before clicking the "Create Account" button.		1. The youth entered the SKWB unique code to the code validation location, and the system matched the code to the added accounts. 2. The system confirms the newly entered username and password and activates the account.	1. The system correctly matches the entered SKWB unique code and displays the current username and password, which gives the user a chance to be modified before the account activation. 2. The system changed the default username and password into the new username and password and then displayed a success message box.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-FUN-002	UC Reference	Youth User Registration with Invalid Unique Code
Objective		This test case is executed to verify an invalid or missing SKWB unique code during the registration process.		
Assumptions/Preconditions		The user is a youth resident but entered an invalid or incomplete SKWB unique code during registration or account activation.		
Actions		**Expected Result**	**Actual Result**	
1. Enter an invalid, incomplete, or empty SKWB unique code at the entry for verification and account activation. 2. Click the "Validate Code" button.		1. The system displayed an error message pointing to an invalid or incomplete SKWB code. 2. The user cannot proceed to the next step since the entered code is invalid or incomplete.	1. The system displayed an error message indicating an invalid entered code. 2. For the empty submitted code, the system displays an error message box indicating a "Field cannot be blank."	
Status	PASSED	**Severity**	**Priority**	

Test Case ID	ISK-FUN-003	UC Reference	Youth User Login and Authentication
Objective	This test case is executed to verify that the system allows users to log in and authenticate successfully.		

Assumptions/Preconditions	The youth user has a registered account with valid credentials.	
Actions	**Expected Result**	**Actual Result**
1. Navigate to the login page. 2. Enter the username in the username field. 3. Enter the corresponding password in its field. 4. Click the "Sign In" button.	1. The user is redirected to the dashboard or homepage once the entered credentials are validated. 2. The system displays an error message once the If the entered credentials are invalidated.	1. When entered credentials are validated, the user is redirected to their dashboard or homepage. 2. When entered credentials are invalidated, the system displays an error message.
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-004	UC Reference	Forgot Password, Youth User
Objective	This test case is executed to verify that the forgot password functionality works as intended for registered users.		
Assumptions/Preconditions	The user has a registered account in the system, and the system has been set up to send recovery or reset code through email.		
Actions	**Expected Result**	**Actual Result**	

1. On the sign-in page, click on the "Forgot password" link. 2. Enter the email address in the provided field. 3. Click the "Proceed" button to request a password reset. 4. Check the inbox of the provided email address for the password reset email. 5. Enter the new password then click the "Submit" button.	1. The user received a password reset email with a link to change their password. 2. The user should be able to successfully reset their password and log in with the new credentials.	1. Once the entered email address is found on the database, the user must receive a 6-digit code to change their password. 2. The user successfully received a code to change their password. 3. After submitting a new password, a successful message box appeared. 4. The user was able to log in with the new credentials.		
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-005	**UC Reference**	Successful Admin Login
Objective	This test case is executed to check if the admin login functionality works properly to ensure that correct access control was granted and to verify the validation of credentials.		
Assumptions/Preconditions	The credentials of the user registered as an admin at the system.		
Actions	**Expected Result**	**Actual Result**	

1. Navigate to the login page. 2. Enter the required information for logging in at the login page. 3. Click the "Sign in" button.		1. The system validated the entered credentials. 2. Once the credentials are correct, the admin will be granted access to the admin dashboard.		1. The system successfully validated the admin credentials. 2. Upon entering the correct credentials, the admin was granted access to the admin dashboard.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-FUN-006		**UC Reference**	Unsuccessful Admin Login
Objective		This test case is executed to verify the behavior of the system regarding the incorrect credentials provided during login by showing an error message that the user cannot proceed.			
Assumptions/Preconditions		The admin user exists in the system with valid credentials and is connected to a database that stores the admin credentials.			
Actions		**Expected Result**		**Actual Result**	
1. Navigate to the admin login page. 2. Enter the incorrect admin username and password in their corresponding field. 3. Click the "Sign in" button.		1. The system will not validate the entered credentials. 2. The admin is not granted access to the admin dashboard.		1. The system displayed an error message box. 2. Admin was not able to log in to their accounts.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-FUN-007		UC Reference	Unregistered User Registration
Objective		This test case is executed to verify that users who are not logged in are restricted access pages or features of the system.			
Assumptions/Preconditions		1. The user is unregistered and not logged into the system. 2. The system should not allow unregistered users to access features or pages restricted to registered users only.			
Actions		**Expected Result**		**Actual Result**	
1. Click "Home" then navigate to a page that requires user registration or login. 2. Attempt to access the page without being logged in.		1. The system should prompt the unregistered user to either register or login before proceeding to the requested page or feature. 2. Unregistered users should not be allowed to proceed to restricted pages without completing the registration or login process.		1. The system correctly prompted the user to register or login before proceeding. 2. The system correctly did not allow the registered users to proceed without completing the registration process.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-008	UC Reference	Unregistered User Invalid Registration
Objective	This test case is executed to verify that the system prevents registration with an invalid verification code.		
Assumptions/Preconditions	The user is attempting to register using a code that was not given by the SKWB.		
Actions	**Expected Result**	**Actual Result**	
1. Enter different or not existing code on the database. 2. Click the "Submit Code" button.	1. The system displayed an error message indicating that the code was incorrect. 2. The user is stuck with the registration until a valid verification code is entered.	1. The system displayed an error message, preventing the user from proceeding with registration.	
Status	PASSED	**Severity**	**Priority**

Test Case ID	ISK-FUN-009	UC Reference	Unregistered User Access Attempt
Objective	This test case is executed to verify that booking reservations through the system's programs are not allowed for non-registered users.		
Assumptions/Preconditions	The user is attempting to make a reservation without logging in or registering.		
Actions	**Expected Result**	**Actual Result**	
1. Navigate to the reservation page as an unregistered user.	1. The system redirected the unregistered user to	1. The system correctly redirected the unregistered	

2. Select and book a reservation.	log in or register before proceeding with the reservation process. 2. The user stopped the unregistered user from completing the reservation until the login or registration process was finished.	user to the registration/login page before completing the reservation.
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-010	**UC Reference**	Navigation Bar: Landing Page Access, Unregistered User, SK officials, Youth User
Objective	This test case is executed to confirm that the navigation bar and landing page load correctly, to ensure that the links in the navigation bar are working fine, and the elements in the page are correctly displayed on the landing page.		
Assumptions/Preconditions	The registered and unregistered users are able to access the landing page of the system without facing any restrictions. This makes the landing page accessible to anyone. All user types should be able to click and navigate the page smoothly with no access issues.		
Actions	**Expected Result**	**Actual Result**	

1. Navigate the different navigation links.	1. Every button clicked, should successfully redirect to its respective page.	1. The buttons in the system functioned correctly and every click redirected the user to the respective pages as expected.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-FUN-012	**UC Reference**	Programs, Youth User
Objective		To ensure that youth users can successfully navigate and view details of each program through the program navigation link.		
Assumptions/Preconditions		Registered youth users should have access to the homepage and be logged in to their account.		
Actions		**Expected Result**	**Actual Result**	
1. On the user's dashboard, click the "More Details" button for the desired program.		1. The system should successfully direct the user to the program details page upon clicking "More Details." 2. The user should be able to view the program description on the details page.	1. After clicking the "More Details" button, users were directed to the program details page, where they could view the program descriptions.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-013	**UC Reference**	View Schedule, Youth User

Objective	To verify that the user can view available time slots in real-time for scheduling.	
Assumptions/Preconditions	The user is logged in and has access to the program's scheduling feature. The program offers available time slots for scheduling.	
Actions	**Expected Result**	**Actual Result**
1. Click the "View Schedule" button of the specific program. 2. Select the desired time and date from the available options.	1. The system should display available time slots for the program in real-time. 2. The user should be able to select their preferred time and date. 3. The system should show whether the selected time and date are taken or still available.	1. Users were able to view the real-time availability of their desired time and date. 2. The system allowed users to select their preferred time and date and displayed whether the specific time slot was available or taken.

Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-014	**UC Reference**	Facility Reservation, Youth User
Objective	To ensure that the youth user can successfully book a facility by selecting a time slot, filling in personal details, agreeing to the waiver, and confirming the reservation.		
Assumptions/Preconditions	The user is logged in and has access to the facility booking/reservation system, and the facility is available for booking.		
Actions	**Expected Result**	**Actual Result**	

1. Navigate to the chosen facility and click the "Book Now" button. 2. Choose between "Solo" or "Group" booking options. 3. Select the available time and date for booking, then click "Confirm." 4. Click the "Next" button after the auto-fill-up information. 5. Check the "Reservation Details" and then click "Go to Waiver." 6. To fully confirm the reservation, check the checkbox, then click "Go to waiver." 7. Click the agreement checkbox to submit the reservation.	Expected Result 1. The system should guide the user through the reservation process, allowing them to choose between "Solo" or "Group" booking. 2. Once the user selects a day and time slot, the system should automatically display their details and allow them to review and confirm the booking. 3. The system should display the reservation details and direct the user to the waiver page, where the user acknowledges and submits the reservation. 4. The user will be redirected to their dashboard after successfully submitting the reservation.	1. Upon clicking the "Book Now" button, a modal appeared, allowing the user to select "Solo" or "Group" booking. 2. The user successfully proceeded to select the desired time and date for the reservation. 3. The system automatically populated the user's details in the reservation form. 4. The system displayed the reservation details for review and allowed the user to proceed to the waiver page. 5. After acknowledging the waiver and clicking "Submit," the reservation was confirmed, and the user was redirected to their dashboard. 6. The system logged the reservation in the user's reservation history, showing the booking details.
Status PASSED	**Severity**	**Priority**

Test Case ID	ISK-FUN-015	**UC Reference**	Reservation Log, Youth User
Objective	To ensure that the system correctly displays the user's reservation history and allows for filtering by facility or equipment, as well as deleting past reservations.		

Assumptions/Preconditions	1. The user is logged in and has made at least one reservation. 2. The system has a functional reservation history feature. 3. The user has access to the "Reservation" section in the navigation.	
Actions	**Expected Result**	**Actual Result**
1. Navigate to the "Reservation" section in the navigation menu. 2. Click the dropdown menu to select either "Facility" or "Equipment" reservation history	1. After clicking "Reservation" in the navigation, the user is directed to the reservation history page. 2. The dropdown should allow the user to filter the history by "Facility" or "Equipment." 3. The system should display the reservation history in a table.	1. Upon clicking the "Reservation" link, the user was redirected to the reservation history page. 2. The dropdown menu allowed the user to select between facility or equipment history. 3. The system displayed the reservation history in a table with details such as the start and end dates, time slot, reservation status, and a delete button for each reservation.

Status	PASSED	Severity		Priority	

Test Case ID	ISK-FUN-016	UC Reference	Reservation Log, Youth User-Delete Action
Objective	To verify that the system allowed registered users to delete a reservation from the reservation log and return to the reservation history page.		
Assumptions/Preconditions	1. The user is logged in and has made at least one reservation.		

	2. The system will display a confirmation message when the user desires to delete some of the reservation data from the reservations log.	
Actions	**Expected Result**	**Actual Result**
1. In the reservation log, look for the column named "Action", then click the "Delete" button under it for the selected reservation by the user. 2. Select a reason for cancellation, then put a check to the checkbox assigned to it. 3. Click either the "Cancel" or "Confirm" button to proceed. 4. Click "Return to Log" after confirmation.	1. When the "Delete" button is clicked, the system redirects the user to the cancellation of reservation called "Cancel Reservation" page. 2. The page will display options to select a reason for cancellation and a confirmation checkbox. 3. If the user clicks the "Cancel" button, they will return to the reservation log without any changes. 4. For the "Confirm" button, the system will display a success modal confirming the cancellation.	1. Upon clicking the "Delete" button, the system navigated to the "Cancel Reservation" page, displaying the reservation details, cancellation reasons, and confirmation checkbox. 2. After selecting a reason and checking the confirmation box, the user clicked the "Confirm" button, and a success modal appeared. 3. After clicking the "Return to Log", the user was redirected to the Reservation Log. 4. The reservation log was updated, showing the history without the deleted reservation.
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-017	**UC Reference**	Equipment Reservation, Youth User

Objective	To check the process of booking or reserving equipment by a youth user to confirm where the flow in the selection of equipment, selection of reservation date, displaying of information, and waiver acknowledgment, followed by reservation confirmation is properly organized.
Assumptions/Preconditions	1. The user is logged in and has access to the equipment reservation section. 2. The system displays all available dates for equipment booking.

Actions	Expected Result	Actual Result
1. Click on the "HiramGamit" link for equipment selection. 2. Choose the desired equipment and specify the quantity, then click "Book Now". 3. Select the preferred reservation date and click "Apply Dates". 4. Click the "Next" button to review reservation information, and/or click the "Previous" button on the equipment selection page. 5. Click the "Go to waiver" button, then click the checkbox and click "Submit."	1. Upon clicking the "Book Now" button, the system will display the available dates for the selected equipment. 2. After selecting a specific date, the system will proceed to the detailed reservation information page. 3. The system will display the reservation details. 4. The user will be able to click on the agreement terms checkbox before submitting the reservation.	1. After clicking the "Book Now" button, the system displayed the available dates for equipment booking. 2. Upon selecting a date, the system directed the user to the reservation information page, which auto-filled participant details. 3. The "Next" button displayed the reservation details, while the "Previous" button successfully returned the user to the equipment selection page. 4. Clicking the "Go to Waiver" button took the user to the Borrower's Waiver page, where they could agree to the terms. 5. After confirming the waiver, the system displayed a success modal and redirected the user to the reservation log

				page, showing the updated reservation details.
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-018	**UC Reference**	Equipment Booking/Reservation - Quantity Checking, Youth User
Objective	To check if the system properly handles equipment quantity selection, allows the users to be informed when the maximum quantity has been reached, and gives them access to close the notice modal.		
Assumptions/Preconditions	1. The user is logged in and can access the Equipment Reservation page. 2. The system displays the available equipment along with the quantity of equipment available for the selected week. 3. The system has a defined maximum quantity limit for each piece of equipment.		
Action	**Expected Result**	**Actual Result**	

1. Navigate to the Equipment page and select the desired quantity for a specific piece of equipment. 2. If the maximum quantity is reached, a notice modal will appear. Click the "Close" button to dismiss the modal.	1. The system should display the available quantity of equipment for the selected week. 2. If the user attempts to select a quantity that exceeds the available stock, a modal will appear, indicating that the maximum quantity has been reached.	1. The system correctly displayed the available quantity for the specific equipment. 2. Upon attempting to select a quantity beyond the limit, the system displayed a notice modal indicating the maximum quantity reached. 3. The user was able to close the modal by clicking the "Close" button.

Status	PASSED	Severity		Priority	

Test Case ID	ISK-FUN-020	UC Reference	Profile, Youth User
Objective	To confirm that the user profile is accessible by the user and it displays important details but without edit capabilities.		
Assumptions/Preconditions	1. The user is logged in as a youth participant. 2. The system's profile page is accessible from the navigation sidebar.		
Actions	**Expected Result**	**Actual Result**	

1. Access the system's navigation sidebar. 2. Click on the "Profile" option.	1. The profile page should be displayed. 2. The system should show the user's vital information that is view-only and cannot be edited by the user.	1. The profile page loaded correctly. 2. The system displayed the user's vital information, which was not editable, as expected.		
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-021	**UC Reference**	Change Password, Youth User
Objective	To verify that a youth user can successfully change their password through the profile page.		
Assumptions/Preconditions	1. The user is logged in and has access to their profile page. 2. The system allows password changes from the profile settings.		
Actions	**Expected Result**	**Actual Result**	
1. From the Profile Page, click "Changed Password". 2. Enter the old, and new, and confirm the new password. 3. Click "Close" to cancel the changes, and click "Submit" to proceed with the changes.	1. The system should successfully change the password when "Submit" is clicked.	1. A modal with designated fields for the old and new passwords appeared. 2. Upon clicking "Submit," the system successfully changed the password, and a confirmation modal was displayed. 3. After confirming the change, the system redirected	

			the user back to the profile page.	
Status	PASSED	**Severity**		**Priority**

Test Case ID		ISK-FUN-022	**UC Reference**	Logout, Youth User
Objective		To verify that a youth user can successfully log out of their account and be redirected to the home page.		
Assumptions/Preconditions		The system allows the youth/user to log out of the system.		
Actions		**Expected Result**	**Actual Result**	
1. Access the Profile on the navigation sidebar. 2. Click the "Logout" button.		1. The account should be successfully logged out.	1. The account was successfully logged out and redirected back to the home page.	
Status	PASSED	**Severity**		**Priority**

Test Case ID	ISK-FUN-024	**UC Reference**	Admin Dashboard - Reservation and User Statistics View
Objective	To verify that the admin has been granted access to view the data regarding reservations made in the system by the users, total users, and equipment per month through the system's dashboard.		

Assumptions/Preconditions	The user must be an admin, which is an SK official or member of it, and logged into the system.	
Actions	**Expected Result**	**Actual Result**
1. Click the "Home" from the sidebar.	1. The admin can view the data tables displaying the reservation and statistics view of the user.	1. The admin successfully viewed the relevant data tables, including the total number of users, reservation counts, and equipment data per month.
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-028	**UC Reference**	Manage Website: Manage Program Page - All Programs, Admin Side
Objective	To test and verify the functionality of the "All Programs" section in the "Manage Program Page" for the admin interface.		
Assumptions/Preconditions	The user must be an admin, which is an SK official or member of it, and logged into the system.		
Actions	**Expected Result**	**Actual Result**	
1. From the Manage website, click "Manage Program Page". 2. From "All Programs", click either of the three: "View Details", "Edit	1. Admin should be able to view, edit, and delete program details. 2. Any changes made should be saved and reflected on both the	1. Admin successfully viewed, edited, and deleted program details. 2. The changes were successfully displayed on both the admin and user interfaces.	

Details", and "Delete Program".	admin and user interfaces.				
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-FUN-029	**UC Reference**	Manage Website: Manage Program Page - Add Program, Admin Side
Objective		To test the functionality of the "Add Program" feature in the "Manage Program Page" for the admin interface.		
Assumptions/Preconditions		The user must be an admin, which is an SK official or a member of it, and be logged in to the system.		
Actions		**Expected Result**	**Actual Result**	
1. From Manage Program, select "Add Program". 2. Enter the required information into the field and choose a file for both the program and the highlights image. 3. Click the "Save Changes" button.		1. Admin should be able to successfully add a new program. 2. The newly added program, along with its images, will be displayed correctly on both the admin and user pages.	1. The new program and its associated information and images were successfully added and displayed on both the admin and user interfaces.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-FUN-030	UC Reference	Manage Website: Manage Contact Us Page, Admin Side
Objective		To test the functionality of the "Manage Contact Us Page" for the admin interface.		
Assumptions/Preconditions		The user must be an admin, which is an SK official or member of it, and logged in on the system.		
Actions		**Expected Result**	**Actual Result**	
1. Under the Manage Website, click "Manage Contact Us Page. 2. Click the "Edit Details" button. 3. Click the "Save Details" button after adding/editing details.		1. The added or changes made to the contact details will be reflected in the admin and youth interface.	1. The content changes were successfully saved and displayed on the youth user interface.	
Status	PASSED	**Severity**	**Priority**	

Test Case ID	ISK-FUN-031	UC Reference	Admin (Reservations' "Gym Reservation" Sub-Module)
Objective	Test and view "Gym Reservation" sub-module functionalities.		
Assumptions/Preconditions	The user must be an admin and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	

1. Click the "Reservations" from the sidebar. 2. Under "Reservations", click "Gym Reservation". 3. Click any day on the calendar. 4. Put a check on the checkbox on the gym reservation list. 5. Click either the "Approve" or "Disapprove" button. 6. Click the "Archive" button.	1. The system should navigate the admin to the appropriate reservation page, either for the gym or display relevant options for managing reservations. 2. After clicking any day on the calendar, the admin should be able to view numbers on dates with reservations. 3. The admin should be able to change the status of the reservation. 4. Admin should be able to archive the selected reservation on the list.	1. The system successfully redirected to the corresponding reservation page, allowing the admin to view and manage reservations. 2. The applied actions successfully changed the reservation status and list of reservations.

Status	PASSED	Severity		Priority	

Test Case ID	ISK-FUN-032	UC Reference	Admin (Reservations' "Inventory" Sub-Module)
Objective	Test and view Reservations' "Inventory" sub-module functionalities.		
Assumptions/Preconditions	The user must be an admin and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	
1. Click the "Reservations" from the sidebar.	1. The system should navigate the admin to the appropriate	1. The system successfully redirected to the corresponding reservation page, allowing the admin to	

2. Click the "Inventory" under "Reservations" from the sidebar. 3. Click the "Edit" button on a specific equipment to edit its details. 4. Click the "Delete" button on a specific equipment to delete it from the equipment list. 5. Click the "Add Equipment" to add new equipment. 6. Fill out the blank contents of the form. 7. Click the "Choose file" to upload the image of equipment that will be displayed on the website. 8. Click the "Add Item" button.	inventory page where relevant data related to it will be shown. 2. By clicking the "Save Changes " button after editing the details of the specific equipment, the changes must be reflected in the "Inventory" of admin and "Equipment Reservation" of youths. 3. After clicking the "Delete" button of specific equipment, the item must be removed from the system. 4. The newly added equipment must be reflected in the "Inventory" of admin and "Equipment Reservation" of youths.	view and manage reservations. 2. The system successfully interacted with the functionalities of each button performed and turned out well.
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-033	**UC Reference**	Admin (Reports)
Objective	View the data related to reservation reports and test generate report functionalities.		
Assumptions/Preconditions	The user must be an admin and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	

1. Click the "Reports" from the sidebar. 2. Select from report classification. 3. Select the range schedule of the report from the date classification. 4. Fill out the admin name in the text box. 5. Click the "Generate PDF" button.	1. After clicking the "Reports" from the sidebar, the admin must be able to view the data related to the reservations that have been made with the system by youths. 2. Proceeding to the page, the admin should be able to view the reports related to the selected category of reports. 3. By filling out the selection and admin name and clicking the "Generate PDF" button, the system should automatically download the report in PDF format.	1. The system successfully redirected to the corresponding reports page, allowing the admin to view lists of reports. 2. The system successfully displayed relevant ideas related to the reports selected. 3. The PDF file successfully printed the selected report chosen from the system and within the time range.

Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-FUN-034	**UC Reference**	Admin (Users)
Objective	Test and view "Users" functionalities.		
Assumptions/Preconditions	The user must be an admin and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	

1. Click the "Users" from the sidebar. 2. Click the "View" button of a specific user to view complete details. 3. Click the "Edit" button of a specific user to edit the user's details. 4. Click the "Delete" button of a specific user to delete it from the users' list. 5. Click the "Add User" to add new user details. 6. Fill out the blank contents of the form. 7. Click the "Add User" button.	1. After clicking the "Users" from the sidebar, the admin must be able to view the data related to the list of users implemented on the database of the system. 2. After clicking the "View" button of a specific user, the modal that has the complete details of the user will be displayed. 3. After clicking the "Edit" button of a specific user, the modal that has the complete current details of the user will be displayed, allowing the admin to edit it and save it by clicking the "Update User" button. 4. After clicking the "Delete" button of a specific user, the modal will pop out to make sure that the admin wants to delete the user information from the system, and by clicking the "Yes" button, the selected user details will be deleted from the system. 5. The newly added user details must be reflected	1. The system successfully redirected to the corresponding user page, allowing the admin to view lists of user details. 2. The system successfully displayed relevant ideas related to the list of users' data. 3. The system successfully interacted with the functionalities of each button performed and displayed necessary information based on the expected outcome.

		in the list of "Users" of the admin.	
Status	PASSED	**Severity**	**Priority**

Test Case ID	ISK-FUN-035	**UC Reference**	Admin (Reservations' "Equipment Reservation" Sub-Module)
Objective	Test and view "Equipment Reservation" sub-module functionalities.		
Assumptions/Preconditions	The user must be an admin and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	
1. Click the "Reservations" from the sidebar. 2. Under "Reservations", click "Equipment Reservation". 3. Click any day on the calendar. 4. Put a check on the checkbox on the equipment reservation list. 5. Click either the "Approve" or "Disapprove" button. 6. Click the "Archive" button.	1. The system should navigate the admin to the appropriate reservation page, either for the gym or equipment and display relevant options for managing reservations. 2. After clicking any day on the calendar, the admin should be able to view numbers on dates with reservations. 3. The admin should be able to change the status of the reservation.	1. The system successfully redirected to the corresponding reservation page, allowing the admin to view and manage reservations. 2. The applied actions successfully changed the reservation status and list of reservations.	

		4. Admin should be able to archive the selected reservation on the list.			
Status	PASSED	**Severity**		**Priority**	

Appendix E

TEST CASE RELIABILITY

Test Case ID		ISK-REL-001	UC Reference	
Objective		To verify if the system allows users to successfully perform a standard reservation workflow repeatedly under normal conditions, ensuring that the system can handle multiple requests without errors.		
Assumptions/Preconditions		The user is a verified youth who is logged into the system, and the system is functioning under normal operating conditions.		
Actions		**Expected Result**	**Actual Result**	
1. Navigate to the program page. 2. Select the desired facility or equipment. 3. Choose a time slot for the reservation, then submit. 4. Repeat the process multiple times to ensure functionality.		1. The system should allow the user to complete the reservation successfully each time. 2. After each submission, the system must confirm and display relevant details. 3. The system will not crash or display any errors during multiple repetitions.	1. The system allows the user to complete the reservation successfully each time. 2. After each submission, the system confirmed and displayed the relevant details. 3. There were no crashes or errors during multiple repetitions.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-REL-003	UC Reference	
Objective	To verify that the system handles errors during reservation operations without crashing and provides error messages to the user.		
Assumptions/Preconditions	The user is a verified youth and logged into the system. Error-handling mechanisms are implemented in the system.		
Actions	**Expected Result**	**Actual Result**	
1. Perform a reservation with incorrect or invalid input. 2. Observe the system's behavior.	1. The system should display an appropriate error message without crashing. 2. The user should be able to correct the error and proceed with the reservation process.	1. The system displayed an error message, and the user was able to correct the mistake and proceed.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-REL-004	UC Reference	
Objective	To verify that the system reliably sends and verifies email addresses without failure and under varying conditions.		
Assumptions/Preconditions	The user is registered and the email system is operational.		
Actions	**Expected Result**	**Actual Result**	

1. Type in a valid email address. 2. Ensure the email verification code is sent to the provided email address.	1. The system should reliably send verification emails without delay or failure. 2. The verification code should be operational. 3. The system should accept the verification code and allow the user to proceed.	1. Verification codes were sent and received by the users without failure. 2. The code worked without any errors or delays.

Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-REL-005	**UC Reference**	
Objective		To verify that the system remains smooth and responsive during various user interactions, including navigating between pages and submitting requests under different load conditions.		
Assumptions/Preconditions		The system is operational.		
Actions		**Expected Result**	**Actual Result**	
1. Navigate through different sections of the system. 2. Perform actions such as making reservations and navigating between pages.		1. The system should remain smooth and responsive, with page loads within acceptable time limits. 2. Actions should be executed without delays, and all UI elements should be properly rendered.	1. The system remained smooth and responsive, with pages loading within acceptable time limits. Actions were executed without delays, and all UI elements were properly rendered. 2. No significant lag or freezing occurred, even under high load	

		3. No significant lag or freezing should occur, even under high load conditions.		conditions, ensuring a seamless user experience.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-REL-006	**UC Reference**	
Objective		To ensure that the system reliably handles conflicts or double bookings, displaying availability status correctly.		
Assumptions/Preconditions		1. The system has a database of available resources. 2. The system must be initialized with correct availability data.		
Actions		**Expected Result**	**Actual Result**	
1. Simulate multiple booking attempts for the same item or facility at overlapping times. 2. Attempt to book the same equipment or facility at conflicting times.		1. The system should correctly indicate when an item or facility is unavailable. 2. The system should not allow double bookings for the same time slot or item.	1. The system correctly indicated when an item or facility was unavailable and prevented double bookings for the same time slot or item. Attempts to reserve an already booked item or facility were blocked, ensuring no overlapping reservations.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID		ISK-REL-007	UC Reference	
Objective		To verify that the booking data remains accurate and consistent during and after the booking process.		
Assumptions/Preconditions		1. The system is operational, and the database is connected. 2. Booking details such as user, date, and time are properly recorded.		
Actions		**Expected Result**	**Actual Result**	
1. Make a booking (select equipment, date, and time). 2. After submission, check the system's database for consistency with the displayed booking details.		1. The booking details should be stored accurately in the system's database. 2. The data should match the details of the user during the reservation process.	1. The system accurately stored the booking details in the database, and the data matched the information provided by the user during the reservation process. There were no discrepancies between the reservation details submitted by the user and those stored in the system.	
Status	PASSED	**Severity**		**Priority**	

Test Case ID	ISK-REL-008	UC Reference	
Objective	To verify that the system allows admins to approve or disapprove facility reservations, and accurately reflects this in the user interface.		
Assumptions/Preconditions	1. The system is initialized and functional. 2. The admin has access to the reservation management interface. 3. The youth has completed a facility reservation.		
Actions	**Expected Result**	**Actual Result**	

1. Log in as an admin and navigate to the facility reservation approval section. 2. For a specific reservation, click "Approve" or "Disapprove." 3. After approval or disapproval, check if the status is updated in both admin and user interfaces.	1. The admin can approve or disapprove the reservation. 2. The reservation status is updated accordingly and will be displayed in both the admin and user interfaces.	1. The admin successfully approved and/or disapproved the reservation. 2. The reservation status was updated accordingly and was displayed in both the admin and the user interfaces.	
Status	PASSED	**Severity**	**Priority**

Test Case ID	ISK-REL-009	**UC Reference**	
Objective	To verify that the system accurately allows the admin to mark equipment reservations as "Returned" or "Not Returned," and reflects this status correctly in the system.		
Assumptions/Preconditions	1. The system is initialized and functional. 2. The equipment reservation has been made. 3. The admin has access to the reservation management interface.		
Actions	**Expected Result**	**Actual Result**	
1. Log in as an admin and navigate to the equipment reservation section. 2. Mark the equipment as "Returned" or "Not Returned." 3. Check if the equipment status is updated accurately in the system.	1. The admin can mark the equipment reservation as "Returned" or "Not Returned." 2. The status should be reflected in the admin interface and the user.	1. The admin successfully marked the equipment reservation as "Returned" and "Not Returned". 2. The updated status is reflected in the both admin and user's interface list.	

Status	PASSED	Severity		Priority	

Test Case ID		ISK-REL-010	UC Reference	
Objective		To verify that the system can handle multiple facility and equipment approvals and status without performance degradation or data loss.		
Assumptions/Preconditions		1. The system is operational. 2. Load simulation tools or test scripts are available to simulate multiple reservation approvals.		
Actions		**Expected Result**	**Actual Result**	
1. Simulate approving or disapproving multiple facility reservations and mark multiple equipment reservations at the same time. 2. Observe the system's performance and ensure that no errors occur.		1. The system should be able to handle the load and update each reservation status accordingly. 2. There should be no performance issues or data corruption during the approval process.	1. The system successfully handled the load, updating each reservation status accordingly without any delays or errors. There were no performance issues, and the system-maintained data integrity throughout the approval process, with all updates reflecting accurately in both the admin and user interfaces.	
Status	PASSED	**Severity**	**Priority**	

Test Case ID		ISK-REL-011	UC Reference	

Objective	To verify the website's responsiveness, ensure it adjusts seamlessly across different screen sizes without loss of functionality or usability.	
Assumptions/Preconditions	1. The user is accessing the website from various devices 2. The website is fully operational and accessible.	
Actions	**Expected Result**	**Actual Result**
1. Open the website on a desktop, tablet, and mobile device. 2. Resize the browser window from large to small (on desktop). 3. Test key features like navigation, buttons, and forms for accessibility and functionality. 4. Check the page load time on different devices. 5. Inspect text, images, and buttons for proper alignment and responsiveness.	1. The website should adjust its layout and content according to the screen size without overlapping or breaking elements. 2. Navigation, buttons, and forms should be accessible and clickable. 3. No significant UI errors or misalignments should occur, even on smaller screens. 7. All functions should remain usable across devices, including mobile-friendly elements like touch-based navigation.	1. The website layout adjusted properly on mobile and desktop, with no overlap of elements. 2. Navigation was functional on mobile and desktop devices. 3. Buttons and forms remained usable and properly aligned. 4. There was no misalignment of text or images.
Status	PASSED	**Severity**		**Priority**	

Appendix F

TEST INCIDENTS LOGS

Test Case ID	ISK-FUN-011	**UC Reference**	Dashboard, Youth User
Objective	To validate whether youth users can effectively navigate the homepage features and submit star rating feedback on their website experience.		
Assumptions/Preconditions	Youth users who are registered and logged in should be able to access and navigate the homepage.		
Actions	**Expected Result**	**Actual Result**	
1. Click on the home navigation link. 2. Navigate through the homepage and use available features. 3. After using the features, submit star rating feedback.	1. The user should be able to navigate the dashboard without any conflicts. 2. The system should successfully collect and store the star rating feedback submitted by users. 3. The star rating submitted by the user will be displayed at its end.	1. The user was able to navigate the dashboard without any problems. 2. The user was able to select a rating, but it was not displayed correctly after submission.	
Status	FAILED	**Severity**	MINOR	**Priority**	LOW

Test Case ID	ISK-FUN-019	**UC Reference**	Breadcrumb Navigation, Youth User
Objective	To validate the functionality of all breadcrumbs across the system's pages, ensuring that they navigate users to the correct previous page.		

Assumptions/Preconditions	The user has navigated to various pages within the system, and is visible and available across all pages.	
Actions	**Expected Result**	**Actual Result**
1. Click on the breadcrumb to navigate to the next page.	1. Each breadcrumb should correctly take the user to the corresponding previous page. 2. All breadcrumbs across the system should function as intended, with correct page navigation.	1. All breadcrumbs worked correctly, except for the breadcrumb on the "Equipment Reservation" page. 2. The "Equipment" breadcrumb in the "Reservation Equipment" page did not navigate the user back to the previous page, causing an issue in breadcrumb functionality.

Status	FAILED	**Severity**	MINOR	**Priority**	MEDIUM

Test Case ID	ISK-FUN-023	**UC Reference**	Reservation Date Selection, Youth User
Objective	To verify that users cannot select Sunday as a reservation day, as it should be marked as unavailable.		
Assumptions/Preconditions	1. The user is logged in and has access to the Programs page. 2. The system is set up to mark Sunday as unavailable for reservations.		
Actions	**Expected Result**	**Actual Result**	
1. Access the chosen program or equipment from the Programs page. 2. Click "Book Now".	1. Sunday should be unavailable for selection and marked as	1. Sunday was selectable, and a reservation was successfully set for that day.	

3. Select the desired date.	such on the reservation calendar.		
Status	FAILED	**Severity**	MINOR	**Priority**	MEDIUM

Test Case ID	ISK-FUN-025	**UC Reference**	Manage Website:Manage Homepage - Add event/spotlight, Admin Side
Objective	To test and verify the functionality of adding events and spotlight images through the "Manage Homepage" feature on the admin side.		
Assumptions/Preconditions	The user must be an admin, which is SK officials or members, and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	
1. Click the "Manage Website" from the sidebar. 2. Click the "Manage Home Page" 3. Click the "Add Event/Spotlight" link. 4. Complete the form with event details 5. Click the "Choose file" to upload the image associated with the event/spotlight. 6. Click the "Add Event/Spotlight" button.	1. The admin should successfully access the "Manage Home Page" interface and be able to add event details and spotlight images. 2. The newly added event/spotlight will be reflected in the "Manage Home Page" (Admin interface), as well as on the user interface visible to youth.	1. The admin successfully accessed the "Manage Home Page". 2. Events and spotlight images/information were successfully added and displayed correctly on the admin interface. 3. The newly added event/spotlight was reflected in both the "Manage Home Page" and "Manage Program Page" for both the admin and user interfaces. 4. In the "Add Events" section, the hover effect was obscured by the	

			uploaded images, making it difficult to interact with the section. 5. The title for "Add Spotlight" was not displayed consecutively, disrupting the intended layout.

Status	FAILED	**Severity**	MINOR	**Priority**	LOW

Test Case ID	ISK-FUN-026	**UC Reference**	Manage Website:Manage Homepage - Edit and delete Events /Spotlight, Admin Side
Objective	To test and verify the functionality for editing and deleting events and spotlight details on the "Manage Homepage" feature from the admin side.		
Assumptions/Preconditions	The user must be an admin, which is SK officials or members, and logged in on the system.		
Actions	**Expected Result**	**Actual Result**	
1. From "All Events" in Manage Homepage, click either "Edit Details" or "Delete Event. 2. Click either "Edit Details" or "Delete Event" for a selected event.	1. Admin should be able to edit the name and description of events and save changes. 2. The admin should be able to delete spotlight images successfully.	1. Admin successfully saved changes when editing event names and descriptions. 2. Spotlight images were not deleted. 3. The updated event details were correctly displayed	

3. Edit the event name and description, then click "Cancel" or "Save Changes". 4. Click the delete icon for a spotlight to remove it.	3. Any changes made will be reflected in both the "Manage Home Page" and "Manage Program Page" on both the admin and youth interfaces.	on both the admin and user interfaces.

Status	FAILED	Severity	MINOR	Priority	LOW

Test Case ID	ISK-FUN-027	UC Reference	Manage Website:Manage About Us Page-Admin Side
Objective	To test and verify the functionality for managing the "About Us" details and SK Council Members section on the "Manage Website" page from the admin side		
Assumptions/Preconditions	The user must be an admin, which is SK officials or members, and logged in on the system.		

Actions	Expected Result	Actual Result
1. From the Manage website, click "Manage About Us Page". 2. Click "Edit Details" for the "Manage About Us Details" section, modify the details, and click "Save Details" to confirm the changes. 3. If needed, click the "Back" button to cancel. 4. Under the "SK Council Members" section, click	1. Admin should be able to edit both the "Manage About Us Details" and the "SK Council Members" sections. 2. Any changes made should be visible on both the user and admin pages.	1. Admin successfully edited the details, and changes were reflected on both the admin and user interfaces. 2. The image added for the "Manage About Us Details" section was not reflected on the user's homepage. 4. There was no "Cancel" button under the SK Council Members section.

the "Edit" button to modify details.			

Status	FAILED	Severity	MINOR	Priority	LOW

Test Case ID		ISK-REL-002	UC Reference	
Objective		To verify that the system performs efficiently and remains stable under varying levels of user load, such as single-user and multiple users accessing the system simultaneously.		
Assumptions/Preconditions		The user is a verified youth and logged into the system, and the system is initialized and fully operational. Load simulation tools or test scripts are available to simulate multiple users.		
Actions		**Expected Result**	**Actual Result**	
1. Log into the system as a single or multiple users accessing the system simultaneously. 2. Perform similar tasks, like reservations, viewing available dates, and canceling reservations. 3. Observe or monitor the system's performance.		1. The system should perform consistently, with no significant performance degradation during single and multiple-user use. 2. System should be able to handle a large number of simultaneous users without crashing slowdowns, or loss of functionality.	1. The system performed consistently, with no significant performance degradation during both single-user and multiple-user use. 2. The system successfully handled a large number of simultaneous users without experiencing crashes, slowdowns, or loss of functionality.	
Status	FAILED	**Severity**	MAJOR	**Priority**	MEDIUM

Appendix G

GANTT CHART

Task	Feb	March	April	May	June	July	Aug	Sept	Oct	Nov	Dec	Jan
Generate various ideas	X											
Communicate with Client (interview)	X											
Title Proposal		X										
Abstract Critique		X										
Data Garthering		X	X									
Title Presentation			X									
Planning			X	X								
Chapter 1-3				X	X							
Review of Related Literature				X	X							
Review of Related Studies				X	X							
Chapter 1-3 Defense					X							
Chapter 1-3 Revisions						X						
Evaluation and Procedure						X						
User Interface					X	X						
Road Map					X							
Change Topic						X	X					
Meeting with New Client							X					
Defense of New Title							X					
Front-End Coding							X	X	X	X	X	
Back-End Coding								X	X	X	X	
Chapter 4-5							X	X	X			
Creating Database									X	X	X	
Connecting Database									X	X	X	
Testing and Debugging										X	X	
Final Evaluation										X	X	
Data Garthering											X	X
Final Defense												X
Final Documentation												X

Appendix H

USER MANUAL

Introduction

Welcome to the user manual for the iSKed Reservation System. This manual is designed to guide you through the features and functionalities of the platform, which provides an efficient and user-friendly online reservation process for various facilities and equipment managed by the SK of Western Bicutan. Whether you're booking a venue or borrowing equipment, this manual will assist you in making the most of the system and ensuring a smooth and efficient reservation process.

1. User Registration and Authentication

a. Creating an Account:

1. Access the iSKed platform via the website link to proceed with account creation.

2. Click the "Sign Up" button.

3. Retrieve your unique activation code from the SKWB office.

4. Enter the activation code to open your account.

b. Sign Up:

1. Use the activation code provided by SKWB to open your account.

2. Choose a username and password.

c. Password Recovery:

1. If you forget your password, click the "Forgot Password".

2. Get the code that was sent to your email to securely reset your password.

3. Set a new password after following the steps.

d. Change Password:

1. Navigate to the "Profile" section.

2. Click the "Change Password" button.

3. Fill in your old password.

4. Set a new password.

5. Confirm your new password.

2. Making Reservations

a. Booking Facilities:

1. View the calendar to check available time slots for facilities.
2. Select one or multiple days by clicking on the desired dates twice to reserve them at the same time.
3. Specify the time slot and fill in the required date and time.
4. Confirm your booking to avoid double reservations.
5. Review and agree to the terms and conditions by checking the waiver box before finalizing your reservation.

b. Booking Equipment:

1. Go to the "Equipment" page.
2. Select the equipment you wish to reserve.
3. Check if the equipment is available for your chosen date and time.
4. Specify the quantity of equipment needed.

5. Use the calendar to check and select available time slots.

6. Select one or multiple days by clicking on the desired dates twice to reserve them at the same time.

7. Confirm your booking to avoid double reservations.

8. Review and agree to the terms and conditions by checking the waiver box before finalizing your reservation.

3. Managing Reservations

a. Viewing and Editing Reservations:

1. Access the "Reservation Log" section from your dashboard to view all bookings.

2. Filter your history to display either facility or equipment reservations as needed.

3. Review detailed logs of reservations, including dates and statuses.

4. Cancel or update upcoming reservations directly from the logs if necessary.

4. Feedback

a. Submitting Feedback:

1. Go to the home dashboard and scroll down.

2. Rate your experience using the 5-star rating system and you call also leave a comments.

Appendix I

THESIS GRAMMARIAN CERTIFICATION

	TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES Ayala Blvd., Ermita, Manila, 1000, Philippines Tel No. +632-5301-3001 local 608| Fax No. +632-8521-4063 Email: cos@tup.edu.ph	Website: www.tup.edu.ph	Index No.	REF-COS-3.5-INT-TGC
		Revision No.	00
		Effectivity Date	06132022
VAA-COS	THESIS GRAMMARIAN CERTIFICATION	Page	1 / 1

THESIS GRAMMARIAN CERTIFICATION

This is to certify that the thesis entitled,

**iSKed: DEVELOPMENT OF FACILITIES AND EQUIPMENT
RESERVATION MANAGENENT SYSTEM FOR BARANGAY WESTERN
BICUTAN**

authored by

JADE S. CORPIN
MARK JHAY CORPUZ
AUSTINE BILL RYAN N. MALIC
JEREMY PAUL V. RICERRA
MARIVIC O. TUAZON

has undergone editing and proofreading by the undersigned.

This Certification is being issued upon the request
for whatever purposes it may serve them.

Asso. Prof. JENNIFER P. ALINSUNOD, CHRA, LPT
Grammarian

Technological University of the Philippines

April 29 , 2025

Appendix J

CERTIFICATE OF SIMILARITY INDEX USING TURNITIN

Austine Bill Ryan Malic

MALIC_iSKed DEVELOPMENT OF FACILITIES AND EQUIPMENT RESERVATION MANAGEMENT SYSTEM FOR BAR...

- UNDERGRADUATE MANUSCRIPT 2024
- UNDERGRADUATE MANUSCRIPT 2024
- Technological University of the Philippines

Document Details

Submission ID
trn:oid:::25992:93586819

Submission Date
Apr 30, 2025, 9:12 AM GMT+8

Download Date
Apr 30, 2025, 10:43 AM GMT+8

File Name
MALIC_iSKed DEVELOPMENT OF FACILITIES AND EQUIPMENT RESERVATION MANAGEMENT SYST....docx

File Size
13.2 MB

234 Pages

31,864 Words

191,575 Characters

13% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

364 Not Cited or Quoted 13%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

4%   Internet sources
1%   Publications
11% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Appendix K

CERTIFICATE OF SIMILARITY INDEX USING TURNITIN FROM URDS

		Index No.	REF-URD-INT-CSI
	TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES Ayala Blvd., Ermita, Manila, 1000, Philippines Tel No. +632-5301-3001 local 711 | Fax No. +632-521-4063 Email: urds@tup.edu.ph | Website: www.tup.edu.ph	Issue No.	01
		Revision No.	01
		Date	04132021
		Page	1 / 1
VRE-URD	**CERTIFICATE OF SIMILARITY INDEX USING TURNITIN**	QAC No.	CC-04132021

This is to certify that the manuscript entitled,

iSKed: DEVELOPMENT OF FACILITIES AND EQUIPMENT RESERVATION MANAGEMENT SYSTEM FOR BARANGAY WESTERN BICUTAN

authored by

JADE S. CORPIN
MARK JHAY CORPUZ
AUSTINE BILL RYAN N. MALIC
JEREMY PAUL V. RICERRA
MARIVIC O. TUAZON

has been subjected to a similarity check on April 30, 2025, using Turnitin with a generated Similarity index of 13%.

Processed by:

DENNIS J. TABUCOL
Faculty Staff, URDS

Certified correct by:

ENGR. HERONAFINE C. DE GUZMAN
Director, URDS

Transaction ID	COS-BSIS-2425008
Signature	

RESEARCHER'S PROFILE

JADE S. CORPIN

Taguig, City   •   +63 976 503 5630   •   jadecorpin01@gmail.com

https://github.com/jadecrpn

OBJECTIVE

Dynamic and results-driven Bachelor of Science in Information Systems student with proven leadership as a Project Manager for a successful capstone project and as the lead researcher for multiple academic papers. Equipped with strong UI/UX design skills and front-end development, I aim to leverage my technical abilities, research experience, and project management skills in an on-the-job training role, contributing to organizational success while gaining valuable industry experience.

EDUCATION

Bachelor of Science in Information Systems   2021 -Present
Technological University of the Philippines - Manila

Secondary Education   2014 - 2019
Biliran Science High School

SKILLS

Technical Skills
- Front-end Development: HTML, CSS, JavaScript, React, Bootstrap
- Back-end Development: Python, C++, PHP
- Database Management: MySQL, PostgreSQL.
- Version Control: Git and Github

Tools and Technologies
- Project Management: Notion
- UI/UX Design: Figma
- Productivity and Collaboration: Microsoft Office Suite, Visual Studio Code, and Git and Github

Core Competencies
- Leadership and Collaboration: Demonstrated ability to lead and collaborate in team settings through academic and project-based work.
- Adaptability and Learning: Quick to adapt to new tools, technologies, and environment, applying acquired skill to diverse IS roles.
- Project Management and Coordination: Experience with planning, coordinating, and tracking project tasks to ensure successful project delivery, utilizing management tools and technologies.

Mark Jhay
Corpuz
Front-end Web Developer

CONTACT ME

📞 +63 9154779835

✉ markjhaycorpuzzz@gmail.com

📍 30 Morato St., Brgy,
Mariblo, Quezon City

EDUCATION

Primary Education

Congressman Reynaldo A. Calalay
Memorial Elementary School
2007 - 2014

Secondary Education/JHS

Sergio Osmeña Sr, High School
2019 - 2021

Secondary Education/SHS

STI College Muñoz-EDSA
2019 - 2021

Tertiary Education

Technological University of the
Philippines- Manila
2021 - 2025

PERSONAL DETAILS

Date of Birth: January 7, 2002

Age: 22

Civil Status: Single

KNOWLEDGE AND SKILLS

- **Basic skills in C, C++, Python, ReactJS**
- **Intermediate skills in HTML and CSS**
- **Knowledgeable in Video Editing**

ACADEMIC ACHIEVEMENTS

With Honor   SY: 2019 - 2020
Grade 11
STI College Muñoz-EDSA

With High Honor   SY: 2020 - 2021
Grade 12
STI College Muñoz-EDSA

Dean's Lister   SY: 2021 - 2022
1st Year - 1st Semester
Technological University of the
Philippines - Manila

Dean's Lister   SY: 2022 - 2023
2nd Year - 1st Semester
Technological University of the
Philippines - Manila

Dean's Lister   SY: 2022 - 2023
2nd Year - 2nd Semester
Technological University of the
Philippines - Manila

President's Lister   SY: 2023 - 2024
3rd Year - 1st Semester
Technological University of the
Philippines - Manila

AUSTINE BILL RYAN N. MALIC

Western Bicutan Taguig City | 09765419776 |

auztine.malic@gmail.com | https://github.com/AustineBill

OBJECTIVES

Web developer dedicated to crafting innovative and efficient digital solutions. Proficient in full-stack development, database management, and scalable deployment, with experience building responsive applications and streamlining processes using modern tools and frameworks. Skilled in coordinating development efforts to meet project goals and enhance user satisfaction in dynamic environments. Committed to continuous learning and leveraging emerging technologies to drive impactful results.

EDUCATION

Bachelor of Science in Information System   Aug 2021 - Present
Technological University of the Philippines - Manila

Secondary Education   2014 - 2021
Western Bicutan National High School

SKILLS

Technical Skills
- Front-End Development: HTML, CSS, JavaScript, React, Bootstrap
- Back-End Development: Python, C++, PHP, JavaScript
- Database Management: MySQL, PostgreSQL
- Version Control: Git and GitHub

Tools and Technologies
- Development tools: VS Code , Git, and Github.
- Database Management: pgAdmin and DBeaver.
- Cloud Platforms: Render and Heroku.

Core Competencies
- Full-Stack Development: Expertise in integrating front-end and back-end technologies to deliver seamless, functional web applications.
- Adaptability and Learning: Rapidly adapt to new tools and technologies, applying knowledge to diverse software development roles.

CERTIFICATES
- Certificate on SoloLearn as Web Development
- Certificate on SoloLearn as SQL
- Certificate on Coursera as Build Website on WordPress
- Certificate on Coursera as Business Analysis and Project Management

Jeremy Paul
Ricerra

Lead Front-end web developer

CONTACT ME

- +63 9707277553
- rjeremypaul@gmail.com
- www.reallygreatsite.com
- 10 Rosas St., Merville Subdivision., Tanza 1., Navotas City

EDUCATION

Primary Education
Malabon Elementary School
2007 - 2014

Secondary Education/ SHS
Malabon National High School
2014 - 2019

Tertiery Education
Technological University of the Philippines
2021 - 2025

SKILLS

- UI/UX
- Visual Design
- Wireframes
- Storyboards
- User Flows
- Web Development (PHP)
- React JS (Front-end framework)
- Hardware

JOB EXPERIENCE

Record Keeper   2019 - 2020
City of Malabon University
Record Keeper and Assistant at the Office of the President, City of Malabon University

PERSONAL ACHIEVEMENTS

Academic excellence award "with honors" April 2019

Certificate of excellence for achieving top 12 overall, in Empowerment technology, 11th grade 2018-2019

Certificate of excellence for achieving top 4, in Empowerment technology, 11th grade, 4th quarter 2018-2019.

Certificate of excellence for Contemporary Philippine Arts of the region-Tour Exhibit of school year 2018-2019, 11th grade Work Immersion Awardee for the year 2019-2020 in the subject of Work Immersion

Award for academic Excellence as With honors Grade 12 A.Y. 2019-2020

Certificate of completing Senior High School in Malabon National High School Senior High School Department.

REFERENCES

Paolo Xavier S. Co

Malabon National High School /Career Guidance Advocate

TUAZON, MARIVIC

CONTACT

- 0947-750-3847
- tuazonmarivic31@gmail.com
  marivic.tuazon@tup.edu.ph
- Tandang Sora Quezon City
- www.linkedin.com/in/marivic-tuazon-8951b2348/

EDUCATION

AMA Sta. Mesa - Manila
Senior High School
With Honor
2019-2021

Technological University of the Philippines - Manila
College
Dean's Lister
2021-Present

Certification
Data Privacy in the Digital Age

PERSONAL PROFILE

I am a 4th-year college student at the Technological University of the Philippines (TUP) Manila, pursuing a Bachelor of Science in Information Systems. I have foundational knowledge in programming languages, documentation, and quality analyst. While I am still improving in some areas, I am eager to learn and grow, especially in skills that align with my goals and interests.

PROJECTS

My Portfolio
- Built an interactive portfolio website using HTML, CSS, and JavaScript.
- Managed version control with GitHub and deployed it on Netlify.

Personal Interest Website
- Created a website showcasing my interests using HTML, CSS, and JavaScript.

Funeral Business Aggregator and E-Commerce (Upcoming)
- Contributed as a Data Researcher and UI Designer.

OneSpot: Mobile Parking Reservation with Real-Time Visualization
- Used JavaScript for development and contributed as a UI Designer and Documentation Specialist.

Inventory Management System
- Worked on front-end design using CSS and assisted with project documentation.

iSKed: Facilities and Equipment Reservation System
- Assisted in front-end development focusing on CSS for UI design.
- Contributed as a Data Researcher and Quality Analyst, performing system testing and ensuring quality.

SKILLS

- *UI/UX Design*

    Knowledgeable in UI/UX design using Figma.

- *Front-End Development*

    Experienced with HTML, CSS, and JavaScript, with a strong willingness to learn other programming languages.

- *Basic Microsoft Office Skills*

    Familiar with Microsoft Word and Excel.