Cloud-Based System for Sustainable Stingless Bee Farm

Chen Wang[1], Yutong Tang[1*], Mohamed Faisal Ahmad-Akhia[1], Hamzah Abdul-Rahman[2], Jeffrey Boon Hui Yap[3]

[1] College of Civil Engineering, Huaqiao University, China
[2] Faculty of Built Environment, University of Malaya, Malaysia
[3] Department of Surveying, Universiti Tunku Abdul Rahman (UTAR), Malaysia
derekisleon@gmail.com, 20013086035@stu.hqu.edu.cn, awzhenye@hotmail.com, hamzahraha@gmail.com, bhyap@utar.edu.my

Abstract

Project Management in Information System (IS) or Information Technology (IT) is one of the fastest-growing industries in developed countries. Cloud-based applications are an upcoming trend compared to on-premise software development, and a large sum of money is being poured into the industry to improve productivity and production efficiency. In Malaysia, most agricultural-based activities continue to use traditional methods that can be quite time-consuming for farmers or breeders. Thus, it can be impossible to gain accurate real-time insights needed for smart farming. Conventionally, the data collected by hand must be entered manually into computers for historical record and reference, resulting in additional efforts (rework) and inefficiency. In this study, a cloud-based Bee-Smart system was developed to optimize the maintenance and monitoring of bee farms. The core of the system is a cloud-based maintenance solution and progress report for stingless bee collection data. It not only reduces farmers' paperwork, but also gives them easy access to the historical records of each hive on a farm, based on hive activity and nectar sources in the area. The prototype was then piloted in BoDen Edu Farm where the applicability of the smart systems was proven to be effective.

Keywords: Bee farm, Sustainable production, Maintenance and monitoring system, Cloud-based system

1 Introduction

The global production of honey has seen an increasing trend as people are looking for more organic and natural sources of food and wellbeing. Stingless bees or trigona bees honey has high nutritional and therapeutics value. It has unique taste of sweet, sour, acidity and with significant "medicinal value" [1]. Even though the market for trigona honey is still in its infancy, consumers are realizing the potential and need for organic products and awareness being raised on environmental protection. Researches on stingless bees farming projects from countries like Ecuador [2], Malaysia [3], and Brazil [4] indicated that a proper guideline in maintenance and progress is vague at best, and to document every procedure could be a daunting task for every farmer, especially when the farmers are rearing a minimum of a hundred hives and most farmers averaging at five hundred hives or more, with an average of two procedures on each hive. The problem and frustration increase if the hives do not produce the desired amount of honey per month [5]. Most bee farm activities or projects using conventional methods can be quite a time consuming for farmers or breeders [6]. The use of Information System (IS) or Information Technology (IT) in agriculture is vital to improve productivity by improving the efficiency of business processes [7].

With technology ever-evolving, we are now looking to cloud-based applications that can reduce the physical storage limitations of the current hardware. This also provide avenues to further improve time, cost and quality management by implementing cloud-based applications in agriculture. Against this background, this research aims to improve the quality of data entry by relieving the burden of manual office-based data entry and replacing it with a reliable online solution. This will reduce human-based errors such as inconsistency in data entry and mis-keying information, so as to improve quality. In particular, farms may face challenges when there is a limitation in staffing. Speed and efficiency in data handling pivotal in a large-scale farm. Having accurate data is vital in maintaining the optimization of the standard operating procedure. Human frequently fails to apply judgement consistently even though they can exercise discretion in the face of ambiguity, something an automated system cannot compute in their system [8]. From operating systems to file storage, most mediums have been moved to the cloud [9]. Cloud computing is internet-based computing, where hardware and software resources are provided to users on-demand to enable remote computing [10]. With cloud-based applications, the software can be delivered through the internet on a browser without installation, host an application, set up remote file storage and database system [9]. There are five principles of cloud computing, namely: a) Resource pooling, b) Virtualization, c) Elasticity, d) Automatic/easy resource deployment, and e) Metered billing. These five principles allow users to understand the advantages of using cloud computing over conventional local system storage. The need to develop the system comes from a less productive traditional method that could reduce time and allow for optimization of work within the field of bee farming [8]. A computerized farm information system was a good investment for farms, returning 220% to 348% in investment and the factors for computer adoption include the age of farmer, their education, goals, farms size, business complexity, increased tenancy, perceptions of risk, type of production, the average expenditure on information, and use of technology by peers and other family members [10].This study developed a simple

*Corresponding Author: Yutong Tang; E-mail: 20013086035@stu.hqu.edu.cn
DOI: 10.53106/160792642022052303012

data cloud management system prototype for stingless bee farms to optimize time management, reduce cost and ultimately increase the quality of operations. The developed cloud-based prototype was validated through trial runs in a real stingless bee farm. This study could help farmers and researchers better understand why certain changes occur by collecting accurate data, analyzing changes in honey production and progress during and after maintenance, as well as to overcome these changes by analyzing the data collected by the system to form hypotheses and solutions. Population focused analytical processing via data repository would accelerate the creation of evidence. This evidence, in turn, could be made available at the point of certain periods where actions are needed to continue optimized production or necessary maintenance.

2 Recent Development on Stingless Bees and Cloud-based Farm Management

The latest development on the connection between cloud-based data applications and stingless bees is summarized. Literature from different fields of study was leading to the need of developing a cloud-based solution for stingless bee management.

2.1 Cloud-Based Applications

There are a few studies regarding cloud-based applications in agriculture [9]. The frameworks for Agri-Cloud or Management of Agricultural Data (MAD) should be developed [11-13]. The issue of bringing cloud-based technology into agriculture was raised with Fujitsu leading the pack in terms of PHP development of data management and server-based facilities [11]. The data storage from Global Position System (GPS) functionality, weather and soil data, image and audio data through mobile devices, noteworthy data extraction from routine work, and material management data with data barcode reading functions. To solve the problem of storing crop growth monitoring data safely and reliably, Zhao et al. [14] proposed an agricultural monitoring data storage method based on sustainable block chain technology. By taking advantage of cloud storage, data is encrypted and stored in the cloud to ensure data privacy and security. Singh et al. [15] proposed a cloud autonomous information system based on service awareness, to provide agriculture-related information as a service by using the latest Internet-based technologies, such as cloud computing and the Internet of Things, which manage various types of agriculture-related data based on different areas of the agricultural industry. Tan [16] developed a framework for a cloud-based decision support and automation system that captures data from a variety of sources, synthesizes application-specific decisions, and controls field equipment from the cloud, with the aim of improving agricultural operations through better data-driven. Thus, cloud-based management is becoming more widely used in the agricultural sector.

2.2 Farm Management and Information Technology

Research done by Vostrovský et al. (2015) [17] highlights the issue of knowledge support for managerial decision making of entrepreneurs with regards to knowledge management principles. The core line of the approach was the capturing of explicit knowledge relevant to given business activities into multidimensional databases that would become part of utilized ICT/IT. These issues are demonstrated in the agriculture domain where the need for the computer storage of relevant knowledge and providing them on-demand is very up to date. Recently, precision agriculture has become a frequently discussed agriculture technique.

Schneider and Wagner [18] developed an example of precision agriculture by using global positioning systems (GPS), geographical information software (GIS), and various sensors and actuators. They concluded that cost was not a factor in the implementation of precision agriculture within a farm, as both small and big farms would be able to turn a profit, and the only thing holding the farms back was the existence of efficient decision-making rules.

Kim and Yoe [19] collected environmental data information on a regional basis to cultivate crops of the best quality and then by practising the gathered information to actual farming, the study offered a greenhouse environmental data consulting system that would help the most optimized growth of crops. The system to be proposed was designed to have an environment sensor to gather data, an embedded server and HDFS to store and process data as well as a Web application to provide greenhouse control and service.

Obiniyi and Ibrahim [20] described a web-based farm management system as the collection of processes and information that is used to manage various phases of a farm and accessible on the Internet. The paper was aimed to improve the way information disseminates to farmers. Web-based farm management was needed for the development of agriculture to improve the life of farmers. The paper examined the basic visualization of a farm in 3D form for the Jigawa state farming environment. This meant that plants of the area would be viewed virtually.

Singh et al. [21] developed a model of Decision Support System for Management, keeping into account the needs of the farmers to collect important and updated information for an interactive, flexible, and quick decision-making solution. The model in question would allow online interactive access to flexible information for their farm. The solution would ultimately help farmers increase their productivity by raising the yield in food grain, which in turn, would lead to economic growth.

Xie et al. [22] proposed a big data procession technology hierarchy of agricultural information system from the aspects of gathering, storing, analysis, and visualization of the agricultural big data. An agricultural information system would be discussed based on large data technology, and showed that the methods selected proved to be effective.

2.3 Trigona Bees

Being part of the Apidae family, stingless bees have been identified as closely related to the honey bee. There were 32 species of trigona bees identified in Malaysia out of the 700 species available throughout the world [4]. Ranging from 2 to 14 mm in size, trigona bees in tropical climates are active all year round and bite as a form of defense in the appearance of predators. Some trigona bee species have self-sacrificial behaviour when there are beneficial consequences like protection of the hive as reported [23]. Trigona bees have been identified to have low mating frequency with queens and worker maternity of male trigona bees [24]. Stingless bees nest anywhere from hollow trunks, tree branches, underground cavities to storage drums, unused pipes, old rubbish bin and even water meters. Nests are closely associated with living (91.5%) and dead (8.5%) trees, either with pre-formed cavities in the trunk or under the tree base [25]. The nests were mostly found in large to very large (86.1% above 60cm dbh) commercial timber trees. They collect nectar and pollen from a huge variety of plants and are robust enough to handle the variation of temperature by regurgitating water and fanning their wings [26]. Using visual odometry, the trigona bees can estimate not only the distance travelled but also height and distance which is different from honey bees which uses optic flow-movement of images across the retina. This is advantageous to trigona bees as it allows them to be more precise and go as high as 40 meters [27].

The propolis is the product used to build the nest of the colony - found mostly in rainy tropical forest, stingless bees are not to exist in challengingly dry and hostile regions where it is difficult to find sources of concentrated sugar [28]. Honey from stingless bees is also considered beer bees, where the fermenting honey encourages the production of alcohol, and then vinegar. Four different solutions for the honey fermentation were suggested: a) Refrigeration, b) Dehydration, c) Pasteurization, and d) Maturation [2]. Bubbles in the honey suggest that ethanol is in the storage pots, but in low concentration. The honey can be used for remedial purposes in the event of sore throats, as a home remedy for stamina, and wellbeing [29]. Being a rare element in science and nature, honey from stingless bee contains a greater amount of water [6]. The sugar and water hold the nutrients together and arrange metal, secondary metabolites, microbes, chemical residue and other final products [30]. Honey can be used as food, cosmetics, and medicine. With the increasing demand for organic and natural foods, the market for triangle bees will be greatly expanded.

3 Research Methods and Procedures

The research process involved developing a website using Unified Modelling Language coupled with a cloud-based server to provide data storage facilities. A biased sampling was conducted to establish a comparison between manual and cloud-based data entry. A standard validation and debugging procedure based on the Institute of Electrical and Electronics Engineers (IEEE) was executed to validate the system codes.

3.1 Development Stages

The development approach adopted a traditional Waterfall Model endorsed by the Institute of Electrical and Electronics Engineers (IEEE). The engineering model followed a business model organization product, which in this study, would be the stingless bees. Minimizing changes to requirements and producing only a web-based solution makes the waterfall model suitable compared to other types of models endorsed by IEEE. The cloud computing service used for this prototype was software as a service (SaaS) where users completed a turnkey application through the web. This was where the software was hosted as a service in the cloud and opened using a browser once they registered for the service. This eliminated the need to install, run and maintain the application on a local computer. Authentication and authorization security policies were placed to ensure the separation of user data, which resulted in a cost reduction as well as price while ensuring the SaaS stayed competitive when compared to traditional methods using local computers. Based on the Work Breakdown Sheet (WBS) developed in Figure 1, the development began with the design followed by development, quality assurance, training, implementation, and post-implementation. The process followed closely the development model endorsed by IEEE. A software system has three main requirement categories, which are requirements for engineering, qualification and delivery. A standard signoff was performed upon completion of each segment and phase. In the design phase, focus was given to Functional Specification, SQL Database, Interfaces, and Reports. Functional Specification emphasized creating a user interface mockup and conducting a design review. This encompassed what the software would do and how the user interface would look like as well as the functions of the software. Within the SQL Database, the focus was on underlining the technical specification, development of the technical specification, and review on the technical specification. Interface designing was broken down into three segments. It was where we identified the technical specification of the interface, and determined the data import/export elements as well as defined the user needs to match with the IT needs. A design interface was finally developed to suit the needs of the system. Designing the reports encompassed the functional specification where the collection of the user's requirements by defining the data elements, frequency, and audience. The report was designed and reviewed upon completion. Delivering and signing off the final functional specification was crucial before proceeding to the development phase. In the training phase, system documentation was recorded where the assembling of the technical specification was implemented along with a system flowchart. Delivering the source code as well as a complete system documentation manual was crucial for optimized usage of the system. Creating training materials by assembling functional specification and developing "as is" and "to be" documentation was highlighted. Updating business processes and obtaining a signoff was needed before moving to the next segment. Training users by identifying trainees and trainer after constructing a training schedule. The same system was implemented for a business partner with an additional step of making them ready to use the system.

The next phase was the implementation phase where hardware and packaged software were determined, selected, purchased, and deployed. A system test was then run to verify the readiness and signoff. An implementation plan was developed by constructing a timeline, identifying a team, identifying the components, and finalizing the plan. Installation required converting the hardware and packaged software to be production-ready. After installing the program into the production environment, the codes were again verified and a limited production run was implemented for user acceptance. The system was then turned over to the user. Upon post-implementation, the system verification was conducted by obtaining user acceptance of the production system and logging issues. The system was continually monitored to verify performance and functionality.

1.1 Design

- 1.1.1 Web User Interface
- 1.1.1.1 Functional Specification
- 1.1.1.1.1 Create User Interface Mock
- 1.1.1.1.2 Conduct Design Review
- 1.1.1.1.3 Deliver Final Functional Specs
- 1.1.1.1.4 Obtain User Signoff
- 1.1.1.2 Technical Specification
- 1.1.1.2.1 Develop lech Specs
- 1.1.1.2.2 Review Tech Specs
- 1.1.1.2.3 Obtain Signoff
- 1.1.2 SQL Database
- 1.1.2.1 Technical Specification
- 1.1.2.1.1 Develop Tech Specs
- 1.1.2.1.2 Review Tech Specs
- 1.1.2.1.3 Obtain Signoff
- 1.1.3 Interfaces
- 1.1.3.1 Technical Specification
- 1.1.3.1.1 Determine data import/export elements
- 1.1.3.1.1.1 Define User Needs
- 1.1.3.1.1.2 Define IT Needs
- 1.1.3.1.2 Design Interface
- 1.1.3.1.3 Obtain Signoff
- 1.1.4 Reports
- 1.1.4.1 Functional Specification
- 1.1.4.1.1 Collect User Requirements
- 1.1.4.1.1.1 Define Data Elements
- 1.1.4.1.1.2 Define Frequency
- 1.1.4.1.1.3 Define Audience
- 1.1.4.1.2 Design Reports
- 1.1.4.1.3 Review Report Design
- 1.1.4.1.4 Deliver Final Functional Specs
- 1.1.4.1.5 Obtain Signoff

1.2 Development

- 1.2.1 Web Front End
- 1.2.1.1 Code Web Pages
- 1.2.1.2 Conduct Unit Test
- 1.2.1.3 Review Web Page Design/Functionality
- 1.2.1.4 Obtain Signoff
- 1.2.2 SQL Database
- 1.2.2.1 Identify Table Relationships
- 1.2.2.2 Build Database Tables
- 1.2.2.3 Review Tables
- 1.2.2.4 Obtain Signoff
- 1.2.3 Interfaces
- 1.2,3.1 Build Interfaces
- 1.2.3.2 Conduct Unit Test of Import/Export Functionality
- 1.2.3.3 Obtain Signoff
- 1.2.4 Reports
- 1.2.4.1 Code Reports
- 1.2.4.2 Conduct Unit Test
- 1.2.4.3 Review Reports
- 1.2.4.4 Obtain Signoff

1.3 Quality Assurance

- 1.3.1 Web Front End
- 1.3.1.1 Verify Design and Functionality
- 1.3.1.2 Perform Integration Test
- 1.3.1.3 Perform User Acceptance Test
- 1.3.1.4 Obtain Signoff
- 1.3.2 SQL Database
- 1.3.2.1 Verify Design/Data Elements
- 1.3.2.2 Verify Relationships
- 1.3.2.3 Perform Integration Test
- 1.3.2.4 Perform User Acceptance Test
- 1.3.2.5 Obtain Signoff
- 1.3.3 Interfaces
- 1.3.3.1 Verify Design and Functionality
- 1.3.3.2 Perform Integration Test
- 1.3.3.3 Perform User Acceptance Test
- 1.3.3.4 Obtain Signoff
- 1.3.4 Reports
- 1.3.4.1 Verify Design and Functionality
- 1.3.4.2 Perform Integration Test
- 1.3.4.3 Perform User Acceptance Test
- 1.3.4.4 Obtain Signoff

1.4 Training

- 1.4.1 Create System Documentation
- 1.4.1.1 Assemble Tech Specs
- 1.4.1.2 Develop System Flowchart
- 1.4.1.3 Deliver Source Code
- 1.4.1.4 Complete System Documentation Manual
- 1.4.2 Creating Training Material
- 1.4.2.1 Assemble Functional Specs
- 1.4.2.2 Develop "As is" and "To be" Documentation
- 1.4.2.3 Update Business Processes
- 1.4.2.3.1 Write New Business Processes
- 1.4.2.3.2 Obtain Signoff
- 1.4.2.4 Complete User Training Manuals
- 1.4.3 Train Users
- 1.4.3.1 Train IT Support Staff
- 1.4.3.1.1 Identify Trainees
- 1.4.3.1.2 Identify Trainers
- 1.4.3.1.3 Construct Training Schedule
- 1.4.3.1.4 Train Users
- 1.4.3.2 Train Business Partners
- 1.4.3.2.1 Identify Trainees
- 1.4.3.2.2 Identify Trainers
- 1.4.3.2.3 Construct Training Schedule
- 1.4.3.2.4 Trains Users
- 1.4.3.2.5 Verify User Readiness

1.5 Implementation

- 1.5.1 Hardware
- 1.5.1.1 Determine Hardware Needs
- 1.5.1.2 Make Hardware Selection
- 1.5.1.3 Purchase Hardware
- 1.5.1.4 Deploy
- 1.5.1.5 Perform System Test
- 1.5.1.6 Verify Production Readiness and Signoff
- 1.5.2 Packaged Software
- 1.5.2.1 Determine Software Needs
- 1.5.2.2 Make Software Selection
- 1.5.2.3 Purchase Software
- 1.5.2.4 Deploy
- 1.5.2.5 Perform System Test
- 1.5.2.6 Verify Production Readiness
- 1.5.3 Develop Implementation Plan
- 1.5.3.1 Construct Timeline
- 1.5.3.2 Identify Team
- 1.5.3.3 Identify Components
- 1.5.3.4 Finalize Plan
- 1.5.4 Installation
- 1.5.4.1 Convert Hardware to Production-Ready Status
- 1.5.4.2 Convert Packaged Software to Production Ready Software
- 1.5.4.3 Install New Programs into Production Environment
- 1.5.4.4 Verify Codes
- 1.5.4.5 Initiate Limited Production Run for User Acceptance
- 1.5.4.6 Turn over System to Users

1.6 Post Implementation

- 1.6.1 Verify System
- 1.6.1.1 Obtain User Acceptance of Production System
- 1.6.1.2 Log Issues
- 1.6.2 Monitor System
- 1.6.2.1 Verify Performance
- 1.6.2.2 Verify Functionality
- 1.6.3 Project Wrap Up
- 1.6.3.1 Obtain Final Project Sign Off
- 1.6.3.2 Document and Review Lesson Learned

Figure 1. Work breakdown sheet

The next phase was the implementation phase where hardware and packaged software were determined, selected, purchased, and deployed. A system test was then run to verify the readiness and signoff. An implementation plan was developed by constructing a timeline, identifying a team, identifying the components, and finalizing the plan. Installation required converting the hardware and packaged software to be production-ready. After installing the program into the production environment, the codes were again verified and a limited production run was implemented for user acceptance. The system was then turned over to the user. Upon post-implementation, the system verification was conducted by obtaining user acceptance of the production system and logging issues. The system was continually monitored to verify performance and functionality.

3.2 Design Flow

The general system flow was developed in Figure 2. Post answers would be compiled by the system, and once the compilation was completed, the system would process the report, which would then be published to be viewed by the users.

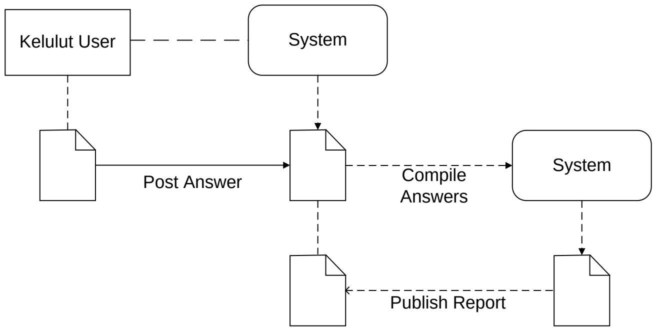

Figure 2. General system flowchart

The reporting flow for the manager was defined in Figure 3. The manager would start by creating a set, in which he or she would enter the type of questions. If the question was not complete, a notification would be delivered. Once the form was completed, the process completed.

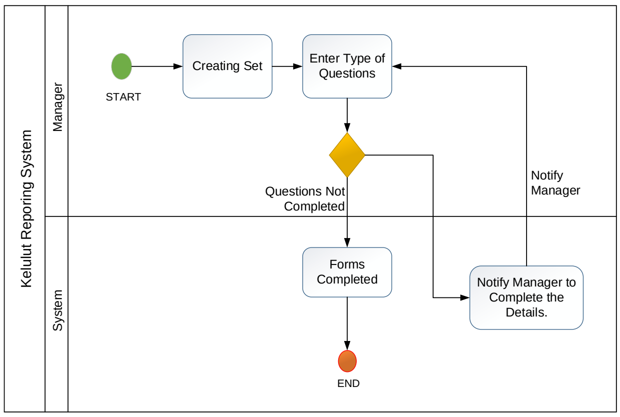

Figure 3. Manager reporting flow

On the other hand, for the user, the flowchart in Figure 4 allowed the user to fill in the forms and to answer the questions set by the manager.

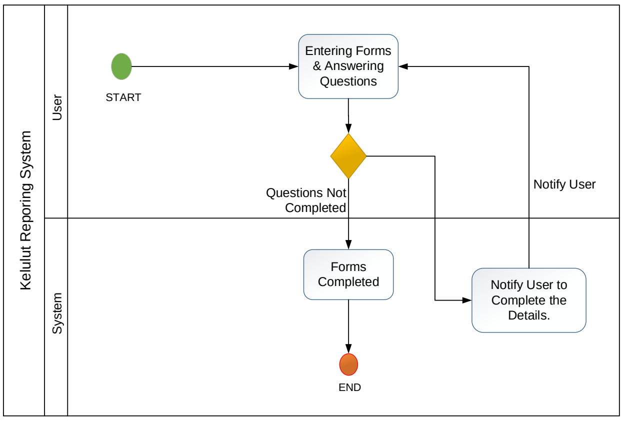

Figure 4. User reporting flow

The reporting system as highlighted in Figure 5 shows that the manager would create the questionnaire, where the system would process it and publish it at the end of the flow. The user would then answer the survey processed by the system.

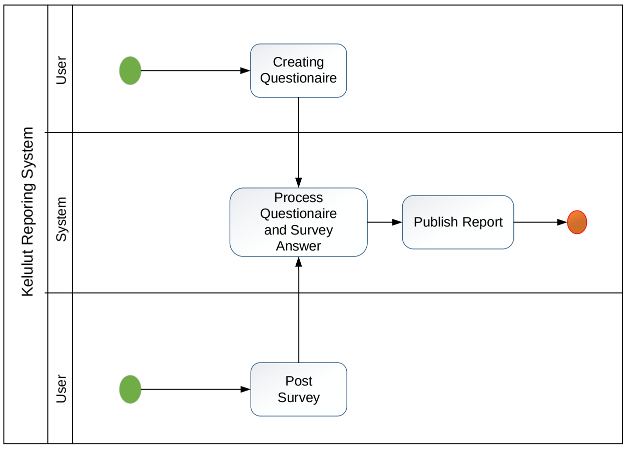

Figure 5. Manager and user reporting system

3.3 Trial Run for Bee-Smart System

A cloud-based system for sustainable stingless bee farm maintenance and monitoring named "Bee-Smart System" was developed in this study, and its trial run was conducted in a stingless bee farm in Serting Ilir, Negeri Sembilan, Malaysia. The area within the farm was a combination of flats and a highland environment. The weather was mostly dry with temperature between 28 to 42 degrees Celsius. The farm is one hundred acres in size with a hundred hives spanning an area of fifty acres. The trigona bee division was assigned two staff on hand to take care of the twelve different trigona bee species hives. The farm followed standard operating procedures to suit the farm's strict guidelines of biosecurity in order to obtain optimized results. The hives were divided into 3 quadrants as follows: a) Quadrant 1 focused on multiple flowers and plants with herbal and medicinal values; b) Quadrant 2 had hives in an environment with multiple flowers and fruit trees; and c) Quadrant 3 is an area with an abundance of flower and plants but with the emphasis on research and development. The development quadrant involved developing a box hive that maintained the same core temperature as the log, use of water under each hive to hinder pests like ants or lizards, and difference in honey production from hives in hot and cool areas. Data logging was performed both manually and Bee-Smart System alternately every 1-3 weeks depending on the availability of honey collection and maintenance work. Once enough data were collected, a paired T-Test was executed to evaluate the significance of the Bee-Smart System when compared with manual data collection. The second paired T-Test was to provide a comparison between the total cost of ownership (TCO) between on-demand and on-premise systems. The consideration of calculation was set up to 5 years. A third paired T-Test was then conducted to provide validation on the quality of usability based on the system design, where a biased sampling of twelve individuals, 8 workers, and 4 stingless bee farmers were selected to test the Bee-Smart system and the manual system. A System Usability Scale (SUS) was used to gather information.

4 Appearance of Bee-Smart Cloud System

A detailed description of the appearance of Bee-Smart was based on the needs and requirements in developing the cloud-based solution for stingless bee management. The wireframe was designed to provide a clearer picture of how the system was developed to be more user-friendly. Data collection requirements for maintenance and monitoring, as well as the honey collection, were identified.

4.1 User Requirements for System Environment

Figure 6 highlights the modular for the high-level view of the system. It provides information for users and administrators. The design map shows the capabilities and authorization for each level. All levels could access the dashboard tiles, news, notification, chart reports, and schedules.

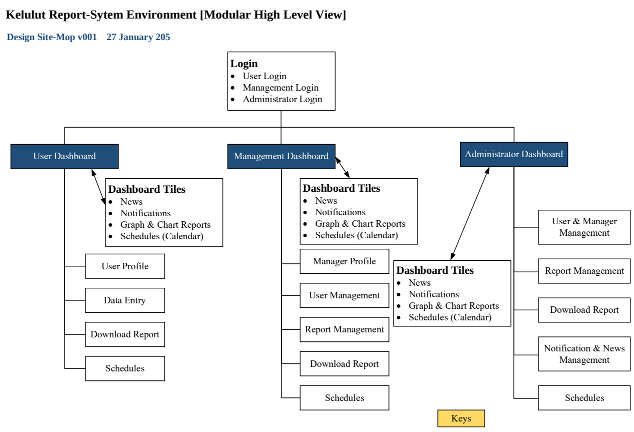

Figure 6. Modular high-level view of the system environment

Figure 7 identifies that the user is only allowed to view and edit their personal information in the user profile. Users are responsible for data entry based on the questions set by the manager. Viewing and downloading reports are also allowed for the user level. Schedule notifications could be received and viewed between user and manager.

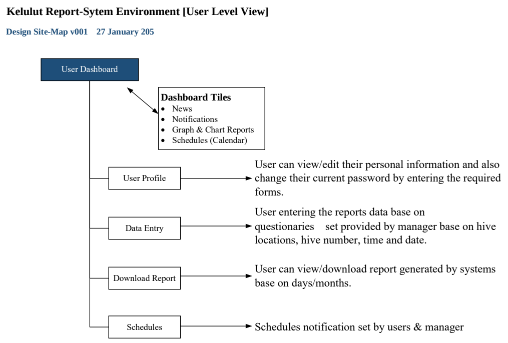

Figure. 7. User-level view

The management-level view in Figure 8 allowed creating, editing, or deleting user-level and manager-level profiles. Management-level could also view and download reports. Schedules are assigned to users for execution.

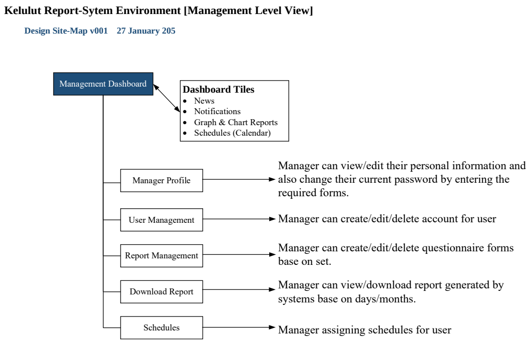

Figure 8. Management-level view

The administrator level in Figure 9 can create, edit or delete the account for user and manager as well as questionnaire forms. Administrators are allowed to publish news and notification updates on user and manager profiles. Administrators can also view, edit or even delete schedules assigned to users and managers.

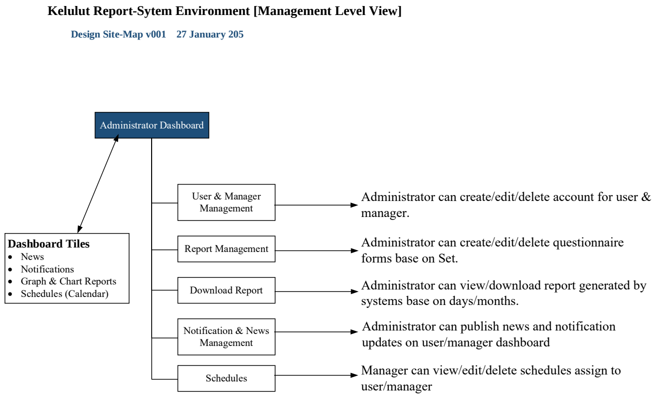

Figure 9. Administrator-level view

4.2 User Interface

The system had two separate reports. One report was on the maintenance and progress report, while the other focused on honey collection for each trigona beehive. The main menu is located at the top of the dashboard for easy access as highlighted in Figure 10. The schedule highlighted dates of data entry for each report submitted by users, while the news featured information to be relayed to other users of the system besides the administrator. An add-and-edit record was provided to the administrator to allow users to add more records using the calendar provided in the first columns as seen in Figure 11. The second column featured all existing data from the last data entry. In this prototype, the maximum number of trigona hives that could be collected was fifty.

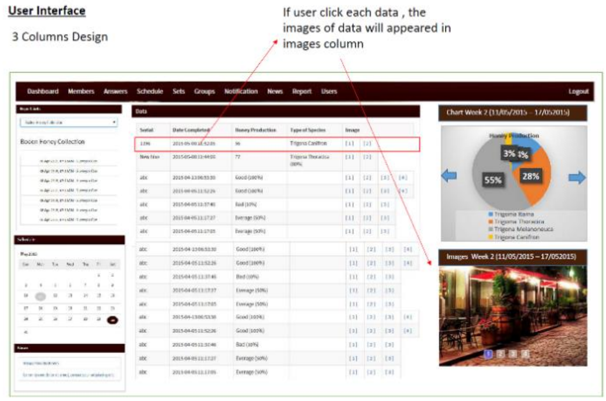

Figure 10. User interface

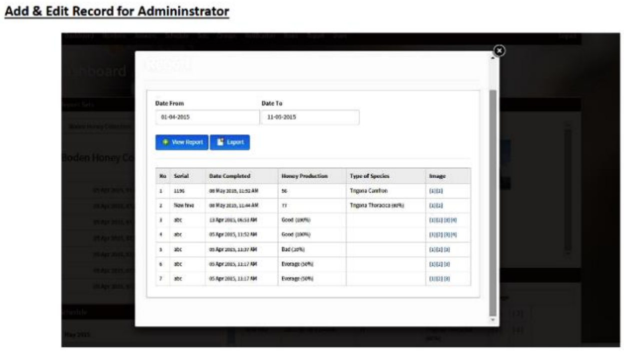

Figure 11. Add-and-edit record for administrator

The third column highlighted a weekly pie chart and images of the hives selected within the data. The second and the third columns were linked whereby if a user clicked on a particular data, the images of that particular trigona hive could be displayed in the images section using the Jquery banner script to show pictures more interactively. The pie chart generated data based on the type of trigona bees, followed by the honey that was collected for that particular week. Using color-coding, the pie chart displayed the amount of honey in percentages to highlight how much honey was collected from each species as seen in the wireframe provided in Figure 12.

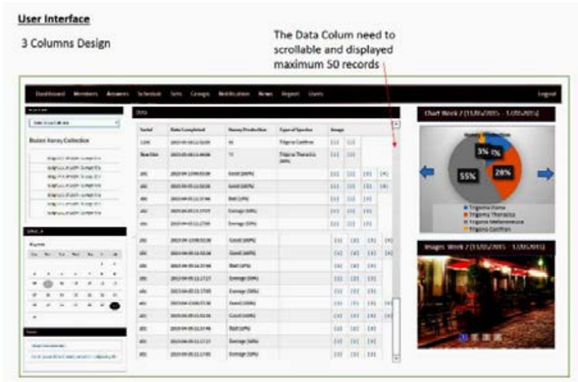

Figure 12. Three-column design

4.3 Maintenance and Progress Report

The system was divided into two different pages. One page was for the administrator and experienced users while the other was specifically designed for those less knowledgeable workers with simple data entry instructions to minimize error. The data entry required was divided into 4 sections, including: Information on the Hive, Nature of the Problem, Service Details, and Progress Update. A Maintenance and Progress Report Number (MPR) was issued for the ease of tracking, along with the date and pre- maintenance and post-maintenance photos of the hive for referencing. The service detail segment highlighted what was done to rectify the hive through service rendered, remarks on the issues faced during maintenance, the status of the hive after service, and defects found during post-inspection. The progress update section was an open-ended remark box to allow users to update the progress of the said hive whether it was recovering well or more maintenance was required to save the trigona hive.

4.4 Honey Collection Database

The honey collection database consisted of the date of collection, the type of species that the honey was collected from, the amount of honey collected, the serial number, and optionally a photo of the hive collected, as highlighted in Figure 13. The data collection plotted in Figure 14 presents different frequencies of input depending on the area being collected so as to not disturb the daily routines of the bees. The estimated data collection frequency was every two to three weeks once.

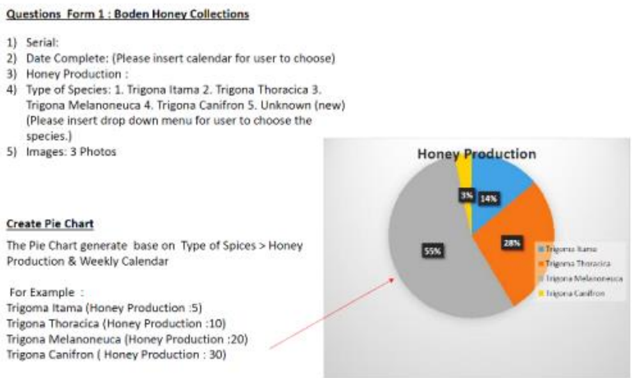

Figure 13. Report questions for honey collection and pie chart (source: author derived)

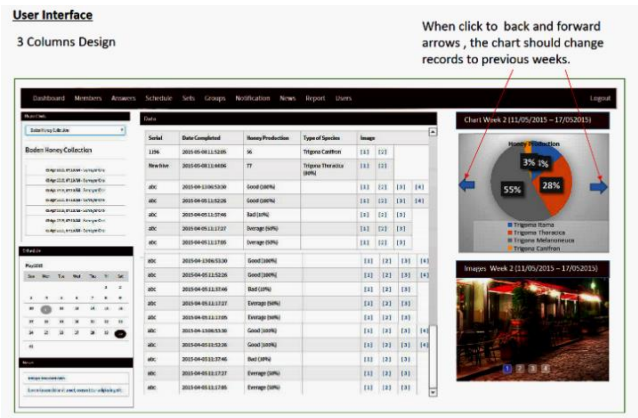

Figure 14. Honey collection database (source: author derived)

4.5 Development Requirements

The Bee-Smart prototype was developed taking into account: a) the needs of stingless bee farming; b) cloud-based solution rules, and c) emphasis on a user-friendly graphic user interface. The system requirements for development were: a) Linux/freebsd for the operating system; b) MYSQL for the database; c) PHP for the scripting language; and d) APACHE for the webserver. The debugging and validation were conducted by the system tester and the consultant. The team members were listed in Table 1.

Table 1. Bee-Smart developing team

Team	Age	Gender	Education	Nationality	Expertise	Role
Member 1	37	Male	Biologist e	Malaysian	Stingless Bees	Project Manager/ Developer
Member 2	37	Female	Computer Science	Chinese	Computer Science	System Consultant/ Tester
Member 3	64	Male	Project Management	Australian	Farm Management	Management Consultant
Member 4	39	Male	Construction Management	Indonesian	Farm Management	Field Tester

4.6 Case Study for Bee-Smart in BoDen Edu Farm

The case study was located in BoDen Edu Farm as in Figure 15 that situated at Lorong BoDen, Jalan Menteri Besar, Serting Ilir, Negeri Sembilan, Malaysia, which used to be the first goat farm to produce its own lineage of boer goats in Malaysia. Spanning an area of one hundred acres, the farm decided to expand into a new segment of agriculture, trigona bee farming. There were currently more than a hundred hives spanning an area of fifty acres.

Figure 15. BoDen Edu Farm (captured by author)

The company had an additional one hundred and fifty hives in two other locations such as Kuala Pilah and Tanjung Malim. This farm was chosen as the testing and sampling area because of the openness of the management to allow for research being implemented in the farm. The farm provides high-quality sampling data collection and accuracy, and this study was conducted using a practice-driven approach, so the possibility of sampling contamination is minimal. A total of twelve species of trigona bees were found within the farm. Six species were identified for testing to optimize honey production and multiplication, with another six species spread throughout the farm for pollination optimisation.

5 Validation and Trial Run

5.1 Validation

Six species described in Figure 16 were used as bias samples to facilitate verification and authentication of the optimization of the Bee-Smart system. The six species were: a) Trigona itama (35 hives); b) Trigona thoracica (5 hives); c) Trigona apicalis (2 hives); d) Trigona canifron (2 hives); e) Trigona melanoneuca (2 hives); and f) New species (1 hive).

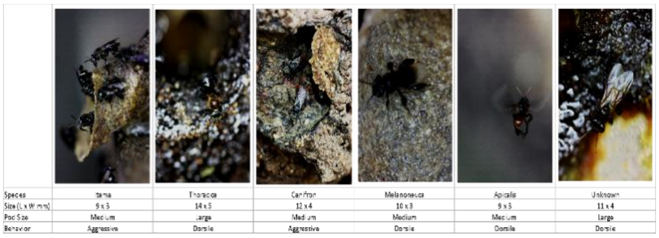

Figure 16. Species selected as bias samples from BoDen (captured by author)

Forty-seven hives in total were involved in the testing and validation. A biased sampling for areas was conducted where the farm was divided into three areas such as the Quadrant One (Island area), the Quadrant Two (Office area), and the Quadrant Three (Research and Development area). The Quadrant one was then divided into smaller areas to allow for easy identification: a) Four areas, parking (area one); b) Big tree (area two); c) Phaleria macrocarpa or God's Crown (area three); and d) Gazebo (area four). The other quadrants were not divided as the hives were all placed every three to five meters in the trees.

5.2 Trial Run of Bee-Smart System

The reduction of data input time was significant to warrant the continuous usage of the Bee-Smart system. Using conventional systems, a total of six hours and thirty minutes was needed to manually record data with an additional two hours for data entry for printing and storing purposes. On the other hand, the Bee-Smart prototype only required a duration between two hours and forty-five minutes to three hours for data entry without the need for additional hours to key in and store data for printing. A dependent T-test was run to determine the significance of the system based on the time taken to input data into the honey collection database. The two-tailed P value was less than 0.01 indicating that the variations between data were considered statistically significant, which proved the reduction in data logging time was significant enough that the prototype could effectively optimize the farm management efficiency. The confidence interval for the test was calculated at 95% from 215.44 to 227.06. The mean value for Manual Collection minus Cloud Collection was equal to 221.25. The Intermediate Values used in the Calculation were $t = 90.1240$, $df = 7$, and standard error of difference = 2.1455. The data are presented in Table 2.

Table 2. Time taken for data collection

Date	Start	End	Added Hours	Total Hours (h)	Total Min (s)
Manual					
5/2/2015	09:30:00	14:00:00	2	6.5	390
26/2/2015	09:30:00	14:00:00	2	6.5	390
19/3/2015	09:30:00	14:00:00	2	6.5	390
2/4/2015	09:30:00	14:00:00	2	6.5	390
16/4/2015	09:30:00	14:00:00	2	6.5	390
7/5/2015	09:30:00	14:00:00	2	6.5	390
14/5/2015	09:30:00	14:00:00	2	6.5	390
11/6/2015	09:30:00	14:00:00	2	6.5	390
Cloud					
29/1/2015	09:30:00	12:30:00	0	3	180
12/2/2015	09:30:00	12:30:00	0	3	180
12/3/2015	09:30:00	12:15:00	0	2.75	165
26/3/2015	09:30:00	12:15:00	0	2.75	165
9/4/2015	09:30:00	12:15:00	0	2.75	165
23/4/2015	09:30:00	12:15:00	0	2.75	165
30/4/2015	09:30:00	12:15:00	0	2.75	165
4/6/2015	09:30:00	12:15:00	0	2.75	165

5.3 Total Cost of Ownership (TCO)

The total cost of ownership (TCO) of Bee-Smart was compared with the On-Premise software for five years indicating that TCO was more cost-efficient in the long run with a difference of $9,417.00 as in Table 3, Table 4 and Figure 17. The SaaS total cumulative spending for five years was $7,480.00 while On-Premise software required $16,897.00. The high cost of On-Premise software was mostly due to cumulative hardware upgrading requirements in the first and fifth year with estimated repair/maintenance cost between year two and five. Bee-Smart did not require regular software and hardware upgrading as it was a cloud-based system. The major cost for Bee-Smart was the development and customization to cater to the specific need with a minimal subscription fee. The two-tailed P value from the paired t-test was less than 0.05 indicating that the variation was statistically significant. The confidence interval for the test was 95% from 177.38 to 9443.02. The mean value of Manual Collection minus Cloud Collection was equal to 4810.20. The Intermediate Values used in the Calculation were $t = 2.8827$, $df = 4$, and standard error of difference = 1688.617.

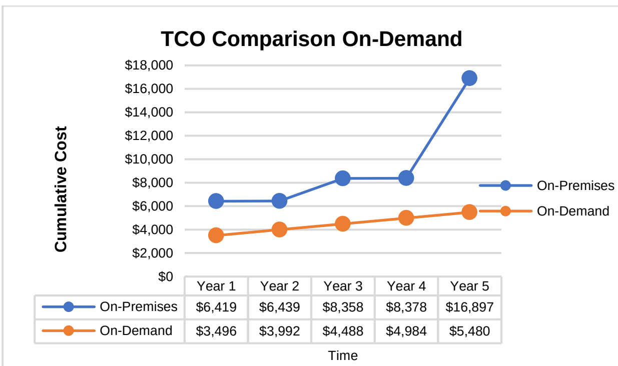

Figure 17. Total cost of ownership comparison graph

Table 3. Total cost of ownership calculation

	Cumulative Costs													
	Year 1	Year 2	Year 3	Year4	Year 5									
On-Demand	$3,496	$3,992	$4,488	$4,984	$5,480									
On-Premises	$6,419	$6,439	$8,358	$8,378	$16,897									
	On Demand/Software-As-A-Service (SaaS)							**On-premises**						
	Year 1	Year 2	Year 3	Year 4	Year 5									
Number Of Users	1	1	2	2	2									
Subscription Fee Per Month	$0	$0	$0	$0	$0									
	Year 1	Year 2	Year 3	Year 4	Year 5	Total	% of total	Year 1	Year 2	Year 3	Year 4	Year 5	Total ($)	% of total
Hardware/Infrastructure														
Servers	$0	$0	$0	$0	$0	**$0**	**0.0%**	$4,500	$-	$-	$-	$4,500	**$9,000**	**53.3%**
Peripherals	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Network	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Other	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
…								$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Hardware/ Infrastructure Costs	$0	$0	$0	$0	$0	**$0**	**0.0%**	$4,500	$-	$-	$-	$4,500	**$9,000**	**53.3%**
Communication														
Local Area Network	$20	$20	$20	$20	$20	**$100**	**1.8%**	$20	$20	$20	$20	$20	**$100**	**0.6%**
Wide Area Network	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Remote Access	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
…	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Communication Costs	$20	$20	$20	$20	$20	**$100**	**1.8%**	$20	$20	$20	$20	$20	**$100**	**0.6%**
Software														
License/ Subscription Fees	$476	$476	$476	$476	$476	**$2,380**	**43.4%**	$1,899	$-	$-	$-	$-	**$1,899**	**11.2%**
…								$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Software Costs	$476	$476	$476	$476	$476	**$2,380**	**43.4%**	$1,899	$-	$-	$-	$-	**$1,899**	**11.2%**
Implementation														
Development/ Customization	$3,000	$-	$-	$-	$-	**$3,000**	**54.7%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Training	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Consulting	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Other	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
…	$-	$-	$-	$-	$-	**$-**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Implementation Costs	$3,000	$-	$-	$-	$-	**$3,000**	**54.7%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Management/Maintenance														
Hardware & Software Upgrades	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$1,899	$-	$3,999	**$5,898**	**34.9%**
Hardware & Software Administration	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Other	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
…							**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Management Costs	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$1,899	$-	$3,999	**$5,898**	**34.9%**
Support														
Support Staff	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Staff Training	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Travel	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Support Contracts	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Overhead Labor	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
…								$-	$-	$-	$-	$-	**$-**	**0.0%**
…								$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Support Costs	$0	$0	$0	$0	$0	**$0**	**0.0%**	$-	$-	$-	$-	$-	**$-**	**0.0%**
Total Costs	$3,496	$496	$496	$496	$496	$5,480		$6,419	$20	$1,919	$20	$8,519	$16,897	

Table 4. Total cost of ownership comparison

Method	Manual	Cloud
Mean	9238.20	4428.00
SD	4330.86	752.62
SEM	1936.82	336.58
N	5	5
P value and statistical significance two tailed P value		<0.0449
Confidence interval mean of manual minus cloud		-4810.20
95% confidence interval of this difference		-177.38 to -9443.02
Intermediate values used in calculation t value		2.8827
DF		4
Standard error of difference		1688.617

5.4 Software Usability Scale (SUS)

The software usability scale (SUS) was used to measure the quality of the developed Bee-Smart prototype compared to those conventional manual data-entry systems. In the paired t-test, the two-tailed P value was less than 0.01 considered statistically significant. The mean value of Manual minus Cloud-Based equaled to -34.867 with a 95% confidence interval from -42.486 to -27.248 as in Table 5 and Table 6, where t = 11.7640, df = 5, and standard error of difference = 2.964.

The prototype was used every one to three weeks depending on the amount of work done on the hives and honey availability. A four-month sampling and validation process was executed to justify usage and to eliminate any bugs found. The time of start and finish were recorded. For maintenance and progress reports, a total of eleven hives were under observation. The honey collection data entry was conducted once every one to two weeks.

Table 5. Software usability scale comparison

SUS Calculation											
Manual Participant	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	SUS Score
p1	3	2	3	2	3	3	2	2	2	3	52.5
p2	2	2	2	2	3	2	2	2	2	4	47.5
p3	3	2	2	2	4	2	2	3	3	2	62.5
p4	3	2	3	2	3	2	2	3	3	3	60.0
p5	2	2	2	2	3	3	3	2	2	3	45.0
p6	2	1	3	3	3	2	2	3	3	3	55.0
											53.8
Cloud Based Data Application											
p7	4	1	4	2	4	1	4	1	3	1	82.5
p8	4	1	5	1	3	1	4	1	4	1	87.5
p9	4	1	5	1	4	1	4	1	4	1	90.0
p10	5	1	4	2	4	1	5	2	4	1	87.5
p11	5	1	5	2	5	1	5	2	3	2	87.5
p12	5	1	5	1	4	1	5	1	5	1	97.5
											88.8

Table 6. Software usability scale

Method	Manual	Cloud
Mean	53.883	88.750
SD	6.883	4.937
SEM	2.810	2.016
N	6	6

P value and statistical significance	
Two tailed P value	<0.001
Confidence interval	
Mean of manual minus cloud	-34.867
95% confidence interval of this difference	-42.486 to -27.248
Intermediate values used in calculation	
T	11.7640
DF	5
Standard error of difference	2.964

Figure 18 shows the designed login page for administrator and member. It is a simplified login page with a picture and login box for easy downloading.

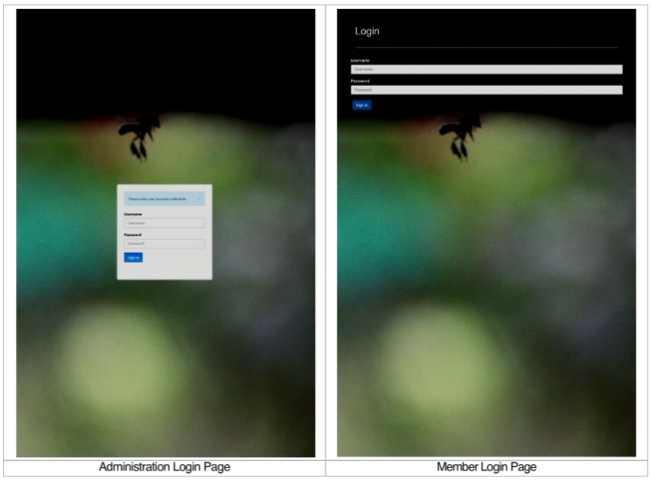

Figure 18. Login page for administration and user (source: author derived)

Figure 19 depicts the administrator's main page, and maintenance and progress report page with sample reports highlighted. A simple background picture was shown for aesthetic purposes while maintaining a three-column design discussed in the methodology.

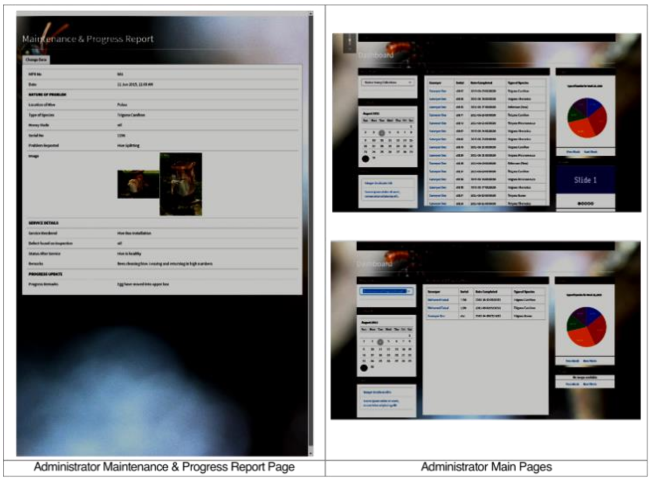

Figure 19. Administrator maintenance and progress report and main page (source: author derived)

By using the system, data inputting was easily done on a smartphone, and tablet device while each sample hive is inspected for the quantity of honey and overall health of each hive. Using the prototype allowed the data to be immediately arranged according to the serial number, and honey collected from each hive. Member's data entry pages are introduced in Figure 20.

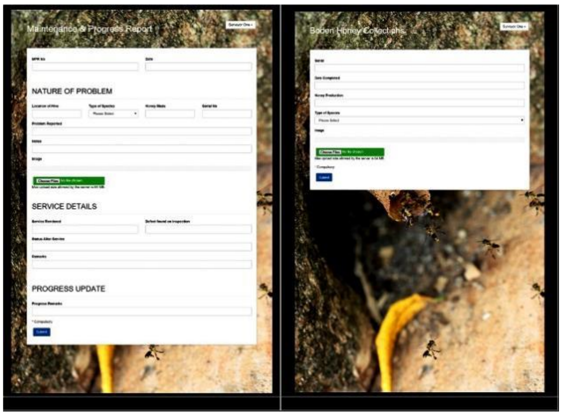

Figure 20. Member data entry pages (source: author derived)

5.5 Test Run of Prototype in Real Life Situation Observation

The test run lasted for half a year whereby the maintenance and progress report could run smoothly even with minimal internet signals between a network connection of EDGE and 3G connectivity. In the test, the size of photo packages ranged from 100MB-140MB and the size of hive packages ranged from 700MB-900MB. The fluctuation of the network would vary depending on which area testing was done within the farm. The time taken to write up a pre and post report was less than one minute while uploading data to the cloud would take an additional minute at the longest during the slowest network connectivity. An average time for maintenance per hive was between thirty minutes to two hours depending on the complexity of the maintenance required. Photo uploading proved to be the most challenging when minimal to low internet connection is available due to the large file size. With good internet connection, a photo can be uploaded in ten seconds. However, it took up to a maximum of 5 minutes per photo when the area has a limited internet connection. Training users without any IT background proved to be a non-issue due to the simplicity of the data input interface being an advantage. Users were able to familiarize themselves with the prototype within five minutes. Feedbacks from users were also positive that there was no need to look for the serial numbers of hives they are checking, which allowed them to easily type in each hive number or to add photos for a new hive without trouble. Users could analyze and develop observational interpretations on current issues and could find patterns in the trigona bee activities such as the estimation of honey production or periods for optimal hive splitting. The prototype allowed users to find out why certain hives were not performing and provided an eagle's view to estimate possible solutions for resource flow. The sharp drop at the end of March and early January was the result of environmental selection when this area was experiencing a drought and natural selection where the eggs were starting to hatch in the hives and the bees were focusing on collecting pollen for most of the day. Assisted by the Bee-Smart prototype, users were able to set up flowing water facilities for the trigona bees to have a constant flow of aqua during the drought and injuring trees with rubber resin for the bees to collect to create pods for the pollen. Introducing potted plants to optimize pollen collection allowed the trigona bees to reduce the amount of time needed to collect pollen. When they have enough pollen, the trigona bees would go out looking for honey. A high spike in honey was a result of a moderate amount of rainfall in the vicinity, which provided cooling and water sources for the honey. This was interpreted as a warning to users that the honey could have a high-water content that could activate the yeast and enzymes within the honey resulting in the sucrose turning to alcohol, and eventually to vinegar. The records showed that the most productive trigona bee was the trigona thoracica instead of trigona itama. Although the trigona thoracica had fewer hives, they were able to produce more honey than trigona itama did. This data could be utilized by a user to focus their efforts on splitting trigona thoracica hives first to allow for multiplication operation when searching for new hives in the forest. It was noted that using the system on a smartphone provided users more flexibility without the need to bring a tablet computer to key in data. Photos could be uploaded using the cell phone camera.

6 Conclusion

This study identified the cloud-based data logging requirements for stingless bee management and developed a cloud-based Bee-Smart system for sustainable stingless bee farm maintenance and monitoring. A simple 3-row web design was created with drop-down menus and open-ended notes to facilitate the ease of use within an area of limited data connection. The maintenance and progress report function allowed the user to log noticeable problems of hives and the rectification actions taken to solve the issues. The honey collection data category focused on the development of a simple data logging solution for honey collection to reduce the need for physical folders or manual data collection. The data collected would highlight how much honey was collected by each species of trigona bees. Individual data of each hive was also made available to the management level that was collected using a simple mobile version on a smartphone or tablet provided to the workers. The Bee-Smart prototype was coded and tested in a real bee farm to validate through actual working conditions for half a year. Bias sampling using forty-seven hives from 6 different trigona bee species was conducted within 6 months to establish credible data collection. A pattern was seen to emerge within the behavior of the trigona bees, based on which, users were able to gauge estimated honey production of the trigona bees during droughts and rainy seasons, to analyze the reason why certain hives were not producing at an optimal level, to judge whether the drop was due to behavioral or environmental changes, and to evaluate the quality of honey produced during rainy seasons when the water content in the honey was high thus measures could be taken to stop the enzymes from turning the honey into alcohol and ultimately into vinegar. Data logging and interpretation is crucial in positioning a bee farm to achieve optimal outputs with low cost. The prototype developed in this study to integrate a cloud-based application within the compounds of trigona bee management is an exciting venture that should be explored further. Advancing the current prototype to include more information such as the addition of humidity levels and correlating those levels with the moisture content of trigona honey could help create a proper baseline for trigona honey quality. Future possibilities could also allow for the correlation of sucrose level and flower species. Individual hive performance would also be a good direction to go in a final production application. The possibility of executing an automatic serial number creator could be a valuable addition to the prototype as there would always be new hives join into the farm.

References

[1] F. C. Lavinas, E. H. B. C. Macedo, G. B. L. Sa, A. C. F. Amaral, J. R. A. Silva, M. M. B. Azevedo, B. A. Vieira, T. F. S. Domingos, A. B. Vermelho, C. S. Carneiro, I. A. Rodrigues, Brazilian stingless bee propolis and geopropolis: promising sources of biologically active compounds, *Revista Brasileira De Farmacognosia-Brazilian Journal of Pharmacognosy*, Vol. 29, No. 3, pp. 389-399, May-June, 2019.

[2] V. Patricia, V. Oliverio, L. Triny, M. Favian, Meliponini biodiversity and medicinal uses of pot-honey from EI Oro province in Ecuador, *Emirates Journal of Food and Agriculture*, Vol. 27, No. 6, pp. 502-506, June, 2015.

[3] N. C. Soh, N. S. Samsuddin, M. M. Ismail, M. S. Habibullah, Technical Efficiency of Commercial Stingless Bee Honey Production in Peninsular Malaysia, *Pertanika Journal of Social Science and Humanities*, Vol. 29, No. 2, pp. 785-797, June, 2021.

[4] L. F. Wolff, J. C. Costa Gomes, Beekeeping and Agroecological Systems for Endogenous Sustainable Development, *Agroecology and Sustainable Food Systems*, Vol. 39, No. 4, pp. 416-435, 2015.

[5] A. Nordin, N. Omar, N. Q. A. V. Sainik, S. R. Chowdhury, E. Omar, A. Bin Saim, R. Bt Hj Idrus, Low dose stingless bee honey increases viability of human dermal fibroblasts that could potentially promote wound healing, *Wound Medicine*, Vol. 23, pp. 22-27, December, 2018.

[6] F. C. Biluca, B. da Silva, T. Caon, E. T. Bramorski Mohr, G. N. Vieira, L. V. Gonzaga, L. Vitali, G. Micke, R. Fett, E. M. Dalmarco, A. C. Oliveira Costa, Investigation of phenolic compounds, antioxidant and anti-inflammatory activities in stingless bee honey (Meliponinae), *Food Research International*, Vol. 129, Article No. 108756, March, 2020.

[7] S. Gawade, V. Turkar, A Role and Potential of E-Krishimitra Tool in Usability Improvement of Agricultural Domain, *International Journal of Computer Applications*, Vol. 178, No. 9, pp. 6-12, May, 2019.

[8] M. D. Byrne, T. R. Jordan, T. Welle, Comparison of Manual versus Automated Data Collection Method for an Evidence-Based Nursing Practice Study, *Applied Clinical Informatics*, Vol. 4, No. 1, pp. 61-74, February, 2013.

[9] M. Wang, Q. Zhang, Optimized data storage algorithm of IoT based on cloud computing in distributed system, *Computer Communications*, Vol. 157, pp. 124-131, May, 2020.

[10] D. Doye, R. Jolly, R. Hornbaker, T. Cross, R. P. King, W. F. Lazarus, L. A. Yeboah, Case Studies of Farmer's Use of Information Systems, *Review of Agricultural Economics*, Vol. 22, No. 2, pp. 566-585, October, 2000.

[11] M. Hori, E. Kawashima, T. Yamazaki, Application of Cloud Computing to Agriculture and Prospects in Other Fields, *Fujitsu Scientific & Technical Journal*, Vol. 46, No. 4, pp. 446-454, October, 2010.

[12] F. S. Nininahazwe, J. Shen, M. E. Taylor, An Augmented Load-Balancing Algorithm for Task Scheduling in Cloud-Based Systems, *Journal of Internet Technology*, Vol. 22, No. 7, pp. 1457-1472, December, 2021.

[13] K. Chandraul, A. Singh, An Agriculture Application Research on Cloud Computing, *International Journal of Current Engineering and Technology*, Vol. 3, Article No. 5, pp. 2084-2087, June, 2013.

[14] Z. Liu, W. Huang, D. Wang, Functional agricultural monitoring data storage based on sustainable block chain technology, *Journal of Cleaner Production*, Vol. 281, Article No. 124078, January, 2021.

[15] S. Singh, I. Chana, R. Buyya, Agri-Info: Cloud Based Autonomic System for Delivering Agriculture as a Service, *Internet of Things*, Vol. 9, Article No. 100131, March, 2020.

[16] L. Tan, Cloud-based Decision Support and Automation for Precision Agriculture in Orchards, *IFAC-PapersOnLine*, Vol. 49, No. 16, pp. 330-335, 2016.

[17] V. Vostrovský, J. Tyrychtr, M. Ulman, Knowledge Support of Information and Communication Technology in Agricultural Enterprises in the Czech Republic, *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis*, Vol. 63, No. 1, pp. 327-336, February, 2015.

[18] M. Schneider, P. Wagner, Prerequisites for the Adoption of New Technologies: the Example of Precision Agriculture, *Agricultural Technique*, Vol. 32, No. 2, pp. 9-14, April, 2007.

[19] S. Kim, H. Yoe, Design of Big Data-based Greenhouse Environment Data Consulting System for Improving Crop Quality, *Advanced Science and Technology Letters*, Vol. 95, pp. 182-187, 2015.

[20] A. Obiniyi, A. Ibrahim, A Web-Based Farm 3D Visualization Management System, Journal of Computer Science & Systems Biology, Vol. 8, No. 1, pp. 49-54, 2015.

[21] M. Singh, P. Singh, S. B. Singh, Decision Support System for Farm Management, *World Academy of*

Science Engineering and Technology, *International Journal of Agricultural and Biosystems Engineering*, Vol. 2, No. 3, pp. 59-62, 2008.

[22] N. F. Xie, X. F. Zhang, W. Sun, X. N. Hao, Research on Big Data Technology-Based Agricultural Information System, *Proceedings of the International Conference on Computer Information Systems and Industrial Applications*, Bangkok, Thailand, 2015, pp. 388-390.

[23] K. Shackleton, H. Al Toufailia, N. J. Balfour, F. S. Nascimento, D. A. Alves, F. L. W. Ratnieks, Appetite for self-destruction: suicidal biting as a nest defense strategy in Trigona stingless bees, *Behavioral Ecology and Sociobiology*, Vol. 69, No. 2, pp. 273-281, February, 2015.

[24] R. J. Paxton, L. R. Bego, M. M. Shah, S. Mateus, Low mating frequency of queens in the stingless bee Scaptotrigona postica and worker maternity of males, *Behavioral Ecology and Sociobiology*, Vol. 53, No. 3, pp. 174-181, February, 2003.

[25] T. Eltz, C. A. Bruhl, Z. Imiyabir, K. E. Linsenmair, Nesting and nest trees of stingless bees (Apidae: Meliponini) in lowland dipterocarp forests in Sabah, Malaysia, with implications for forest management, *Forest Ecology and Management*, Vol. 172, No. 2-3, pp. 301-313, January, 2003.

[26] J. O. Macias-Macias, J. J. G. Quezada-Euan, F. Contreras-Escareno, J. M. Tapia-Gonzalez, H. Moo-Valle, R. Ayala, Comparative temperature tolerance in stingless bee species from tropical highlands and lowlands of Mexico and implications for their conservation (Hymenoptera: Apidae: Meliponini), *Apidologie*, Vol. 42, No. 6, pp. 679-689, November, 2011.

[27] M. A. Eckles, D. W. Roubik, J. C. Nieh, A stingless bee can use visual odometry to estimate both height and distance, *Journal of Experimental Biology*, Vol. 215, No. 18, pp. 3155-3160, September, 2012.

[28] É. W. Teixeira, E. A. Ferreira, C. F. P. D. Luz, M. F. Martins, T. A. Ramos, A. P. Lourenço, European Foulbrood in stingless bees (Apidae: Meliponini) in Brazil: Old disease, renewed threat, *Journal of Invertebrate Pathology*, Vol. 172, Article No. 107357, May, 2020.

[29] B. Maringgal, N. Hashim, I. S. M. A. Tawakkal, M. T. M. Mohamed, M. H. Hamzah, N. I. A. Shukor, The causal agent of anthracnose in papaya fruit and control by three different Malaysian stingless bee honeys, and the chemical profile, *Scientia Horticulturae*, Vol. 257, Article No. 108590, November, 2019.

[30] K. Komosinska-Vassev, P. Olczyk, J. Kazmierczak, L. Mencner, K. Olczyk, Bee Pollen: Chemical Composition and Therapeutic Application, *Evidence-Based Complementary and Alternative Medicine*, Vol. 2015, pp. 1-6, March, 2015.

Biographies

Chen Wang MIEEE, MRICS, MIETI, MCRIOCM is a Distinguished Professor in the College of Civil Engineering, Huaqiao University, Xiamen, China, and the Head of Department, Department of Construction Management. He serves as the director of Intelligence and Automation in Construction Fujian Province Higher-educational Engineering Research Centre, China. He was awarded Overseas Price by Institution of Civil Engineers The United Kingdom. (ICE, London, U.K.).

Yutong Tang is a postgraduate student at the College of Civil Engineering, Huaqiao University, Xiamen, China. Her research interests include BIM, information system, and smart control.

Mohamed Faisal Ahmad-Akhia is a researcher at the Intelligence and Automation in Construction Fujian Province Higher-educational Engineering Research Centre, College of Civil Engineering, Huaqiao University. His research interests include smart farming, ICT, and computer programming.

Hamzah Abdul-Rahman was a Professor at the University of Malaya, Kuala Lumpur, Malaysia and the former President of the International University of Malaya-Wales, Kuala Lumpur, Malaysia before holding the current position as the President of Vision College, Petaling Jaya, Malaysia. His areas of interest in project management include risk management and quality management.

Jeffrey Boon Hui Yap is an Associate Professor in built environment at the Department of Surveying, Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman (UTAR), Kajang, Selangor, Malaysia. He obtained his PhD in project management from International University of Malaya-Wales (IUMW), Kuala Lumpur, Malaysia, in which he was recognised as IUMW Valedictorian. His research has appeared in a range of peer-reviewed journals. His areas of research include Design Changes and Control, Construction Engineering, Project Learning, Strategic Project Management, and System Dynamics in Engineering Management.