Applied Information Technology And Computer Science Vol. 3 No. 2 (2022) 1076-1093

AITCS

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

A Development of COVID-19 Support Application System

Ainin Sofiya Zahari[1], Muhaini Othman[1]*

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

DOI: https://doi.org/10.30880/aitcs.2022.03.02.065
Received 27 July 2022; Accepted 25 October 2022; Available online 30 November 2022

Abstract: COVID-19 Support Application System is a system that connect affected people during the pandemic to the non-governmental organizations (NGOs) that will aid them in providing the assistance that they required. The NGOs is currently provided the assistance through several ways such as receiving phone calls and messages from individuals telling the NGOs about their daily struggle. The NGOs also have created Google Form as initiatives to collect information. Consequently, the current information collection system is not systematic and centralized. This practice may cause miscommunication and information may be mishandled or lost, resulting the assistance never reaching those in need. As such, there is the need to develop a centralized database system to handle the collected data systematically. The system consists of two main users, which is administrators from the NGOs and applicants who is an affected people during the pandemic. The prototype model is being used as the guideline and standard along the process of system development. The system will be developed using PHP language and MySQL database. The system will ease operation for the administrator and the applicants to require for an assistance.

Keywords: application system, decision support system, web-based system

1. Introduction

A support application system is a type of system that connect affected people during the pandemic to the non-governmental organizations (NGOs) that will aid them in providing the assistance that they require. The COVID-19 epidemic has taken a severe toll on many people, resulting in catastrophic loss of human life worldwide and creating an unprecedented public health problem[1]. As it has swiftly impacted people's everyday lives, undoubtedly many people have lost their jobs, been financially insecure, and jeopardized their mental health. COVID-19 has had a huge financial impact on everyone's family. Many individuals who rely only on enterprises such as restaurants, stores, and others have suffered significantly because of the execution of the movement restriction order.

The support system is developed to assist the NGOs in better managing individuals who seek assistance. Currently, the NGOs has provided the assistance through several ways such as receiving phone calls and messages from individuals telling the NGOs about their daily struggles. The NGOs also

*Corresponding author: muhaini@uthm.edu.my

have created Google Form as initiatives to collect information this type of information. Consequently, the current information collection system is not systematic and centralized. It is also handled by individual NGOs representative. This practice may cause miscommunication and information may be mishandled or lost, resulting in the assistance never reaching those in need.

Therefore, there is the need to develop a centralized database system to handle the collected data systematically that can also be shared among NGOs representatives. Through the development of a centralized database system, problem such as data mishandled, and data lost can be minimized. It will also make it easier for NGOs to keep track of the applicants who request for assistance. With a systematic system in place, it will ensure fast and proper assistance is given to the right individuals.

The proposed system will require user to register their account and sign in to apply permission to use the system. All the application will be reviewed once the applicants submitted it. The system will notify the applicants of their current status such as pending, accepted or rejected. The procedure will cost them no more than one to two weeks working days after the date submitted.

This article is divided into six chapters. The first chapter provides an overview of the proposed system's features and functioning. The second chapter is a study of the proposed system's literature. The approach utilized in the project to develop the system will be discussed in chapter three. Furthermore, chapter four will discuss the system's analysis and design, and chapter five will illustrate how the suggested system is implemented. Finally, chapter six will review the system's benefits and drawbacks. Simultaneously, future works and project conclusions are outlined.

2. Related Work

The COVID-19 Support Application System is focused on providing the assistance for those who is in need during the pandemic. MyFundAction, the organization in the case study, currently relies on the manual way of supporting individuals in need. The applicants were required to request for assistance through phone call, text message and by filling out a Google Form. The manual process is inefficient, especially when dealing with large database containing sensitive information.

Through preliminary research with the NGOs, roughly they have been received about 13,000 request per year asking for an assistance ever since the pandemic has hit Malaysia in late January 2020. By October 2021, the total number of cases exceeded the two million marks with there are currently 60,000 active cases where 99% in mild condition and 1% in serious or critical condition [2]. As Malaysia has been under lockdown for over two years, it is evident that many individuals have suffered financially and psychologically.

According to a recent survey done by The Department of Statistics Malaysia, which included 168,182 respondents aged 15 and above, over half of self-employed employees had lost their jobs, with 94.8 percent reporting a decrease in monthly income [3]. Suicides and calls to helplines in Malaysia are also on the rise as a result of the outbreak [4]. The growing number of vulnerable people has been crying out for assistance, resulting with many NGOs, companies, and others aiding to those in need. In addition, the government has also launched several programs to aid the individuals.

As a result, the suggested system is designed to complement the present system of delivering aid with a computerized system, ensuring that those in need are never neglected and the process of getting assistance is much improved.

The three existing system are studied and compared to the features of the proposed system. The existing system studied included Kitajaga.co, Marhaen.my and Sambal SOS. This includes the modules contained in the COVID-19 Support Application System. The comparison results are shown in Table 1.

Table 1: System's comparison

Features / System	Kita Jaga	Marhaen	Sambal SOS	Proposed System
System Type	Web-based, mobile application	Web-based	Web-based	Web-based
Login Module	Available	Available	Available	Available
Registration Module	Available	Available	Available	Available
Category Module	Available	Unavailable	Available	Available
Data Management Module	Available	Available	Available	Available
Application Module	Unavailable	Unavailable	Unavailable	Available
Decision Module	Unknown	Unknown	Unknown	Available
Database	Unknown	Unknown	Unknown	mySQL

3. Methodology

The prototype model was chosen as the methodology in this project. There are six phases in the prototype model which is planning, analysis, design, implementation, testing and evaluation [5]. It was chosen since users can provide prompt response and comments on the system functionality [6]. Prototype is a software development in which a prototype is produced, tested and revised until it is acceptable [7]. Table 2 shows the software development phases and the activity conducted.

Table 2: Software Development Activities

Phase	Activity	Deliverable
Planning	• Identify the stakeholder and gather information about the project • Identify the problem statement • Determine the scope and objective of the project	• Project proposal • Gantt chart
Analysis	• Analyze the problem and study the feature to develop in the system • Decide the system requirements	• DFD • ERD

Table 2: (cont.)

Design	• Design system design, database design and user interface	• System flowchart • Database schema and data dictionary • User interface design
Implementation	• Develop the system based on the requirement of the system	• Program code
Testing	• Demonstrate and validate the feature	• Test case
Evaluation	• System deployed on the servers • Evaluate the progress made towards the completing the requirements	• Introduce new features • Resolving bugs • Complete documentation and workable system

4. Analysis and Design

This section discusses the study of the system analysis and design, which are used during the generic framework activities of modelling. Proper system analysis offers an effective path to the development by minimizing future IT requirements.

In this project, structured approach is optimized to perform the system analysis. Context diagram is shown in Figure 1. Context diagram presents the overview of interaction between the system and its user. The external entities of the system consist of administrator and applicants.

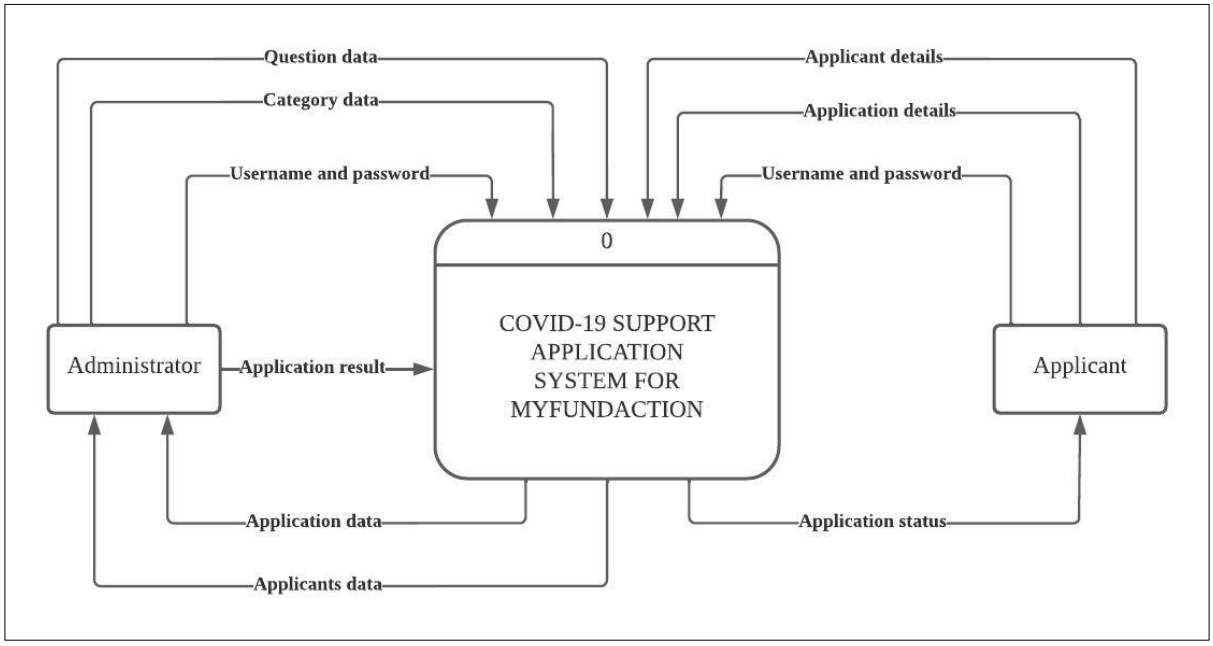

Figure 1: Context Diagram

Data Flow Diagram (DFD) is a graphical presentation of the flow of data or input from an entity through a process, which then generates output either to another entity or stored in storage data. Figure 2 shows the DFD Level 0 of the developed system.

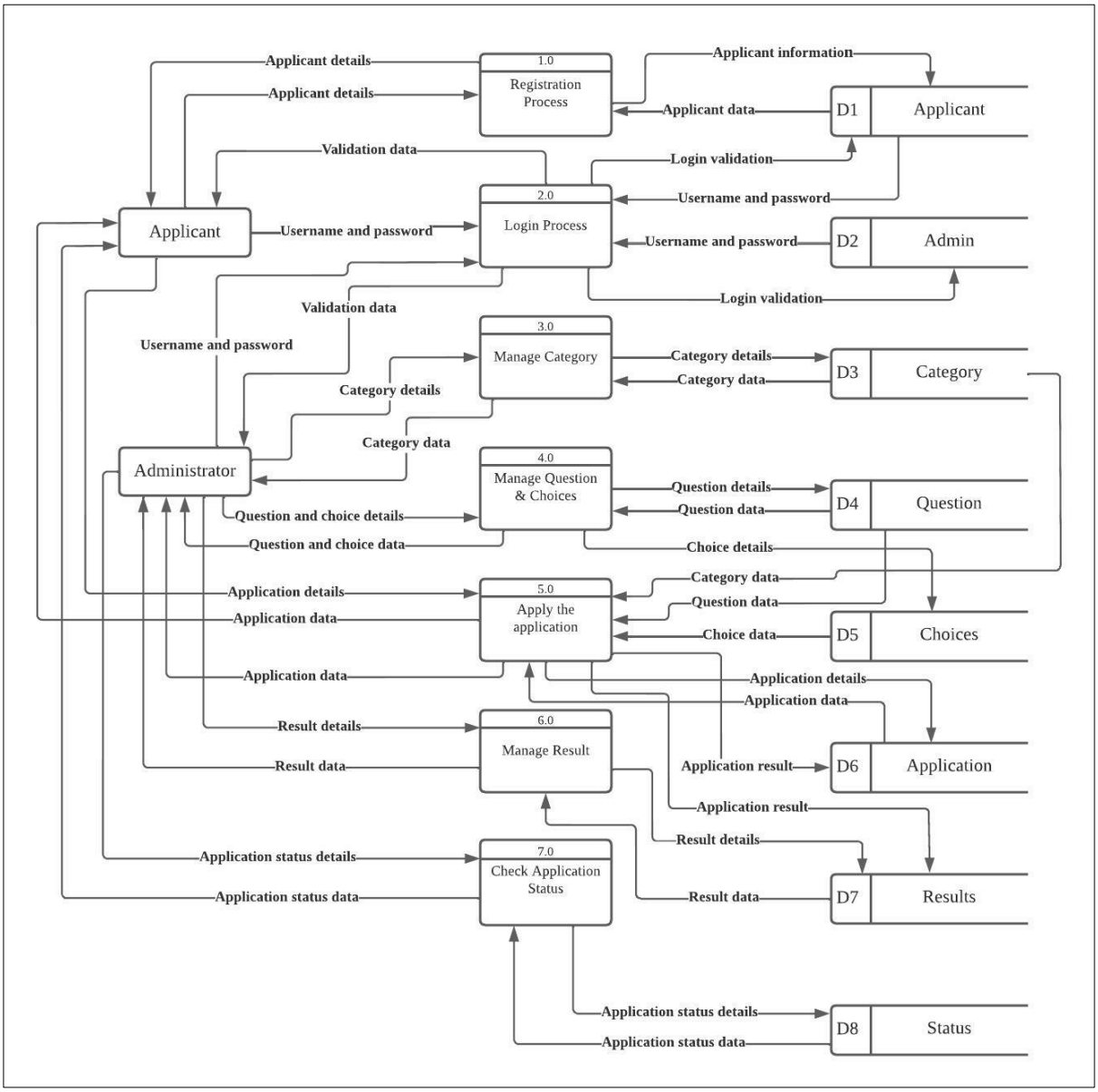

Figure 2: Data Flow Diagram

An entity relationship diagram (ERD) is a specialized graphic that illustrate the inter relationship between entities in a database. ERD use symbol to represent entities. Figure 3 shows the ERD of the developed system.

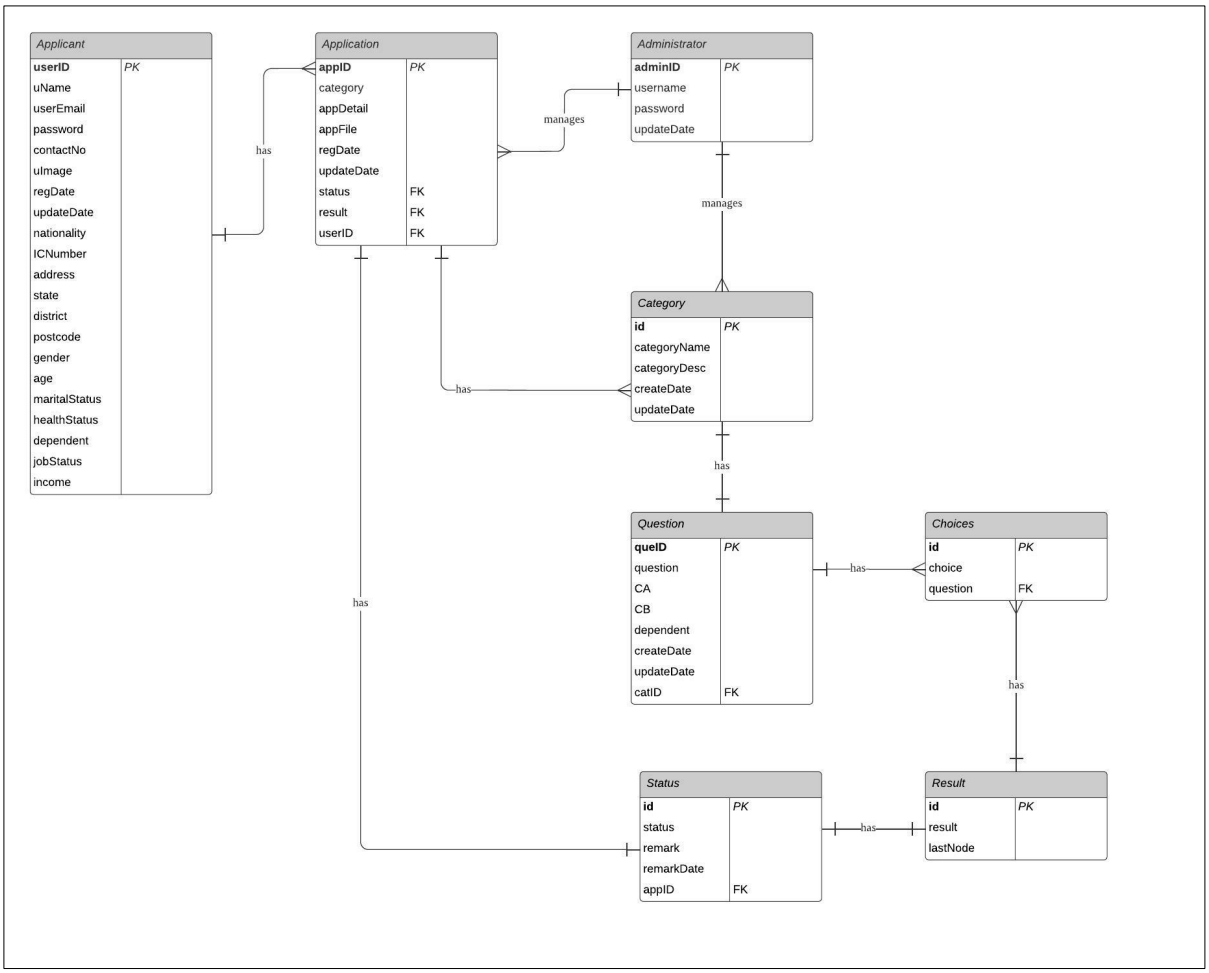

Figure 3: Entity Relationship Diagram

Flowchart is created to illustrate the overall process that can be executed by the administrator and applicants. Figure 4(a) shows the flowchart for the administrator and Figure 4(b) shows the flowchart for applicants.

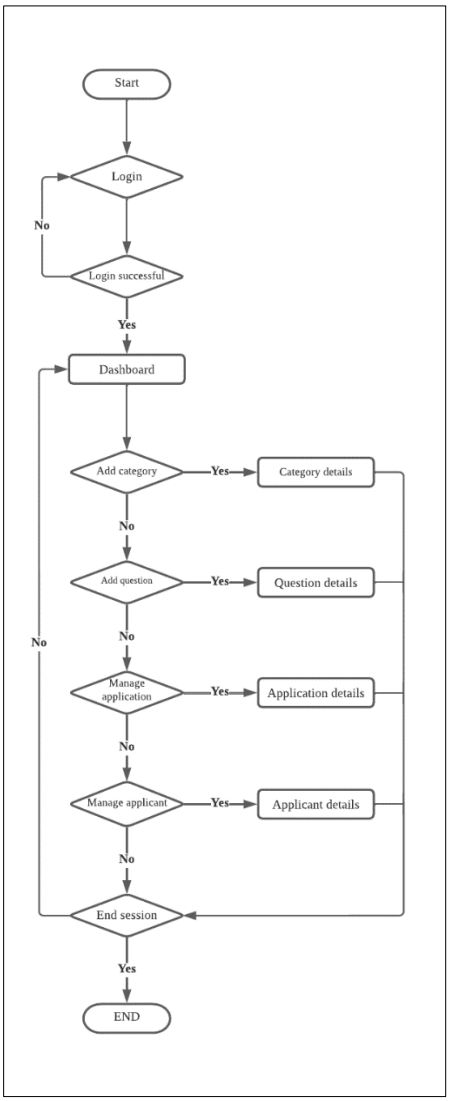

Figure 4(a): Flowchart for administrator

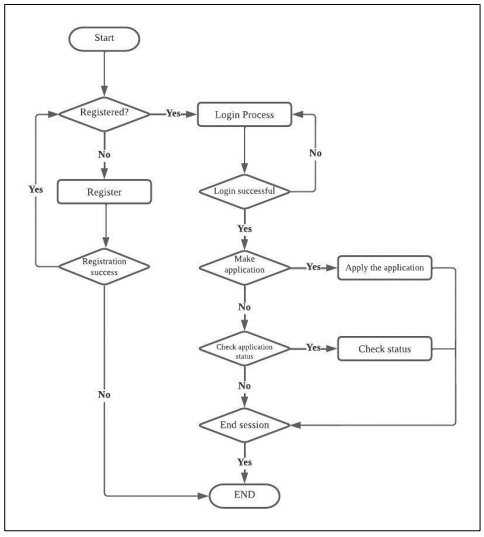

Figure 4(b): Flowchart for applicant

Decision tree uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs and utility[8]. Figure 5(a), (b) and (c) show a few of the decision tree based on the respective category.

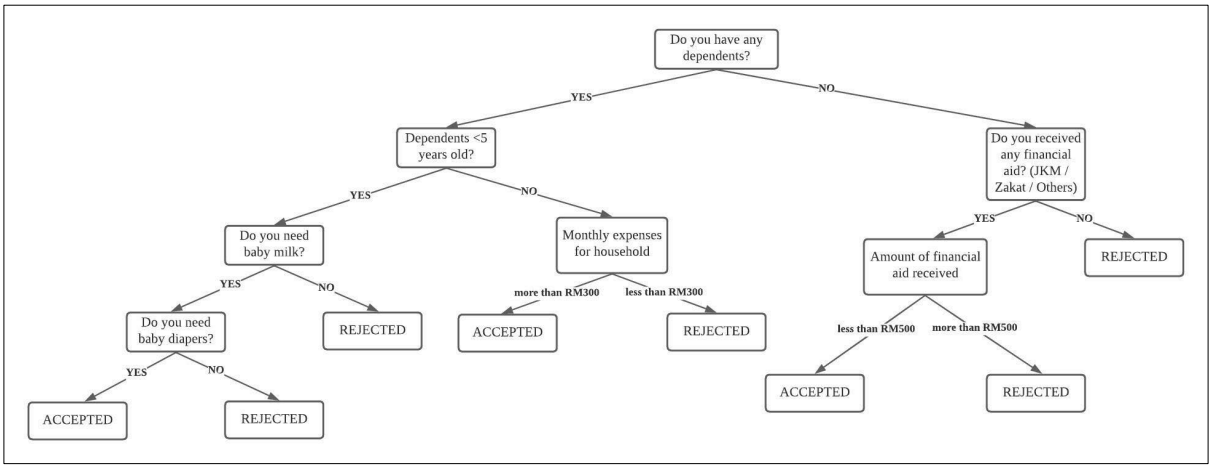

Figure 5(a): Decision Tree for General Category

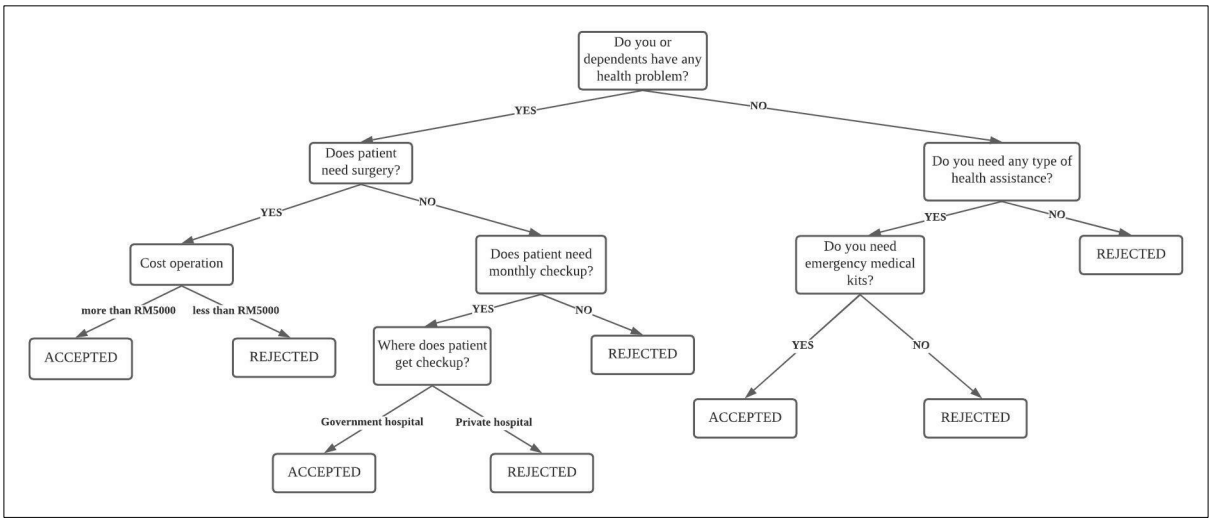

Figure 5(b): Decision Tree for Health Category

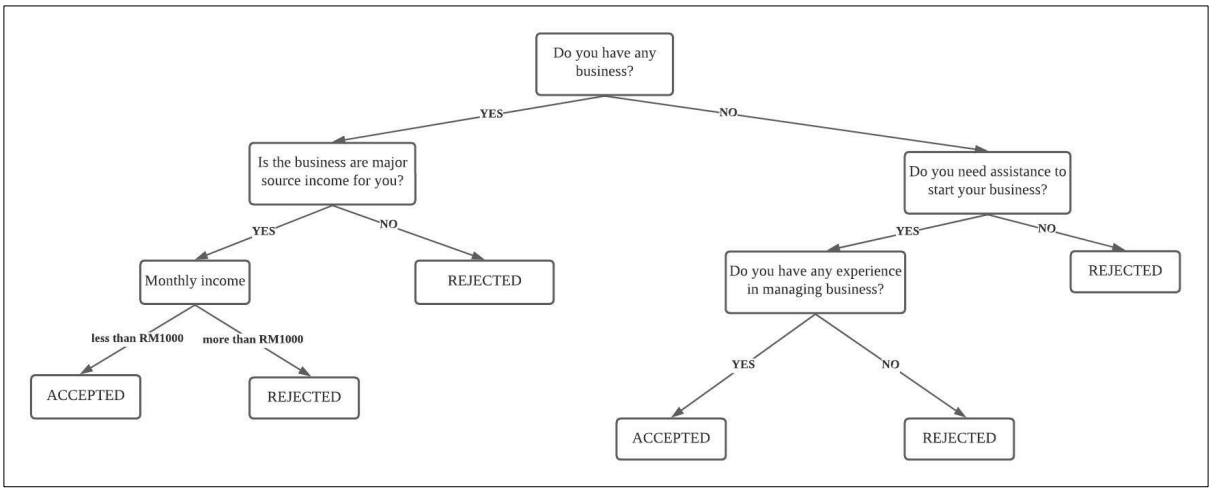

Figure 5(c): Decision Tree for Business Category

System interface design is to deliver the user a first impression of how the system appears and to serve as a medium for the user to engage with the system. Figure 6(a), (b) and (c) illustrate a few of user interface design of the developed system.

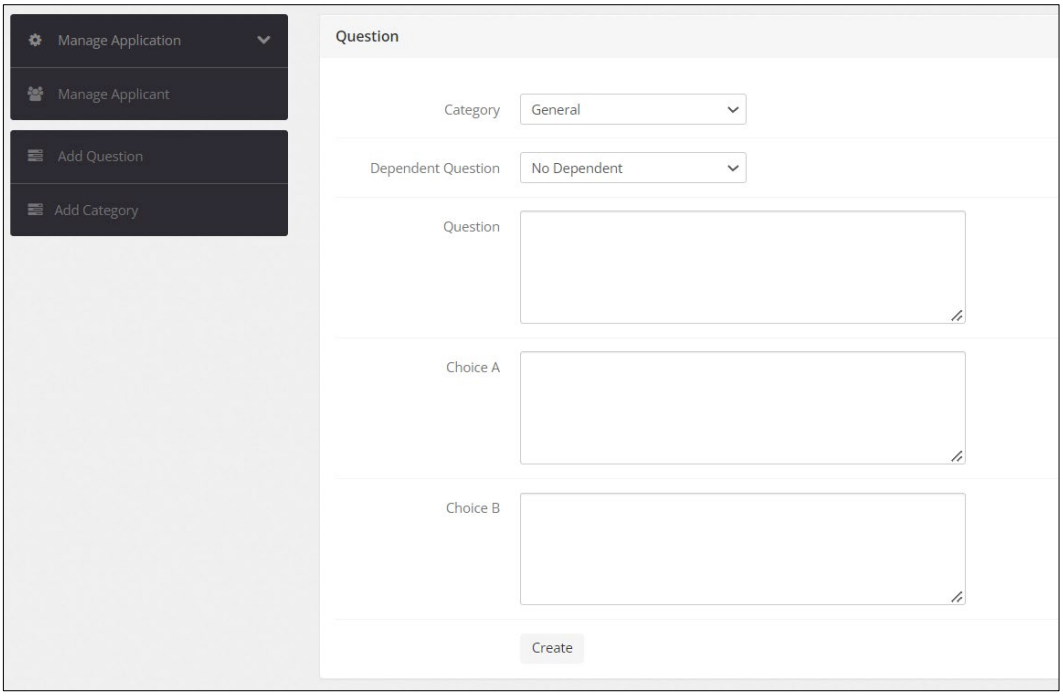

Figure 6(a): Interface of Add Question Page

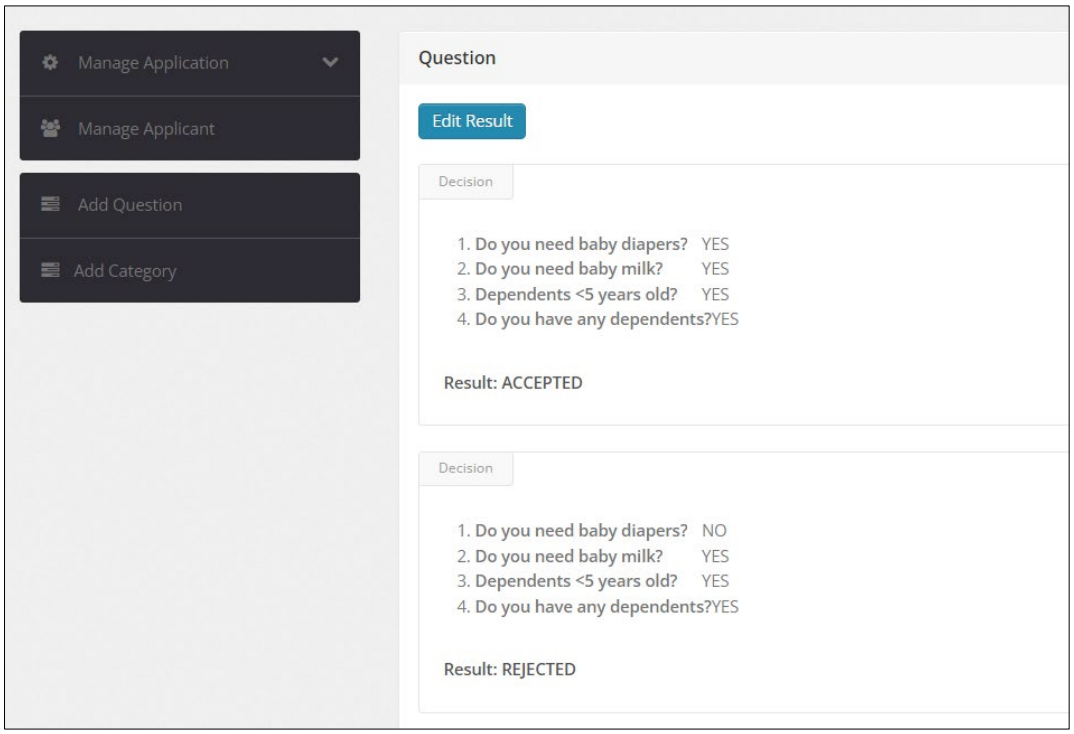

Figure 6(b): Interface of Decision Page

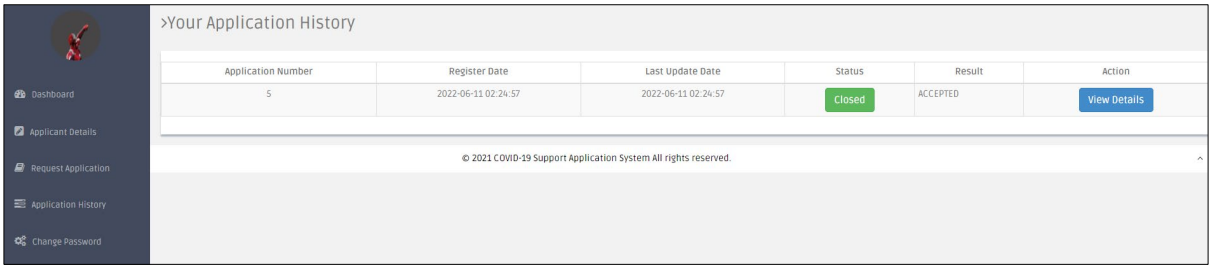

Figure 6(c): Interface of Application History Page

5. System Implementation and Testing

The system implementation phase involves the writing of software code and the building of a database for the proposed system, which was well planned and assessed throughout the system planning and design process. The planned system is constructed and tested to guarantee that no development faults occur.

This section will concentrate on how a decision support system works. The system may only be accessed by people who have a registered account. Before proceeding with the request for assistance, the user must fill out a form with their information. The system will ask applicants to select the type of help they want to obtain. For example, if the applicant selects a general category, just a questionnaire regarding that category will display, and the applicant will be required to response to it. After completing the questionnaire, the application's acceptance or rejection status will be displayed.

Figure 7(a) show an interface for applicant to answer the questionnaire, Figure 7(b) show the decision result after applicants completed the questionnaire. Figure 7(c) and (d) show the code segment of the questionnaire.

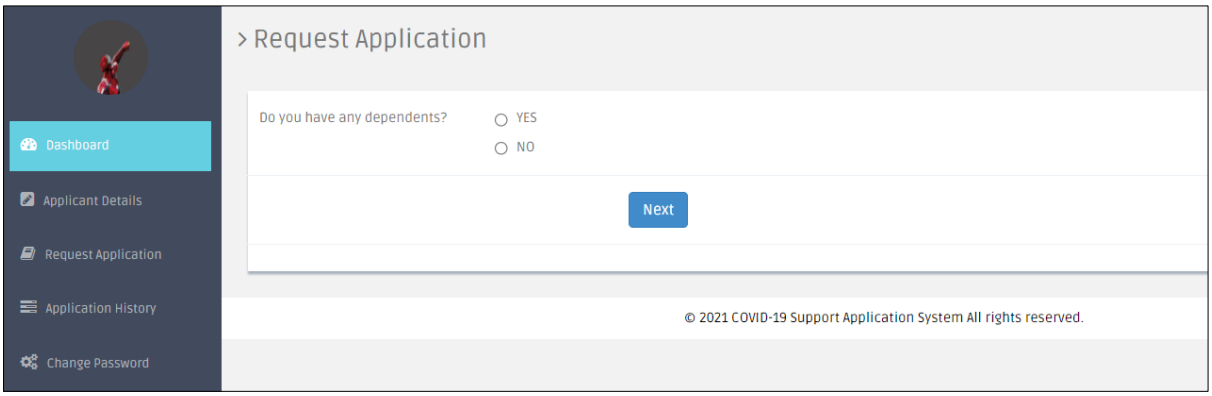

Figure 5(a): Questionnaire for Applicants

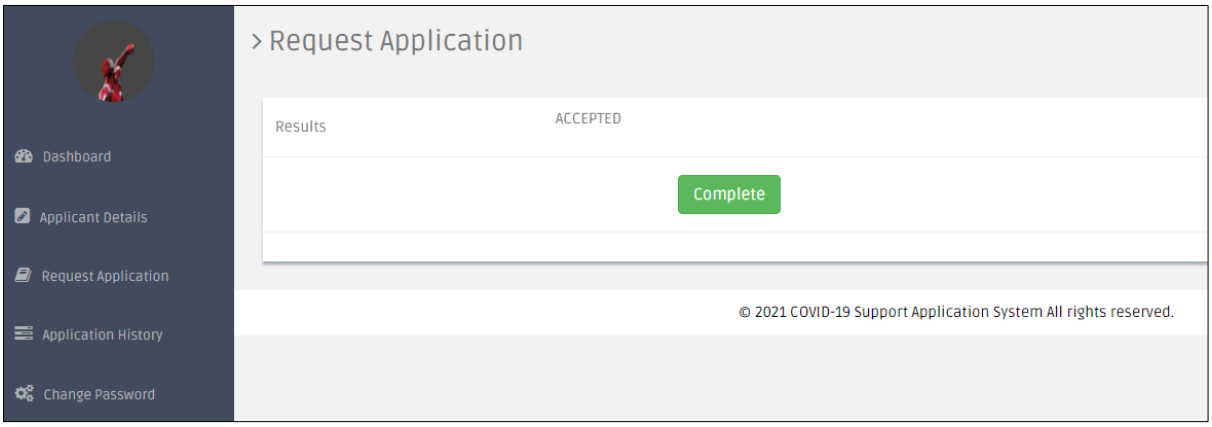

Figure 7(b): Result after Completing Questionnaire

```php
<?php
    if (isset($_GET['first_question']) || isset($_POST['next_question'])){

        $query = null;
        $queryQue = null;
        $lastNode = false;

        if (isset($_GET['first_question'])){
            extract($_GET);
            $queryQue=mysqli_query($con,"select *, question.queID as questionID, choices.id as choiceId, question.question as questionText
            from question LEFT JOIN choices on question.queID=choices.question WHERE question.catID='$cat' AND question.dependent IS NULL");
            $query=mysqli_query($con,"select *, question.queID as questionID, choices.id as choiceId, question.question as questionText
            from question LEFT JOIN choices on question.queID=choices.question WHERE question.catID='$cat' AND question.dependent IS NULL");
            $q = mysqli_fetch_assoc($queryQue);
        }

        if (isset($_POST['next_question'])){
            extract($_POST);

            $lastNodeQuery = mysqli_query($con,"select * from results WHERE lastNode='$answer'");

            if (mysqli_num_rows($lastNodeQuery) > 0){
                $lastNode = true;
                $r = mysqli_fetch_assoc($lastNodeQuery);

                $rId = $r['id'];

                mysqli_query($con,"UPDATE application SET result='$rId' WHERE appID='$appId'");

            }else{
                $queryQue=mysqli_query($con,"select *, question.queID as questionID, choices.id as choiceId, question.question as questionText
                from question LEFT JOIN choices on question.queID=choices.question WHERE question.catID='$cat' AND question.dependent='$answer'");
                $query=mysqli_query($con,"select *, question.queID as questionID, choices.id as choiceId, question.question as questionText
                from question LEFT JOIN choices on question.queID=choices.question WHERE question.catID='$cat' AND question.dependent='$answer'");
                $q = mysqli_fetch_assoc($queryQue);
            }
```

Figure 7(c): Code Segment of Questionnaire

```
if(isset($_POST['submit']))
{
$category=$_POST['category'];
    $question=$_POST['question'];
    $choiceA=$_POST['choiceA'];
    $choiceB=$_POST['choiceB'];
    $dependent = $_POST['dependent'];
    $query = $dependent == "0" ? "insert into question(question,CA,CB,catID) values('$question','$choiceA', '$choiceB','$category')"
    : "insert into question(question,CA,CB,catID, dependent) values('$question','$choiceA', '$choiceB','$category', '$dependent')"; ;

$sql=mysqli_query($con,$query);

    $question_id = mysqli_insert_id($con);

    $sql2=mysqli_query($con,"insert into choices(choice, question) values('$choiceA','$question_id')");
    $sql3=mysqli_query($con,"insert into choices(choice, question) values('$choiceB','$question_id')");

$_SESSION['msg']="Question Created!";
```

Figure 7(d): Code Segment of Questionnaire

5.1 Testing

A validation phase will be performed to confirm that all functionalities are functioning properly. As a result, it guarantees that the equipment fits the specification. This phase will be conducted after the system has been fully developed and deployed. Table 3(a), (b), (c), (d), (e) and (f) depicts the test case based on respective modules.

Table 3(a): Login Module

Test Cases	Expected Result	Result	Status
TEST_100_001	The system should allow the user to sign in using their respective ID and password.	Username and password entered normally	PASS
TEST_100_002	To sign in as a user, the system should allow the user to input a valid user ID and password.	Username and password can be entered normally	PASS
TEST_100_003	The system should notify the user if the ID and password are wrong.	The system shows the error message	PASS
TEST_100_004	If user has successfully logged in, the user will be redirect to respective dashboard.	System redirects to respective dashboard	PASS

Table 3(b): Registration Module

Test Cases	Expected Result	Result	Status
TEST_200_01	The system will require registration before signing into the new user	A button to register an account is available	PASS
TEST_200_02	The system should indicate an error when entering a duplicate username	The system shows the error message	PASS
TEST_200_03	The system will allow the user to login after successfully registered	The system allowed user to login after successfully registered	PASS
TEST_200_04	The system should not allow the user to login after the failed registration	The system did not allow an unregistered account to login	PASS

Table 3(c): Category Module

Test Cases	Expected Result	Result	Status
TEST_300_01	The system allow new category to be added into the system	User can add the new category when clicking the “Add Category”	PASS
TEST_300_02	The system should allow to edit the category after create	User can edit the category	PASS
TEST_300_03	The system should display the updated category after was edited	The system displays the updated category	PASS

Table 3(c): (cont.)

TEST_300_04	The system should allow user to delete the category	User can delete the category	PASS
TEST_300_05	The system should not display the deleted category	The deleted category was not displayed	PASS

Table 3(d): Data Management Module

Test Cases	Expected Result	Result	Status
TEST_400_01	The system should allow the user to view the manage page	User can redirect to manage application and applicants page	PASS
TEST_400_02	The system should allow the user to retrieve the data from the database	All of the data application and applicant can be displayed	PASS
TEST_400_03	The system should allow the user to view the details of the data	User can view the details of the application and applicants	PASS
TEST_400_04	The system should allow the user to delete the data of applicants	User can delete the applicant's data	PASS
TEST_400_05	The system should allow the user to update the status of application based on the decision result	User can update the status	PASS

Table 3(e): Application Module

Test Cases	Expected Result	Result	Status
TEST_500_01	The system should allow the user to	User can redirect to the request application	PASS

Table 3(e): (cont.)

	request for an application		
TEST_500_02	The system should allow the user to choose the category and input all the required data	User can choose category and input the data	PASS
TEST_500_03	The system should not allow the user to request for an application without choosing the category	The system shows an error and did not allow the user to proceed without choosing the category	PASS
TEST_500_04	The system should display a message after the request application is successful	The system popup an alert message	PASS
TEST_500_05	The system should allow the user to view the history of application after it was submitted	User can view the application history	PASS
TEST_500_06	The system should be able to display all of the details of the application	User can view the application details	PASS
TEST_500_07	The system should be able to display the status of the application	User can view the status of the application	PASS

Table 3(f): Decision Module

Test Cases	Expected Result	Result	Status
TEST_600_01	The system should be able for user to add the question	User can add the question	PASS
TEST_600_02	The system should be able to display all of the question that was created	User can view the question	PASS
TEST_600_03	The system should be able to input the decision result based on the questionnaire	User can input the result of the questionnaire	PASS
TEST_600_04	The system should be able for user to answer the questionnaire	User can answer the questionnaire	PASS
TEST_600_05	The system should be able to generate the decision after the questionnaire was completed	System generates the decision result	PASS
TEST_600_06	The system should display the decision result once the questionnaire was complete	System displays the decision result	PASS

6. Discussion

COVID-19 Support Application System is a system that assist individuals in requesting for an assistance. The system was developed to solve the limitation of the manual procedure that the NGOs currently used. In addition, the system will ease operation for the administrator and the applicant to request for an assistance.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support and encouragement throughout the process of conducting the study.

References

[1] S. C. K. Jamal Hisham Hashim, Mohammad Adam Adman, Zailina Hashim, Mohd Firdaus Mohd Radi, "COVID-19 Epidemic in Malaysia : Epidemic Progression, Challenges, and Response," vol. 9, no. May, pp. 1–19, 2021, doi: 10.3389/fpubh.2021.560592.

[2] MKM, "COVIDNOW in Malaysia ," *Ministry of Health Malaysia* , 2022. https://covidnow.moh.gov.my/ (accessed Nov. 11, 2021).

[3] D. of S. Malaysia, "REPORT OF SPECIAL SURVEY ON EFFECTS OF COVID-19 ON ECONOMY AND," no. Round 1, pp. 0–16, 2020.

[4] H. Hassan, "Malaysia sees rise in suicides and calls to helplines amid Covid-19 pandemic, SE Asia News & Top Stories - The Straits Times," *The Straits Times*, 2021. https://www.straitstimes.com/asia/se-asia/malaysia-sees-rise-in-suicides-and-calls-to-helplines-amid-covid-19-pandemic (accessed Nov. 11, 2021).

[5] K. P. Sayan, "Software Engineering | Phases of Prototyping Model | Set - 2 - GeeksforGeeks," 2018, [Online]. Available: https://www.geeksforgeeks.org/software-engineering-phases-prototyping-model-set-2/.

[6] A. R. Dennis, R. N. Burns, and R. B. Gallupe, "Phased design: A mixed methodology for application system development," *ACM SIGMIS Database*, vol. 18, no. 4, pp. 31–37, 1987, doi: 10.1145/1017816.1017819.

[7] M. Martin, "Prototyping Model in Software Engineering: Methodology, Process, Approach," *Aug 27*, 2021. https://www.guru99.com/software-engineering-prototyping-model.html.

[8] L. G. Kabari, E. O. Nwachukwu, R. S. Polytechnic, R. State, and P. Harcourt, "Decision Support System Using Decision Tree and Neural Networks," vol. 4, no. 7, pp. 8–20, 2013.