Applied Information Technology And Computer Science Vol. 4 No. 1 (2023) 505-523

AITCS

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

Implementation of Blocking Function in Web Based Charity Management System

Lee Zhi Han[1], Suhaimi Abd Ishak[1]*

[1]Fakulti SainsKomputerdanTeknologiMaklumat,
UniversitiTun Hussein Onn Malaysia, Parit Raja, BatuPahat, 86400, MALAYSIA

*Corresponding Author Designation

DOI: https://doi.org/10.30880/aitcs.2023.04.01.029
Received 16 June 2022; Accepted 26 September 2022; Available online 30 June 2023

Abstract: Due to the Covid-19 pandemic many people are facing difficulties such as financial problem and it is not efficiency for event organizer to manage the event using social media and difficult for volunteers to search for charity events using the same method. The charity management system can simplify the participation process for the volunteers at the same time provide convenience for the event organizer to manage the event. The charity management system will be developed using HTML, CSS, and C# with ASP.NET framework and is compiled using Microsoft Visual Studio and Microsoft Server SQL. Volunteers can participate in a charity event and make donation to a fundraising through this system while event organizer can create and manage their event. Besides that, blocking function was implemented where blocked users cannot participate or create new event to reduce the case of scammer and uncooperative volunteers.

Keywords: Charity, Charity Management System, Management System, Blocking， Organizer, Volunteer

1. Introduction

Charity is a kindness action taken by organization or individual by giving resources such as money, food, or help to people who are facing difficulties, or holding activities without making profit [1].

Nowadays, due to the Covid-19 pandemic there are countless families facing financial problem. According to the research of Paul Selva Raj, the rate of unemployment and underemployment has increased in Malaysia where there around 3.4 million of people which are unemployment and underemployment which cover about 19.8% of the workspace. Besides that, the average salaries of the employee dropped by 9% while the median salaries drop by 15.6% [2].

Charity management system is a web-based system that will shows information about charity events and enables users to take part in it. Charity management system provides platform for users to learn more about the charity events and gives opportunities for them to participate in it. The system will be

*Corresponding author: suhaimi@uthm.edu.my

managed by organizer to manage various charity events. Users need to log in into the website to do registration and take part in the charity events.

The objectives of this project are to:

- Design a charity management system using object-oriented approach based on user requirements.
- Develop charity management system using HTML, CSS, C# programming language and Microsoft SQL Server Management Studio.
- Test all the modules and functions of the charity management system.

Statements below show the scopes that had been set up:

- This project target users are all people in Malaysia. The management system will show nonprofit fundraising and physical events that are held in Malaysia.
- Event organizer or workers need to register a organizer account to create and manage events, they can also report volunteers that are not cooperative. They can manage the expenses and update the attendance of the volunteers.
- Volunteers also need to register an account to participate into events or make donation.
- The system will be manage by an admin and admin can block users from participating into events or create new events.
- Users can access the system through any devices with web browser.

For the remaining section, section 2 will explained about the related work of this project. section 3 will cover the methodologies that are used in this project with functional analysis, section 4 will present the design of the system, and Section 5 will discussed about implementation and testing of the system.

2. Related Work

This section discusses on literature review that had been done for the charity management system and the current system that being used by the event organizer. Section 2.1 will discusses about the management system. IDEs software will be discussed in Section 2.2 Then, in Section 2.3 presents about the software development life cycle (SDLC) and in Section 2.4 discusses about the comparison between five types of existing system that had been studied with the developed system.

2.1 Management Information System

Management information system is a system that is used by specific people to collect, transmit, process, store, update and maintain the information through any computer hardware, software, and network communication equipment [3]. The main purpose of the management in the organization is to make sure that the achievements of the organization are stick to its objectives and goals [4]. Management Information System contains three main parts which are the theory of management, analysis in information, and decision-making system [5].

2.2 Integrated Development Environments (IDEs)

Integrated Development Environment (IDE) is a popular software application that provide convenient support to the developers during the implementation of the system, modern IDEs provide debuggers, automated refactors, source code editor and also version control to the developers [6]. In contrast, IDE is a integration of several software programs for development such as compiler, editor, linker, and operating commands. There are various type of IDEs such as multi-language IDE, HTML IDE, and

Clouded-based IDE and each of the developers will use different IDE according their preferences and the requirement of the software developed.

2.3 Software Development Life Cycle (SDLC)

System Development Life Cycle (SLDC) is the phases that exist in the development process where each phases consists of several activities. According to two groups of researcher, Lai and Tsen, Goyal and Ram said that SDLC is a concept which is used in project management to represent the phases from first phase to final phase that were deployed in the system development [7]. Most of the developers will use SDLC to improve the quality of the system developed and ensure a steady overall development process with lower cost. Waterfall model, Agile model, Iterative model, V-shaped model, Spiral model, and Big Bang model are the commonly SDLC model that is used by developers. There are six phases exist in the SDLC which are planning, analysis, design, implementation, testing and maintenance.

2.4 Comparison With Existing Systems

Table 1 shows the comparison of the existing system with the proposed system in this project in terms of the function and the feature provided by the system. These features that are compared are the ability to create outdoor charity events, ability to create fundraising events, the method to deliver latest information to the users, type of users that can use the system, ability to block uncooperative volunteers, ability to manage estimate costing, ability to generate report, contains admin panel, event registration method, and attendance recording.

Table 1: Comparison Between Existing System and Proposed System

Feature / System	Facebook [8]	GiftBox[9]	CharityEngine [10]	Agile Case[11]	Raklet[12]
Create outdoor charity events	Available	Not available	Not available	Not available	Available
Create Fundraising	Available	Not available	Available	Available	Available
Method to deliver latest information	Publish Post	Emails	Emails	Emails	Emails and Post
Type of users that can use the system	All public users	Staffs or volunteers only	Staff and donors only	Staff, volunteers, donors only	All public users
Block uncooperative volunteers	Not available	-	-	-	Available
Manage Estimated Costing	Not available	Available	Available	Available	Not available
Generate Report	Not	Available	Available	Available	Not

	available				available

Table 1: Comparison Between Existing System and Proposed System (cont)

Admin Panel	Not available	Available	Available	Available	Available
Event Registration Method	Through other platform	-	-	Through this system	Through this system
Record Attendance of volunteers	No	Available	-	Available	Available

For the proposed system, all type of users including the visitors can use this system. It will have the ability to create charity events and fundraising. They can also manage the expenses of the created event . The Event organizer can publish post to deliver latest information about the event. Event organizer can also record the attendance of the volunteers and report uncooperative volunteers after the events. Volunteers may register to participate in the event after the registration process for this system. Admin panel will be available to check the events and manage the users. Admin and event organizer can also generate report through this system.

3. Methodology

The section discuss about the methodology of the study and the analysis and design of the charity management system. Section 3.1 present the methodology of the study while section 3.2 present the requirement gathering and analysis.

3.1 Methodology of The Study

Waterfall model was chosen as the methodology in this project. Waterfall model is a type of software development life cycle (SDLC) approach with sequential design process. The reason it is called so is because every phases in the model is executed one by one in downward stream flow just like a waterfall [13]. The reason of choosing this model was because it is common and well-known method and easy to understand while the requirement in this project is clear and the project definition is stable.

3.2 Requirement Gathering and Analysis

The user requirements were determined using data collecting method which is questionnaire. Google Form was used to create questionnaire and analyzed the gathered data.

Table 2 shows the system development requirement while Table 3 shows the system requirement to use the system.

Table 2: System Requirement - Users

Software Requirements	Hardware Requirements
- Any web browser such as Google Chrome, Firefox, and Microsoft Edge.	- Mobile phone, laptop, desktop, smartwatch, tablet, or any devices with internet connection.

Table 3: System Development Requirement

Software Requirements	Hardware Requirements
- Operating System: Windows 10 - Web browser : Google Chrome - IDE development tool: Visual Studio - Database: Microsoft SQL Server Management Studio - Web Development tool: C# programming language, HTML, CSS	- Model : MSI Laptop - Processor: Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz - RAM: 8 GB - Storage: 219GB SSD

3.3 System Requirement Analysis

Requirement analysis is related with the user's aims and objectives and act as the framework for the function of the system with determined requirements, environment, and plan [14]. The requirements were based on the information that was gathered through questionnaire. Table 4 shows the functional requirements while Table 5 shows the non-functional requirement for the charity management system.

Table 4: Functional Requirement

No	Module	Functionalities
1	Register and Login - Volunteers - Event Organizers - Admin	● Users register as a new user ● System alert for any invalid input ● Users login using their own username and password
2	Event Manage Module - Event Organizers	● Event Organizers can create charity events and fundraising. ● Event Organizers can insert the cost description for the events. ● Event Organizers can update attendance of the volunteers.
3	Event Module - Volunteers - Event Organizers	● Volunteers can search for charity events. ● Volunteers can participate into charity events. ● Volunteers can make donation for fundraising. ● Volunteers can make comments under post.

4	Report - Event Organizers - Admin	● Event Organizers can generate the report with event details and list of participants and donors. ● Admin can generate the report with the list of the charity events and fundraising.

Table 4: Functional Requirement (cont)

5	Contact Us - Volunteers - Event Organizers -Admin	● Users can send feedback or complaints to admin. ● Admin can view the feedback list
6	User Management - Admin	● Admin can view the list of users. ● Admin can block users from participate into events or create new events.

Table 5: Non-Functional Requirement

Requirement	Description
Operational	● The system is accessible in any web browser as long as the internet connection is available.
Availability	● The system is available for almost 24 hours a day.
Security	● Users need to log in to the system before using the services provided by the system such as participating in charity events or creating charity events.
Usability	● The system and its user interface should be user-friendly and easy to understand.

4. Design

This section discuss about the design of the system where context diagram, level 0 data flow diagram(DFD), level 1 DFD, flowchart, and Entity Relationship Diagram (ERD) were presented.

4.1 Context Diagram

Figure 1 shows the context diagram of the system. Context Diagram shows the domain of the system by presenting the data flow sets of that flow through in and out of the system [15]. There are three entities which are volunteers, event organizers, and admin:

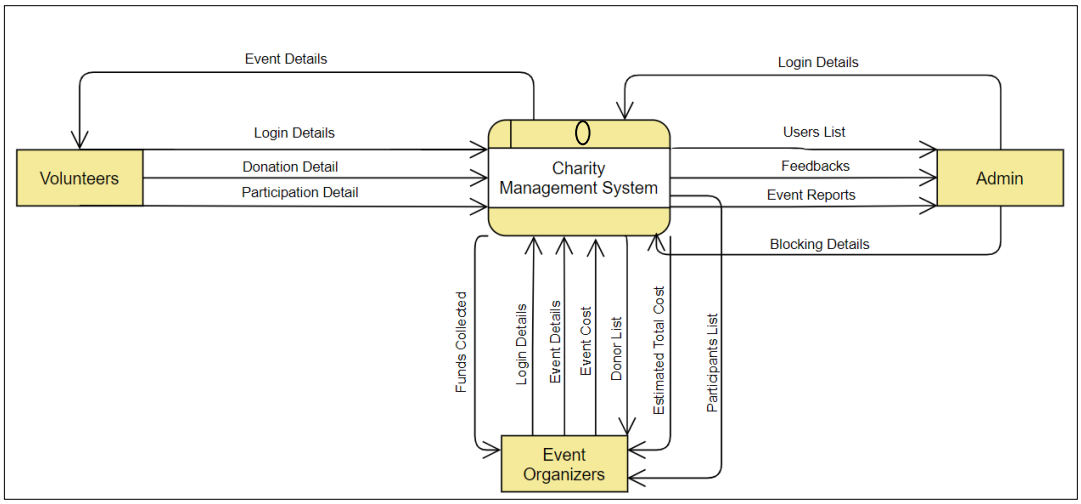

Figure 1: Context Diagram

4.2 Data Flow Diagram (DFD)

Figure 2 shows the level 0 data flow diagram of the system to present the data flow between the entities, processes, and the databases. There are total of eight processes and ten databases in this DFD. Figure 3 shows the level 1 DFD for Process 3.0 which is event management while Figure 4 and Figure 5 shows the level 1 DFD for Process 4.0, Process 5.0 which is event participation process and donation process, and Process 8.0 which is about user management.

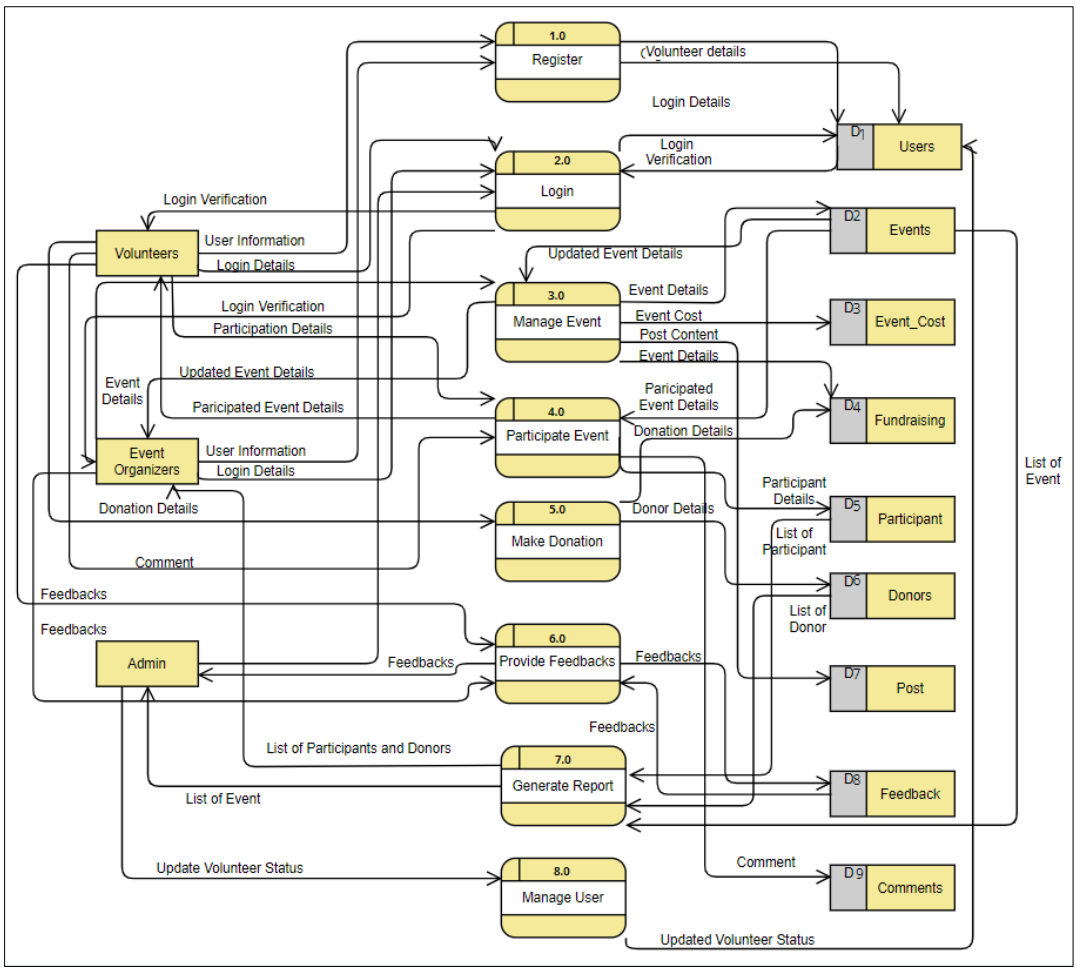

Figure 2: Level 0 DFD

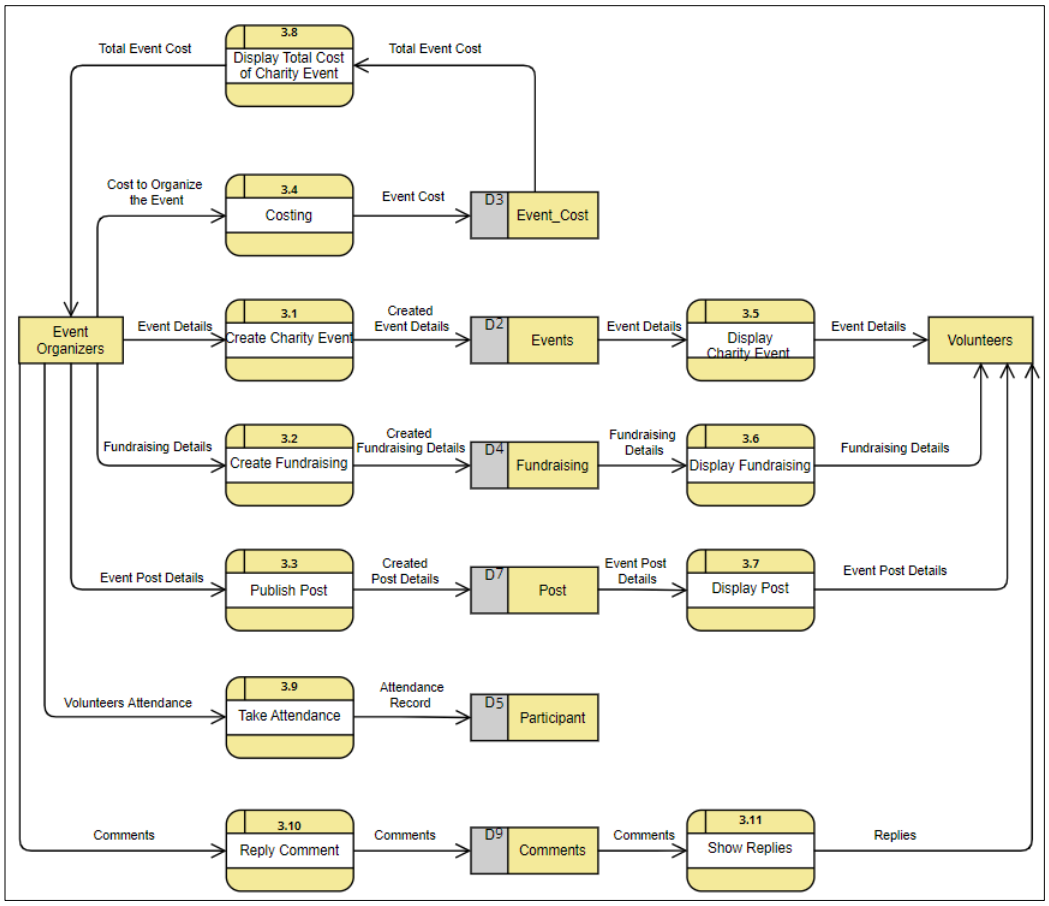

Figure 3: Level 1 DFD Process 3.0

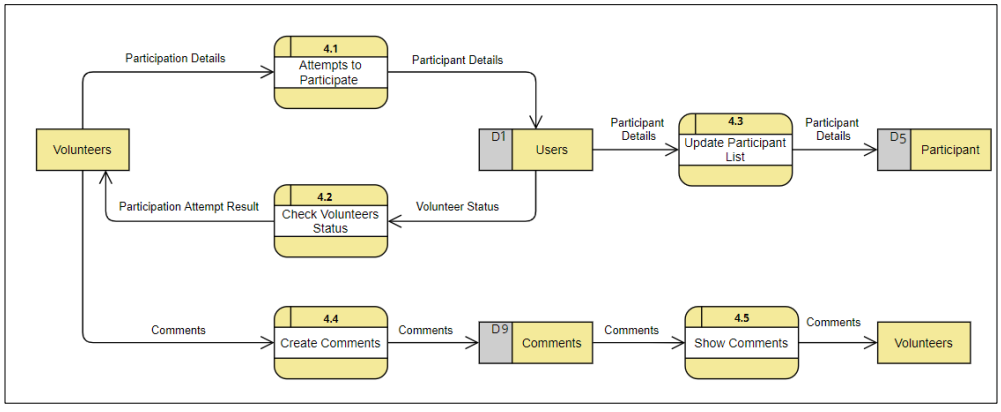

Figure 4: Level 1 DFD Process 4.0

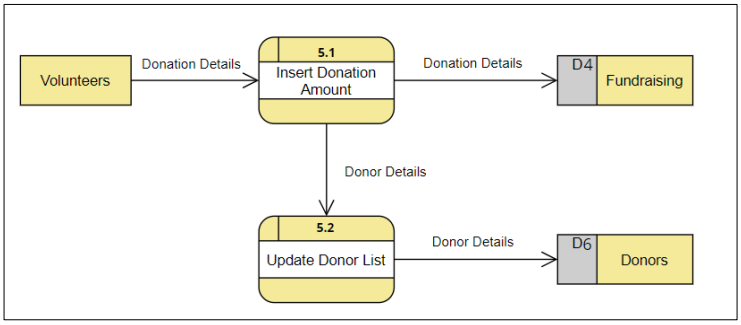

Figure 5: Level 1 DFD Process 5.0

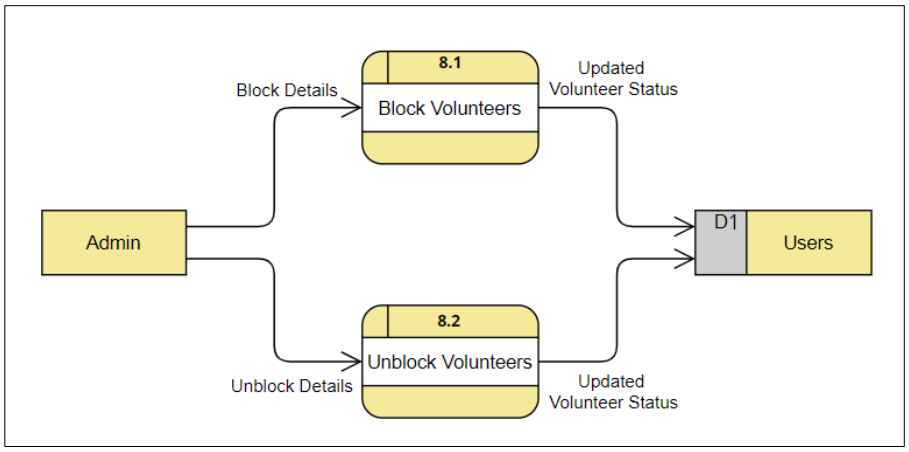

Figure 6: Level 1 DFD Process 8.0

4.3 Flowchart

Figure 7, Figure 8, and Figure 9 shows the flowchart on the volunteer side, event organizer side, and admin side respectively.

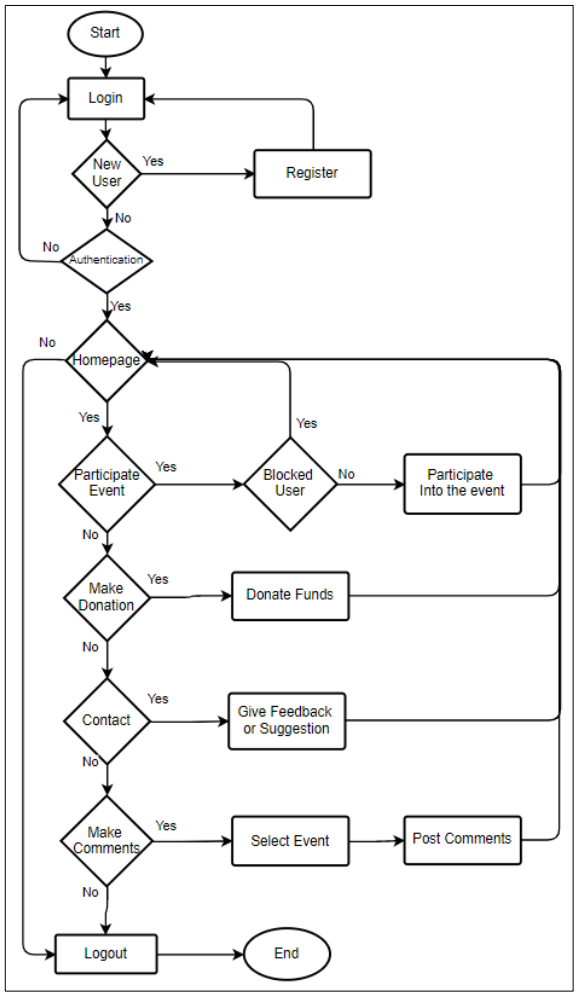

Figure 7: Flowchart (Volunteer Side)

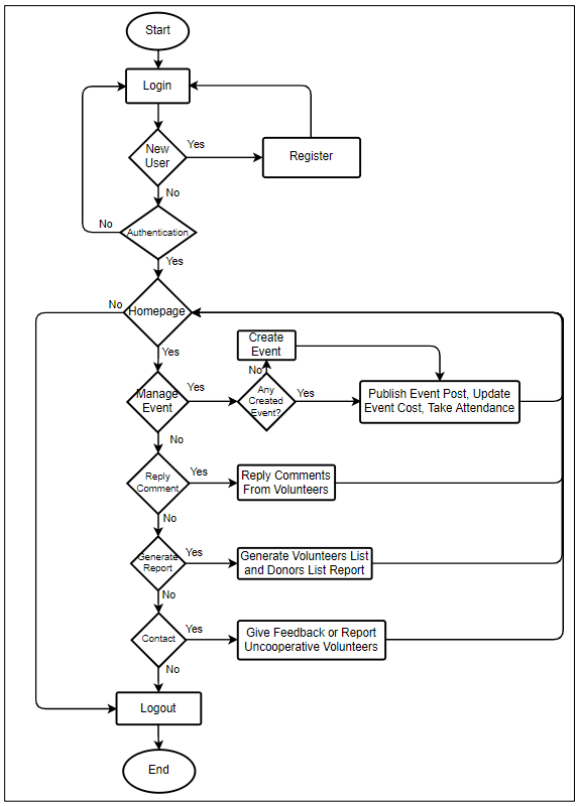

Figure 8: Flowchart (Organizer Side)

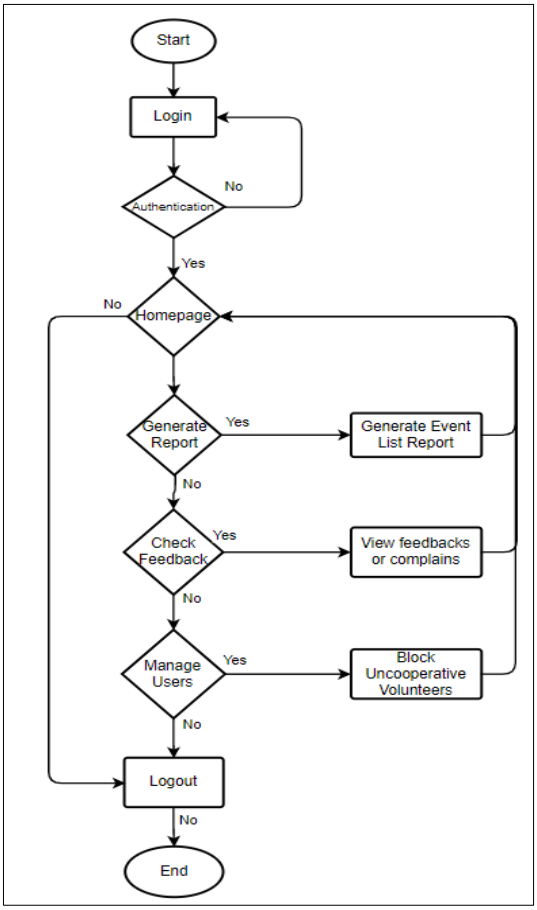

Figure 9: Flowchart (Admin Side)

3.7 Entity Relationship Diagram (ERD)

Figure 9 represents the ERD diagram of the charity management system.

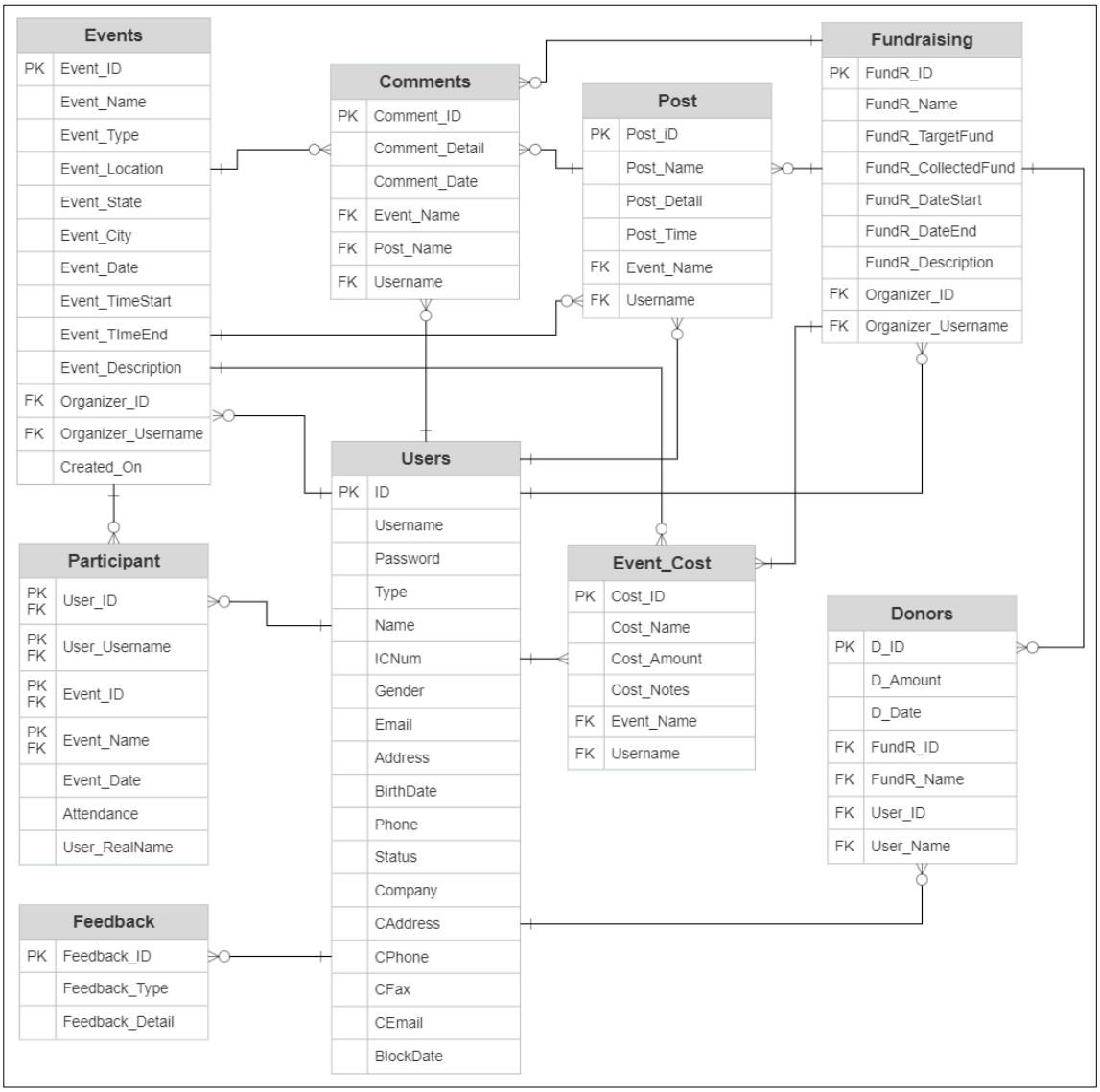

Figure 10: Entity Relationship Diagram (ERD)

5. Implementation and Testing

This section discuss about the implementation of the proposed system where the module's interfaces and code segments are presented. Beside that, testing was done after the implementation of the system and the test plan results are shown.

5.1 Implementation

Figure 11 show the interface of the homepage for volunteer side where it show the list of the event participated by them.

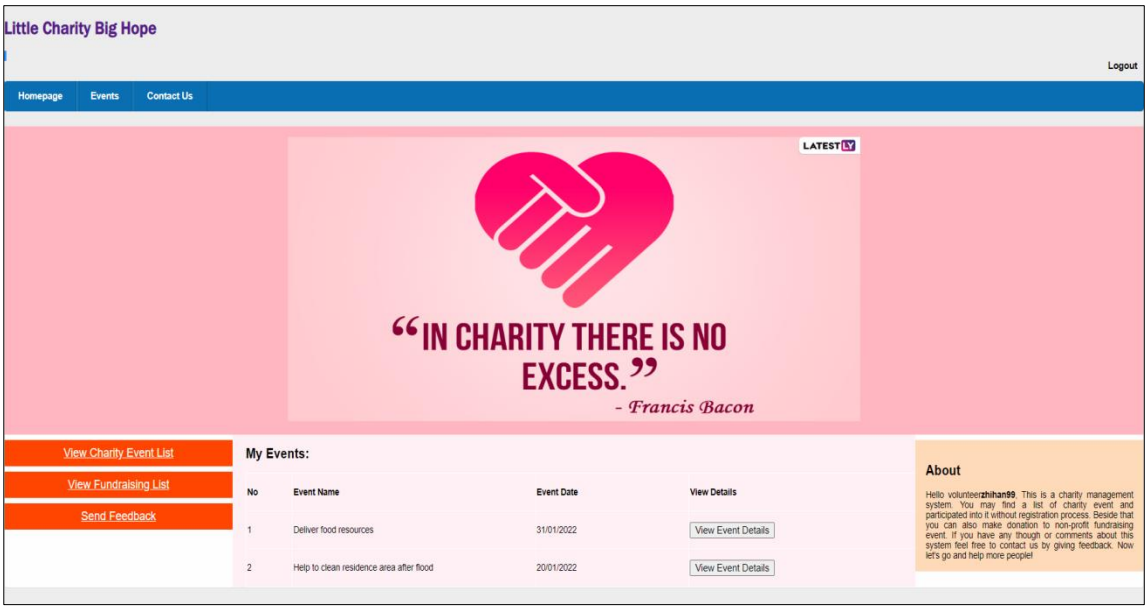

Figure 11: Homepage Interface

Figure 12 show the login and register interface that shown on the homepage of visitor.

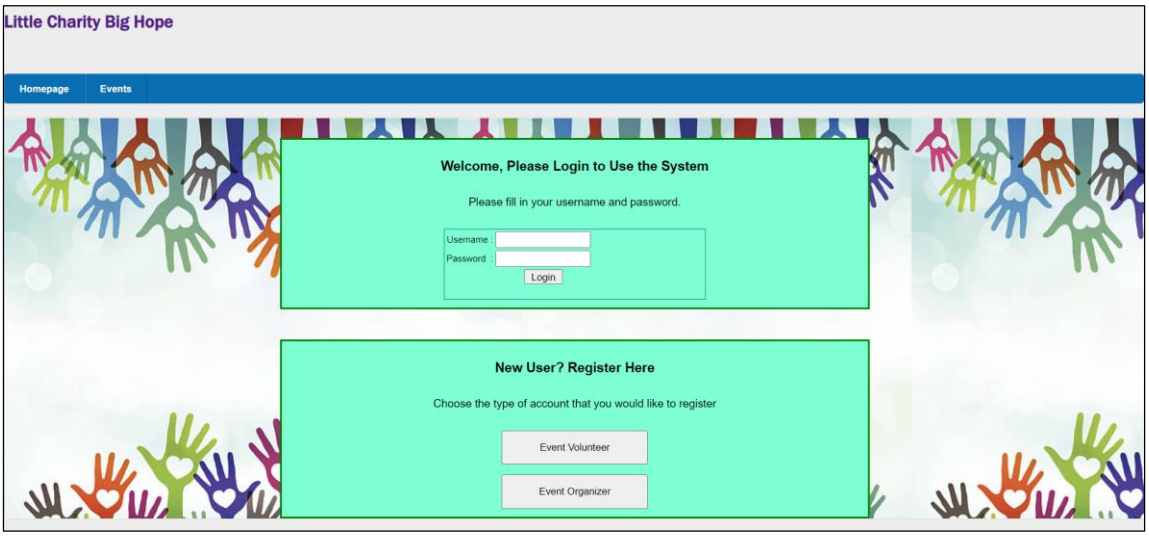

Figure 12: Login and Register Interface

Figure 13 show the interface of the registration form for organizer while figure 14 show the code segment to extract the birth date and gender from the IC number inputted by user in the form.

Figure 13: Registration Form Interface

```
public static BirthdayAgeSex GetBirthdayAgeSex(string icnum)
{

    if (icnum.Length != 14)
    {

        return null;
    }


    string birthyear = icnum.Substring(0, 2);

    BirthdayAgeSex entity = new BirthdayAgeSex();
    string strSex = string.Empty;
    if (int.Parse(birthyear) <= 22)
    {
        entity.Birthdate = "20" + icnum.Substring(0, 2) + "-" + icnum.Substring(2, 2) + "-" + icnum.Substring(4, 2);
        strSex = icnum.Substring(12, 2);
    }
    else
    {
        entity.Birthdate = "19" + icnum.Substring(0, 2) + "-" + icnum.Substring(2, 2) + "-" + icnum.Substring(4, 2);
        strSex = icnum.Substring(12, 2);
    }


    if (int.Parse(strSex) % 2 == 0)
    {
        entity.Sex = "Female";
    }
    else
    {
        entity.Sex = "Male";
    }
    return entity;
}
```

Figure 14: Code Segment to Get Gender and Birth Date from IC Number

Figure 15 show the interface of the event list that will be displayed to all types of user. The same interface applied on fundraising list also.

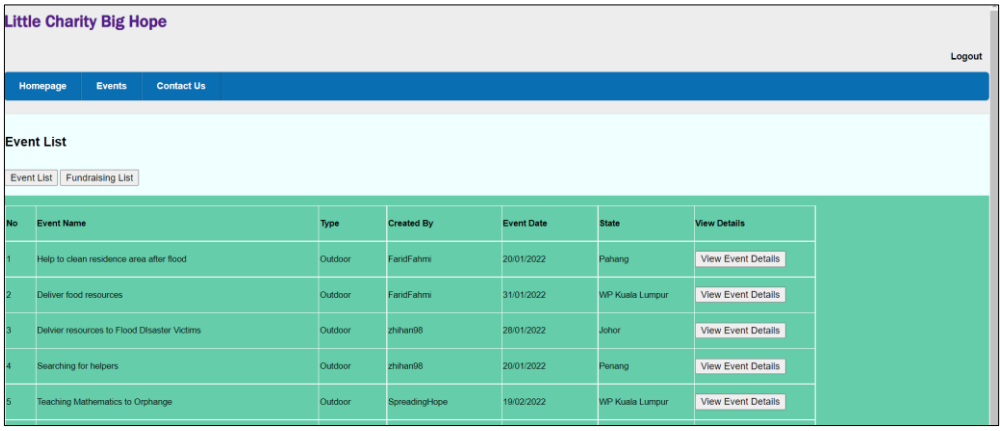

Figure 15: Event List Interface

Figure 16 show the interface of the event details on volunteer side. The list of post is displayed to volunteer and admin and only volunteers can participate to the event. Alert Message will shown if the volunteer was blocked or already participated to the event. Figure 17 show the code segment to identify whether the volunteer was in the block list when they participated to the event.

Figure 16: Event Details Interface

```
string sqldb = "Data Source=.\\SQLEXPRESS;Initial Catalog=CharityMS;Integrated Security=True"; ;
SqlConnection conn = new SqlConnection(sqldb);
conn.Open();
SqlCommand cmd = new SqlCommand();
cmd.Connection = conn;
cmd.CommandText = "SELECT Events.Event_ID, Events.Event_Date, Users.Status, Users.Name FROM Events, Users WHERE Events.Event_Name = @eventname AND Users.Username= @username";
cmd.Parameters.AddWithValue(@"eventname", Session["eventname"]);
cmd.Parameters.AddWithValue(@"username", Session["username"]);
cmd.CommandType = CommandType.Text;
SqlDataReader dr = cmd.ExecuteReader();
if (dr.Read())
{
    int event_id = dr.GetInt32(0);
    DateTime event_date = dr.GetDateTime(1);
    string status = dr.GetString(2);
    string realname = dr.GetString(3);
    conn.Close();

    if (status == "Blocked")
    {
        lblmsg.Text = "You have been blocked";
    }

    else
    {
```

Figure 17: Code Segment to Check Status of Volunteer

Figure 18 show the interface of the manage event list on organizer side. An error message will be displayed if when the organizer was blocked from creating new event or fundraising. Figure 19 show the code segment when organizer wants to create a new event.

Figure 18: Manage Event List Interface

```
protected void Create_Event(object sender, EventArgs e)
{
    string sqldb = "Data Source=.\\SQLEXPRESS;Initial Catalog=CharityMS;Integrated Security=True";
    SqlConnection conn = new SqlConnection(sqldb);
    conn.Open();
    SqlCommand cmdd = new SqlCommand();
    cmdd.Connection = conn;
    cmdd.CommandText = "SELECT Status FROM Users WHERE Username= @username";
    cmdd.Parameters.AddWithValue(@"username", Session["username"]);
    cmdd.CommandType = CommandType.Text;
    SqlDataReader dr = cmdd.ExecuteReader();
    if (dr.Read())
    {

        string status = dr.GetString(0);
        conn.Close();

        if (status == "Blocked")
        {
            lblmsg.Text = "You have been blocked from creating events and fundraising";
        }
        else
        {
            Response.Redirect("CreateEvent.aspx");
        }
    }
}
```

Figure 19: Code Segment when Organizer Attempt to Create a New Event

Figure 20 show the management of the event details on organizer side. Organizer can edit the event details, manage the expenses, and update the attendance of participant of the event.

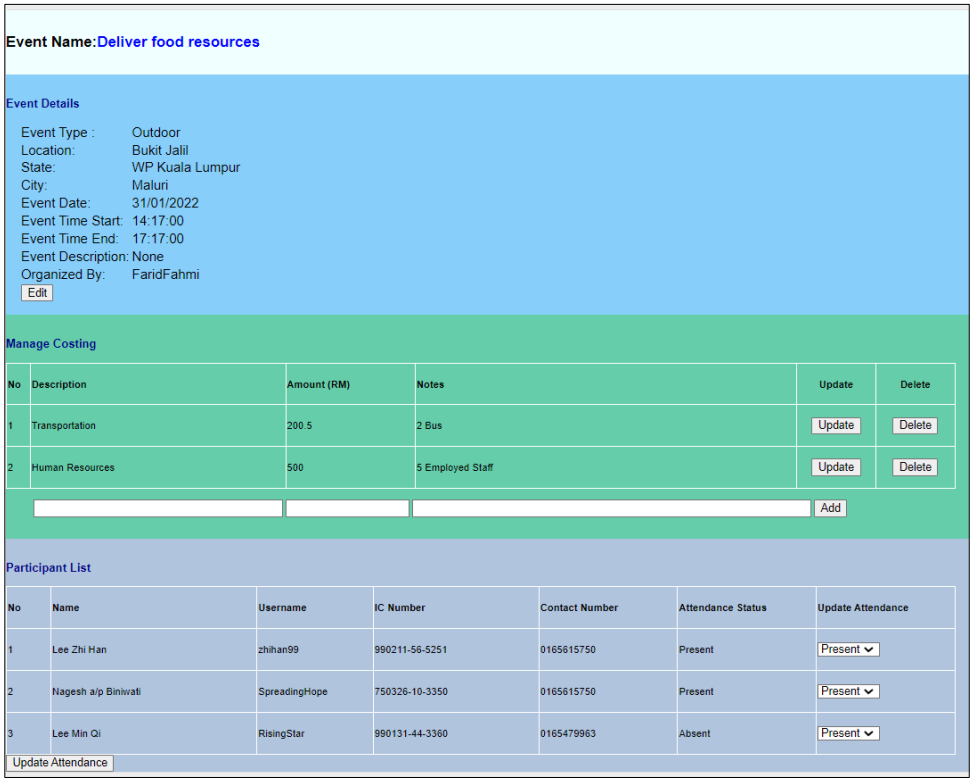

Figure 20: Manage Event Details Interface

Figure 21 show the user management interface on admin side. Admin can block or unblock the users. Blocked volunteers will be unable to participate into charity event while blocked organizer will be unable to create new charity events or fundraising. Figure 22 show the code segment when admin try to block an user.

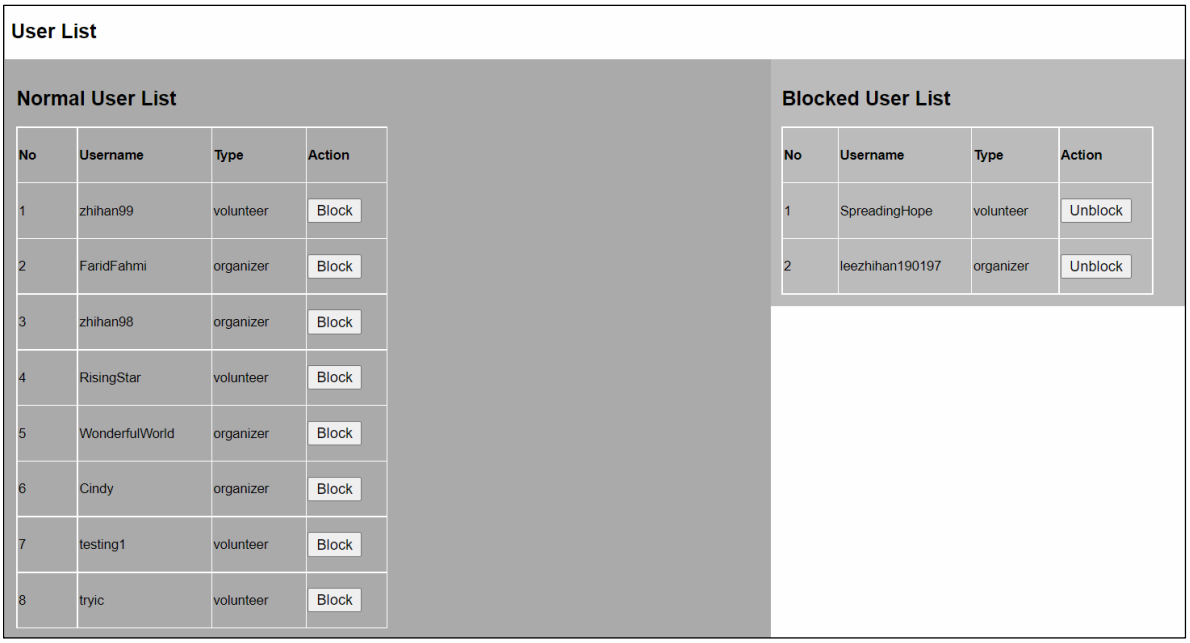

Figure 21: Manage User Interface

```
protected void BlockUsers(object sender, EventArgs e)
{
    string newstatus = "Blocked";
    GridViewRow grd = (GridViewRow)(sender as Control).Parent.Parent;
    Label lb1 = GridView1.Rows[grd.RowIndex].Cells[1].FindControl("Label2") as Label;
    string username = lb1.Text;
    SqlConnection con = new SqlConnection("Data Source=.\\SQLEXPRESS;Initial Catalog=CharityMS;Integrated Security=True; Pooling = False");
    DateTime block = DateTime.Now;

    try
    {
        con.Open();
        //Response.Write("Successful\\n");
        SqlDataAdapter cmd = new SqlDataAdapter();
        cmd.UpdateCommand = new SqlCommand("UPDATE Users SET Status=@newstatus, BlockDate=@blockdate WHERE Username=@Username", con);
        cmd.UpdateCommand.Parameters.AddWithValue(@"newstatus", newstatus );
        cmd.UpdateCommand.Parameters.AddWithValue(@"blockdate", block);
        cmd.UpdateCommand.Parameters.AddWithValue(@"Username", username);

        cmd.UpdateCommand.ExecuteNonQuery();
        con.Close();

        SqlDataSource1.DataBind();
        SqlDataSource2.DataBind();
        GridView1.DataBind();
        GridView2.DataBind();
    }
    catch (Exception ex)
    {
        Response.Write(ex.ToString());
    }
}
```

Figure 22: Code Segment to Block User

Figure 20 show the management of the event details on organizer side. Organizer can edit the event details, manage the expenses, and update the attendance of participant.

5.2 User Acceptance Test (UAT)

User acceptance test was carried out to find out whether the proposed system fulfill the user requirements. A questionnaire was designed was distributed to the users of the system. There were 21 respondents answer the questionnaire and out of them there were 2 admin, 13 volunteers, and 6 organizers. The questionnaire focus on the aspect of user interfaces and functionalities of module in the Charity Management and were rated from 1 which is very unsatisfied or strongly disagree to 5 very satisfied or strongly agree. Table 6 show the overall result of the user acceptance test form based on user interface while Table 7 show the result based on the functionalities of the system.

Table 6: User Acceptance Test Result Based On User Interface

No	Terms	Rankings					Total Respondent
		1	2	3	4	5	
	User Interface (All User)						
1	The user interface design in the system is attractive.	0	0	5	8	8	21
2	The interfaces of the text (font size, font color) are suitable with this system	0	0	1	9	11	21
3	The interface of the system is easy to understand	0	0	2	10	9	21
4	The interface of the system is easy to use	0	0	1	10	10	21
5	The navigation bar in the system are clear	0	0	4	9	8	21

Table 7 : User Acceptance Test Result Based On Functionalities

No	Terms	Rankings					Total Respondent
		1	2	3	4	5	
	Login and Registration Module						
1	Login process in this system	0	0	1	3	17	21
2	Registration process in this system (Volunteer and Organizer)	0	0	3	6	10	19
	Event Module						
1	View Event and Fundraising List	0	0	4	12	5	21
2	View Event and Fundraising Details	0	0	2	10	9	21
3	View Post and Comment Box	0	0	4	8	9	21
4	Participation Process (Volunteer Only)	0	0	0	8	5	13
5	Donation Process (Volunteer Only)	0	0	2	2	9	13
	Manage Event Module						
1	Create, edit, and delete charity event or fundraising (Organizer Only)	0	0	2	3	1	6
2	Manage costing of the event (Organizer Only)	0	0	0	2	4	6
3	Manage post (Organizer Only)	0	0	2	3	1	6
4	Participant or donor list (Organizer Only)	0	0	0	2	4	6

Table 7 : User Acceptance Test Result Based On Functionalities (cont.)

5	Update Attendance of Participant (Organizers only)	0	0	0	3	3	6
	Report Module						
1	Generate report of event or fundraising details (Organizers only)	0	0	1	4	1	6
2	Generate report for list of event and fundraising (admin only)	0	0	0	2	0	2
	Contact Us Module						
1	Send feedback or complaint function (Volunteers and Organizers)	0	0	0	5	14	19
2	View feedback list (Admin only)	0	0	0	2	0	2
	User Management Module						
1	View user list	0	0	1	1	0	2
2	Block or unblock user	0	0	0	1	1	2

6. Conclusion

For the suggestion on the future improvement in the charity management system, users should be able to update their profile in case of there is a change of environment in user. Beside that, notification should also be implemented to notify users for example volunteers will receive notification if the event that there they were participated in it was deleted by organizer.

In conclusion, all the objectives in the project were achieved successfully. With this charity management system, volunteers and organizer can play their own role in doing charity with more efficient way. Although there was some limitation and room for improvement in the system, future works can be done to improve the system performance and deliver better user experience and more convenience to the users.

Acknowledgment

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

References

[1] Cambridge Dictionary. 2021. CHAIRTY| Meaning in the Cambridge English Dictionary University. [online] Available at: <https://dictionary.cambridge.org/dictionary/english/charity>[Accessed 8 October 2021].

[2] Paul Selva Raj. 2021. LETTER | Economic impact of Covid-19 on consumers severe. [online] Available at: <https://www.malaysiakini.com/letters/587364> [Accessed 10 October 2021].

[3] X. Sun, G. Yang and J. Zhang, "A Real-time Detection Scheme of User Behavior Anomaly for Management Information System," 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2020, pp. 1054-1058, doi: 10.1109/ITNEC48623.2020.9084982.

[4] Anyim, W. O., "A Literature Review of Management Control System in University". 4197. September 2020. [Online]. Available: Library Philosophy and Practice (e-journal), https://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=7842&context=libphilprac [Accessed November 17, 2021].

[5] L. Wan, "Digital Web Archive Collection Framework Based on Intelligent Management Information System and Optimization," 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 1-4, doi: 10.1109/ICICCS51141.2021.9432158.

[6] S. Amann, S. Proksch, S. Nadi and M. Mezini, "A Study of Visual Studio Usage in Practice," 2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER), 2016, pp. 124-134, doi: 10.1109/SANER.2016.39.

[7] O. Sabri and F. Alfifi, "Integrating knowledge life cycle within software development process to produce a quality software product," 2017 International Conference on Engineering and Technology (ICET), 2017, pp. 1-7, doi: 10.1109/ICEngTechnol.2017.8308172.

[8] Facebook Help Center. 2021. Creating A Fundraiser. [online] available at: <https://www.facebook.com/help/356680401435429> [Accessed 12 October 2021].

[9] GiftBox System. 2021. GiftBox System - Intuitive Charity Management Software. [online] available at: <https://www.giftbox.systems>[Accessed 12 October 2021].

[10] CharityEngine. 2020. Donation Management. [online] available at : <https://charityengine.net/DonationManagement>[Accessed 12 October 2021].

[11] Agile Case. 2021. Charity Management Software. [online] available at: <https://www.agilecase.com/charity-management-software>[Accessed 13 October 2021].

[12] Raklet. 2021. Nonprofit Software Raklet. [online] available at: <https://hello.raklet.com/non-profit-software/?utm_source=capterra&utm_source=capterra&utm_medium=cpc&utm_campaign=non-profit&utm_content=donation-management&utm_term=>[Accessed 13 October 2021].

[13] Naga malleswari, D., Pavan Kumar, M., sathvika, D., & Ajay Kumar, B, " A Study on SDLC For Water Fall and Agile," 2018 International Journal of Engineering & Technology, 2018, pp. 10-13, doi: http://dx.doi.org/10.14419/ijet.v7i2.32.13516.

[14] Gunawardhana, L, "Process of Requirement Analysis Link to Software Development," 2019 Journal of Software Engineering and Applications, 2019, 12, pp. 406-422. doi: 10.4236/jsea.2019.1210025.

[15] TANABE, M., & KOBAYASHI, N., "A method to visualize the Scope with no Data Leakage: Context Diagram and Assurance Cases should Do," in 41st Annual Conference of Japan Creativity Society, Ishikawa, Japan, September 2019.