

VIETNAM GENERAL CONFEDERATION OF LABOUR  
TON DUC THANG UNIVERSITY  
FACULTY OF INFORMATION TECHNOLOGY



NGUYEN KIEU THUY TIEN – 521H0313  
LE CAM TU – 521H0426  
MAI THANH BINH – 521H0439

## CUSTOMER EVENT MANAGEMENT AND NOTIFICATION SYSTEM

### FINAL REPORT

ENTERPRISE SERVICE-ORIENTED ARCHITECTURE

HO CHI MINH CITY, YEAR 2024

VIETNAM GENERAL CONFEDERATION OF LABOUR TON  
DUC THANG UNIVERSITY  
FACULTY OF INFORMATION TECHNOLOGY



NGUYEN KIEU THUY TIEN – 521H0313  
LE CAM TU – 521H0426  
MAI THANH BINH – 521H0439

## CUSTOMER EVENT MANAGEMENT AND NOTIFICATION SYSTEM

### FINAL REPORT

ENTERPRISE SERVICE-ORIENTED ARCHITECTURE

Advised by  
Mr. Duong Huu Phuc

HO CHI MINH CITY, YEAR 2024

## **ACKNOWLEDGMENT**

With heartfelt gratitude, we would like to express our deepest appreciation to Mr. Duong Huu Phuc. His unwavering dedication and commitment in guiding us have been truly remarkable. He has imparted invaluable lessons that are not only theoretical but also practical, directly applicable to real-world projects. The knowledge and skills he has shared have broadened our horizons, enabling us to grasp the latest trends in the field.

We would also like to extend our sincere thanks to Ton Duc Thang University. The university has provided an optimal learning environment where we could grow holistically, from academic knowledge to life skills. The university has ceaselessly offered us opportunities for learning, experiencing, and personal development.

Ho Chi Minh City, Day 08 Month 04 Year 2024

*Your Full Name*

## **DECLARATION OF AUTHORSHIP**

We solemnly affirm that this report is a product of our own efforts, guided and supervised by Mr. Duong Huu Phuc. The work and findings contained herein are original and have not been submitted elsewhere for any other purposes. The data and figures used in this report, intended for analysis, commentary, and evaluation, are derived from various sources and have been appropriately acknowledged in the reference section.

Furthermore, any comments, reviews, and data borrowed from other authors and organizations have been duly recognized and explicitly cited. We accept full accountability for any fraudulent activity detected in our report. Ton Duc Thang University holds no responsibility for any copyright infringement that may arise from my work.

Ho Chi Minh City, Day 08 Month 04 Year 2024

*Your Full Name*

## **ABBREVIATIONS**

TDTU	Ton Duc Thang University
ERD	Entity Relationship Diagram
PDM	Physical Data Model
CRUD	Create, Read, Update, Delete

# 1 CHAPTER 1. SYSTEM OVERVIEW

## 1.1 Introduction

Our team is diligently working on a unique project titled "Customer Event Management and Notification System". This system is designed to send notifications to customers based on their event activity preferences.

The primary function of this system is to automatically send notifications to users upon their registration for an event. This ensures that users receive timely and accurate updates about the event activities they are interested in.

Furthermore, the system offers additional functionalities such as user information editing, email confirmation for event registration, and a feature that allows users to review their registered events. Users with the Admin role have the privilege to send event notifications to desired emails.

In addition to these features, our system also incorporates a survey functionality. Users with the User role can participate in system surveys, enabling the Admin to gather data and initiate suitable events based on the survey results. Admin users can view graphical statistics on the total number of participants for free events and paid events.

With the "Event Notification System", we aim to provide users with an optimal experience in participating and tracking events. We believe that this system will significantly enhance user engagement and satisfaction.

## 1.2 System Specification

Our team is committed to the development of a unique project, the "Customer Event Management and Notification System". This innovative system is designed with precision to manage and dispatch event notifications to customers based on their registered interests. The system architecture is divided into two roles: Admin and User, each equipped with distinct functionalities and responsibilities. Additionally, our system integrates a survey functionality, allowing users to participate in system surveys. This enables the Admin to collect data and initiate appropriate events based on the survey outcomes. Admin users can also access graphical statistics on the total number of participants for both free and paid events. With this comprehensive system, we aim to offer users an unparalleled experience in participating and tracking events, thereby significantly enhancing user engagement and satisfaction.

### 1.2.1 For Users

Our system presents an intuitive interface, allowing users to effortlessly browse through a comprehensive list of ongoing events. Users can delve into the specifics of each event and register for those that spark their interest. However, to ensure a secure and personalized experience, users are required to log in or register an account before they can sign up for an event.

Upon registration for an event, the system promptly sends an email to confirm the user's participation. This email also serves as a convenient reference,

providing all the necessary event details. Moreover, users have the freedom to view all the events they have registered for at any time, offering them a clear overview of their event schedule. The system also enables users to update their personal information, ensuring their profile remains current.

Events are categorized into two types: free and paid. For free events, users can register to participate without incurring any costs. For paid events, users select the event they wish to attend, register to purchase tickets, and proceed with online payment. The ticket confirmation is then sent to the user's email. Additionally, users can participate in surveys to provide information about their event preferences. This information is collected by the admin to create better and more suitable events for users in the future.

### **1.2.2 For Admin**

Admins are equipped with comprehensive capabilities to manage a diverse range of events. They have the authority to execute create, read, update, and delete (CRUD) operations on both free and paid events. This enables them to launch new events, access details of existing events, modify event information, and even eliminate events from the system in line with evolving demands and trends.

For free events, admins can effortlessly set them up, enabling users to register without any financial obligations. This fosters broader participation from users who are interested in the event but may not have the resources or the inclination to pay for participation. It also serves as an excellent platform for organizations or individuals who wish to disseminate knowledge, ideas, or experiences without a commercial intent.

Conversely, for paid events, admins can establish a secure and efficient payment gateway for users to purchase tickets. This entails setting up a reliable and user-friendly interface where users can select the event they are interested in, choose the number of tickets they wish to purchase, and make the payment securely. The system ensures that the payment process is smooth and hassle-free, thereby enhancing the user experience.

Once the payment is successful, the system automatically dispatches a confirmation to the user's registered email, providing them with the details of the event and the purchased ticket. This serves as a proof of purchase and can be used by the user to gain entry into the event.

In terms of user account management, admins can view a detailed list of registered user accounts along with the events they have participated in. They have the authority to delete or edit user accounts, ensuring the user database is accurate and relevant. However, to maintain the integrity of user data, admins are prohibited from creating user accounts.

One of the standout features for admins is the ability to dispatch event notifications to selected user emails. This allows admins to target specific users with notifications about events they might be interested in, thereby enhancing user engagement and satisfaction.

Moreover, the system also provides admins with valuable insights into the popularity and reach of the events. They can view statistics on the total number

of attendees for free and paid events, and compare these numbers to identify trends and patterns. This data can be used to make informed decisions about future events, ensuring they are aligned with user preferences and interests.

In conclusion, the "Event Notification System" is designed to revolutionize the way users and admins manage and participate in events. By providing a seamless and personalized experience, we believe this system will significantly enhance user engagement and satisfaction.

### 1.3 Scope of Topic

#### 1.3.1 Limited Objects and Functions

**For Administrators** Administrators can only delete, edit, and view the event registration history of user accounts, they cannot create accounts for users. Users will initialize themselves on the system. It is not possible to send event notifications to users' Skype, phone number, Zalo, etc. The system does not have a function to track whether users attend the event or not.

**For Users** The profile has not added an avatar, cannot register for events and receive notifications via Skype, phone number, Zalo, etc. It is not possible to check-in or notify the system that you have attended the event. Users also cannot update their event participation status on the system. All personal information and activities on the system are confidential and only the user has the right to decide to share their information.

#### 1.3.2 Technology Limitations

There are some limitations when using higher and more secure technologies due to certain issues. Currently, the system is being developed using MongoDB, HTML/CSS, and Nodejs.

In addition, transitioning to new technology can pose compatibility issues and requires significant time and resources to implement. This may include re-training teammate, updating systems, and testing to ensure everything operates smoothly. Moreover, maintaining and updating new technology can also incur additional costs.

However, the inability to use higher and more secure technologies does not mean the system is not secure. MongoDB, HTML/CSS, and Nodejs are popular and well-tested technologies that can provide a secure and efficient system if implemented correctly. The most important thing is to implement good security measures such as data encryption, using strong passwords, and regular updates to ensure the system is always protected.

### 1.4 Practical Significance

The "Customer Event Management and Notification System" our team is working on has practical implications in various areas:

**Improved User Experience** By providing users with timely and accurate updates about the events they are interested in, the system enhances the user experience. The ability for users to review their registered events and receive confirmation emails provides convenience and assurance.

**Efficient Event Management** For administrators, the system offers a comprehensive platform for event management. The ability to view user registration details, manage user accounts, and control event details allows for efficient and effective event management. In addition, the system provides survey management features to generate appropriate data for future events. The management of organizations also plays a crucial role in initiating events. Viewing statistics will help administrators identify which events have the most participants, enabling them to make informed decisions in the future.

**Targeted Communication** The system's ability to send event notifications to selected user emails allows for targeted communication. This can improve user engagement and satisfaction by providing users with relevant information.

**Security and Personalization** The system ensures security by requiring users to log in or register an account before registering for an event. It also allows users to edit their personal information, providing a personalized experience.

**Practical Applications** The system can be highly beneficial for various organizations such as businesses and universities. For instance, the Ho Chi Minh City University of Technology (HCMUT) is currently utilizing this system for managing and monitoring event participation. This practical application demonstrates the system's versatility and effectiveness. This system has the potential

to revolutionize event management and participation, providing a seamless and personalized experience for both users and administrators. It can significantly enhance user engagement and satisfaction, making it a valuable tool in any setting where event management and notifications are essential.

## 1.5 Report layout

## **2 CHAPTER 2. SYSTEM ANALYSIS AND DESIGN**

### **2.1 System Functional Requirements**

#### **For Users:**

- Register account
- Login account
- View Free Event and view Paid Event
- Register to participate in the event
- Buy tickets to attend the event
- Receive email notifications when there are new events or changes to registered events
- Edit personal information
- View a list of participated events
- View Survey list
- Take a survey

#### **For Admin:**

- View, edit, and delete user accounts
- Create, view, edit, and delete events: Free event and Paid Event
- Send event notifications to user emails
- View a list of events that users participate in
- View a list of participants in events
- Create, view, edit, and delete organizations
- Create, view, edit, and delete survey
- View Events statistics ( Free Event and Paid Event )
- View survey statistics

## 2.2 Non-functional Requirements of the System

- **Reliability:** The system must operate stably and continuously, ensuring that users can always access and use the system's functions without encountering problems or interruptions.
- **Performance:** The system must respond quickly, providing real-time event information and notifications.
- **Scalability:** The system must be designed to scale and adapt to the growth of users, events and surveys.
- **Usability:** The system's user interface must be user-friendly and easy to use, helping users to easily understand and use the system effectively.
- **Data Security:** The system must protect user data from attacks and information leaks. This may include encrypting data, using security measures like firewalls, and conducting regular security checks to ensure the system is always protected.

## 2.3 System Function Diagram

### 2.3.1 Use-Case Diagram

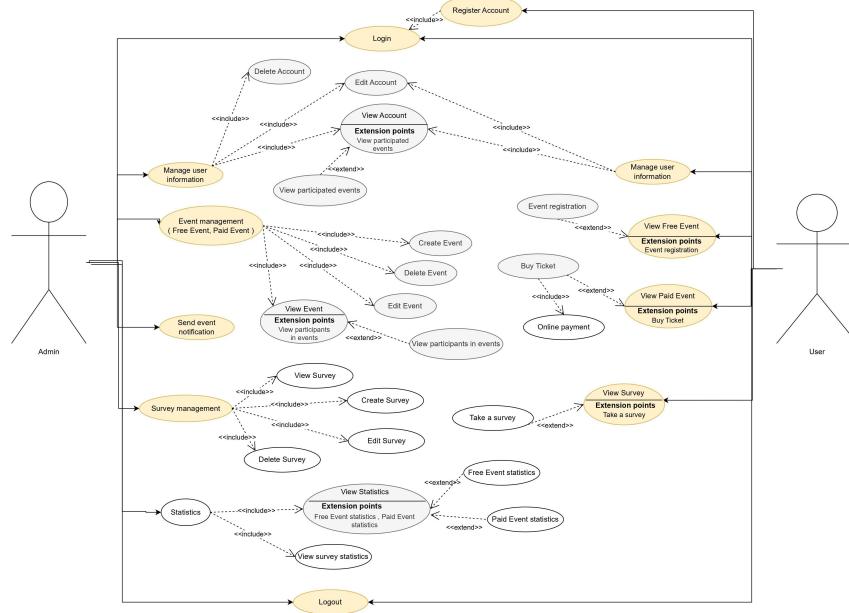


Figure 1: System usecase.

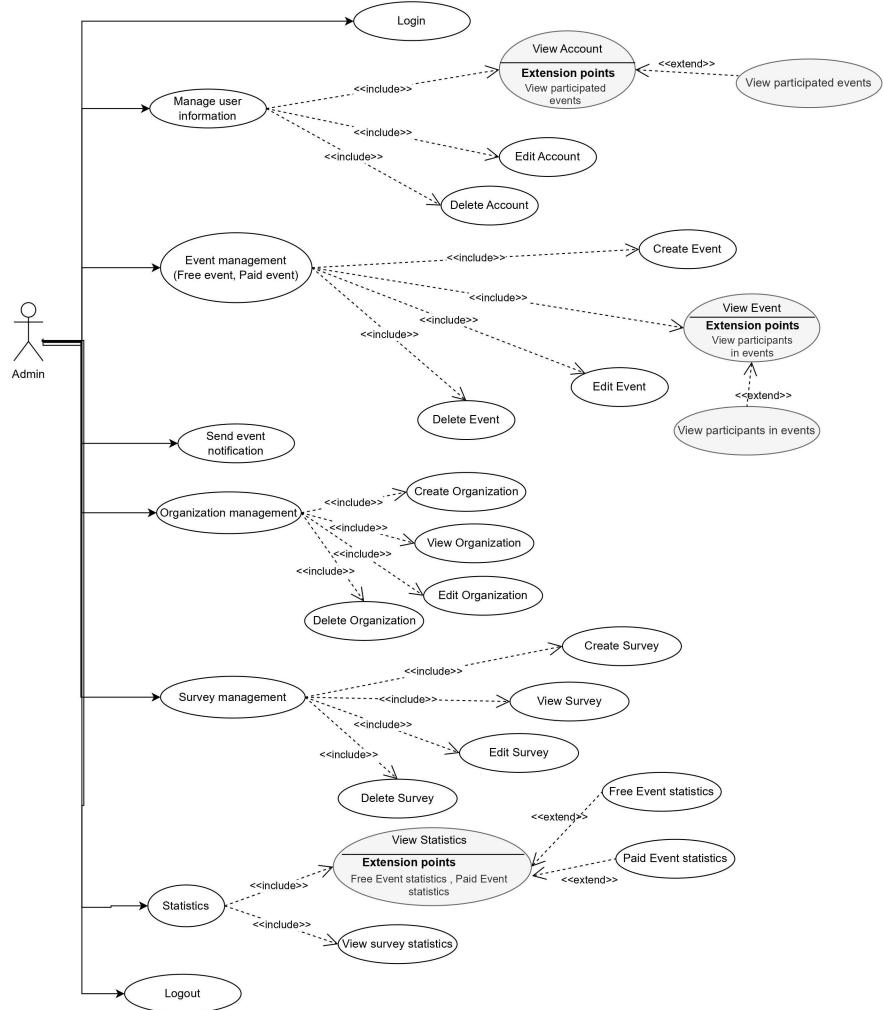


Figure 2: Usecase for admin.

### 2.3.2 Use-case description

#### 2.3.2.1 Use-cases for Admin

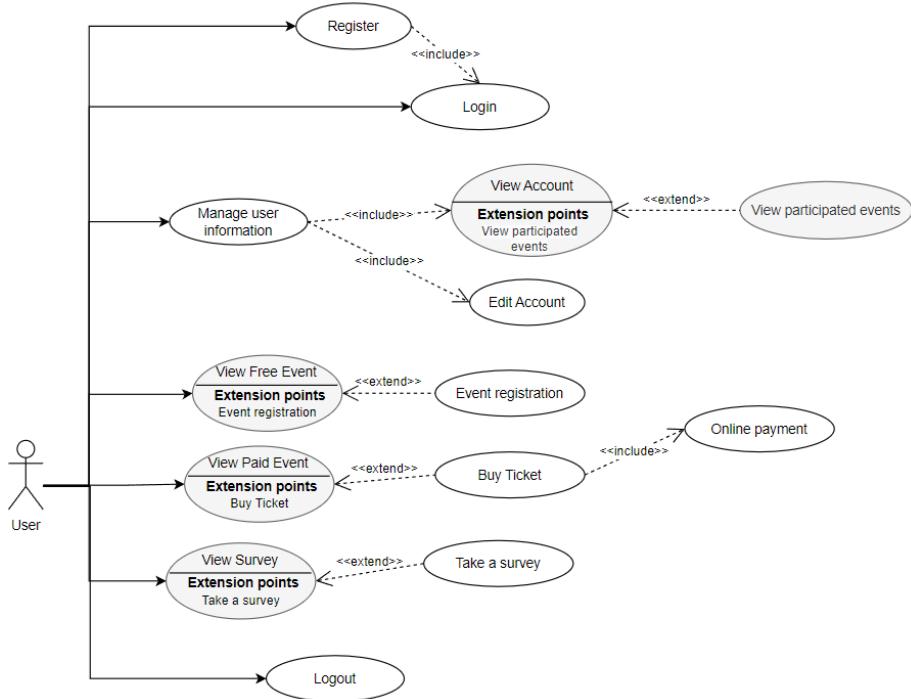


Figure 3: Usecase for user.

<b>Use Case Name</b>	View, Edit, and Delete User Accounts
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the admin to view, edit, and delete user accounts.
<b>Triggering Action/Actor Initiating Use Case</b>	Admin accesses the system.
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	Admin views the list of user accounts, selects an account to edit or delete. The system updates the information and confirms with the admin.
<b>Alternative Flow of Events</b>	None

Table 1: Usecase manage Accounts (RUD) for Admin

<b>Use Case Name</b>	Manage Events ( Free event - Paid event )
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the admin to create, view, edit, and delete events.
<b>Triggering Action/Actor Initiating Use Case</b>	Admin accesses the system.
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	Admin creates a new event or selects an event to edit or delete. The system updates the information and confirms with the admin.
<b>Alternative Flow of Events</b>	None

Table 2: Usecase Manage Event of Admin

<b>Use Case Name</b>	Send Event Notifications to User Emails
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the admin to send event notifications or notice to user emails.
<b>Triggering Action/Actor Initiating Use Case</b>	Admin selects an event (or not) and a list of user emails, then writes the content of the notification that the admin wants to send.
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	The system sends event notifications or notices to user emails and confirms with the admin.
<b>Alternative Flow of Events</b>	None

Table 3: Usecase Send event notifications to user emails for admin.

<b>Use Case Name</b>	Manage survey
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the admin to create, view, edit, and delete surveys.
<b>Triggering Action/Actor Initiating Use Case</b>	Admin accesses the system.
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	Admin creates a new survey or selects an survey to edit or delete. The system updates the information and confirms with the admin.
<b>Alternative Flow of Events</b>	None

Table 4: Usecase manage survey.

<b>Use Case Name</b>	Manage organization
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the admin to create, view, edit, and delete organization.
<b>Triggering Action/Actor Initiating Use Case</b>	To assign an organization tag to an event.
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	Admin creates a new organization or selects an organization to edit or delete. The system updates the information and confirms with the admin.
<b>Alternative Flow of Events</b>	None

Table 5: Usecase manage organization.

<b>Use Case Name</b>	View Events Statistics
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	View statistics on the total number of attendees for free events and paid events.
<b>Triggering Action/Actor</b>	Compare total attendees between Free Events and Paid Events.
<b>Initiating Use Case</b>	
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	Admin tracks the total number of people who register or buy tickets to free and paid event events.
<b>Alternative Flow of Events</b>	None

Table 6: Usecase view events statistics.

<b>Use Case Name</b>	View survey Statistics
<b>Primary Actor</b>	Admin
<b>Secondary Actor</b>	None
<b>Description</b>	View survey response statistics.
<b>Triggering Action/Actor</b>	Compare the most and least selected responses of surveys.
<b>Initiating Use Case</b>	
<b>Relationships of Use Case</b>	None
<b>Main Flow of Events</b>	Admin statistics of the total number of questions selected by users of surveys.
<b>Alternative Flow of Events</b>	None

Table 7: Usecase view survey statistics.

### 2.3.2.2 Use-cases for Users

<b>Use Case Name</b>	Register Account
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	System
<b>Description</b>	User registers to create an account to log into the system.
<b>Triggering Action/Actor Initiating Use Case</b>	User wants to create an account to log in to use the system
<b>Relationships of Use Case</b>	Include to "Login".
<b>Main Flow of Events</b>	If the user does not have an account, the user will register an account and then use the newly registered account to log in to the system.
<b>Alternative Flow of Events</b>	If the user has used email to register an account before, it will display an error and ask the user to use another appropriate email.

Table 8: Usecase Register Account for User

<b>Use Case Name</b>	Log In
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the user to log into the system.
<b>Triggering Action/Actor Initiating Use Case</b>	User selects log in on the system interface.
<b>Relationships of Use Case</b>	include from "Register"
<b>Main Flow of Events</b>	User enters login information, system authenticates and allows user access.
<b>Alternative Flow of Events</b>	If login information is incorrect, the system displays an error message and asks the user to re-enter.

Table 9: Usecase Login for User

<b>Use Case Name</b>	View Event List ( Free Event and Paid Event )
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the user to view a list of ongoing events.
<b>Triggering Action/Actor Initiating Use Case</b>	User accesses the system.
<b>Relationships of Use Case</b>	Include from "Log In" and "Free event" extend to "Register for Event", "Paid event" extend to "Buy tickets"
<b>Main Flow of Events</b>	The system displays the event list.
<b>Alternative Flow of Events</b>	None

Table 10: Use-case: View Event for User.

<b>Use Case Name</b>	Register for Event
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the user to register for events.
<b>Triggering Action/Actor Initiating Use Case</b>	User selects an event they want to attend from the free event list.
<b>Relationships of Use Case</b>	Extend from "View Free Event List".
<b>Main Flow of Events</b>	The system asks the user to confirm personal information. The user confirms and the system sends a registration confirmation email.
<b>Alternative Flow of Events</b>	None

Table 11: Usecase Register for Event for User

<b>Use Case Name</b>	Buy tickets to attend the event
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allows users to buy tickets to attend events and pay online.
<b>Triggering Action/Actor Initiating Use Case</b>	User selects an event they want to attend from the paid event list.
<b>Relationships of Use Case</b>	Extend from "View Paid Event List".
<b>Main Flow of Events</b>	The system requires users to select an online payment type to be able to buy tickets to paid events.
<b>Alternative Flow of Events</b>	None

Table 12: Usecase Buy tickets to attend the event

<b>Use Case Name</b>	View Survey List
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the user to view a list of ongoing surveys.
<b>Triggering Action/Actor Initiating Use Case</b>	User accesses the system.
<b>Relationships of Use Case</b>	Include from "Log In"
<b>Main Flow of Events</b>	The system displays the survey list.
<b>Alternative Flow of Events</b>	None

Table 13: Use-case: View Survey List for User.

<b>Use Case Name</b>	Take a survey
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allow users to take surveys.
<b>Triggering Action/Actor Initiating Use Case</b>	User selects the survey the user wants to take.
<b>Relationships of Use Case</b>	Extend from "View Survey List"
<b>Main Flow of Events</b>	Users take the survey they want to provide data to the system.
<b>Alternative Flow of Events</b>	None

Table 14: Use-case: Take a survey.

<b>Use Case Name</b>	Edit Personal Information
<b>Primary Actor</b>	User
<b>Secondary Actor</b>	None
<b>Description</b>	Allows the user to update their personal information.
<b>Triggering Action/Actor Initiating Use Case</b>	User accesses the account section.
<b>Relationships of Use Case</b>	Include from "Log In"
<b>Main Flow of Events</b>	User enters and saves changes to personal information. The system updates the information and confirms with the user.
<b>Alternative Flow of Events</b>	none.

Table 15: Usecase Manage Personal Information for User

## 2.4 System's Database Schema

### 2.4.1 ERD schema

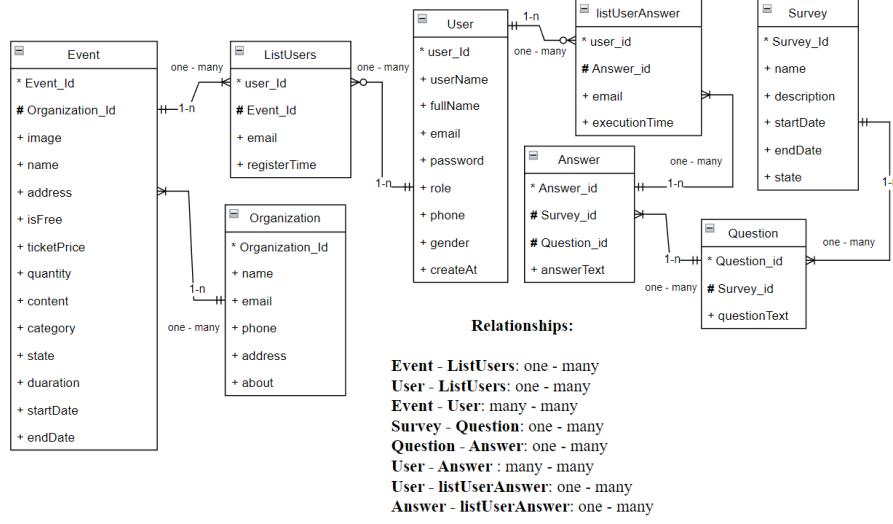


Figure 4: Entity Relationship Diagram.

An intricate many-to-many relationship exists between ‘Event’ and ‘User’. Each User has the ability to register for a multitude of Events, while an Event can be populated by registrations from a variety of Users. This dynamic interaction leads to the creation of ‘ListUsers’, a repository that holds the data of Users who register for different Events.

While an Event can be associated with multiple registered Users, its ListUsers is exclusive to that Event. Although a User can appear in several ListUsers, each ListUsers maintains a unique record of that User, ensuring there is no duplication.

In addition to the relationship between ‘Event’ and ‘User’, there are other significant relationships structuring the interaction within this system. The ‘Survey’ and ‘Question’ entities share a one-to-many relationship, meaning a single Survey can contain multiple Questions, but each Question is associated with only one Survey.

Similarly, a one-to-many relationship exists between ‘Question’ and ‘Answer’. Each Question can have multiple Answers but each Answer is linked to a single Question. A many-to-many relationship is

established between 'User' and 'Answer', indicating that Users can provide multiple Answers and each Answer can be associated with multiple Users.

'listUserAnswer' acts as a junction table facilitating the many-to-many relationship between User and Answer, ensuring data integrity and relational flexibility. Each User's answers are uniquely recorded in 'listUserAnswer', which maintains a one-to-many relationship with both User and Answer entities respectively.

## 2.4.2 Physical Database Schema

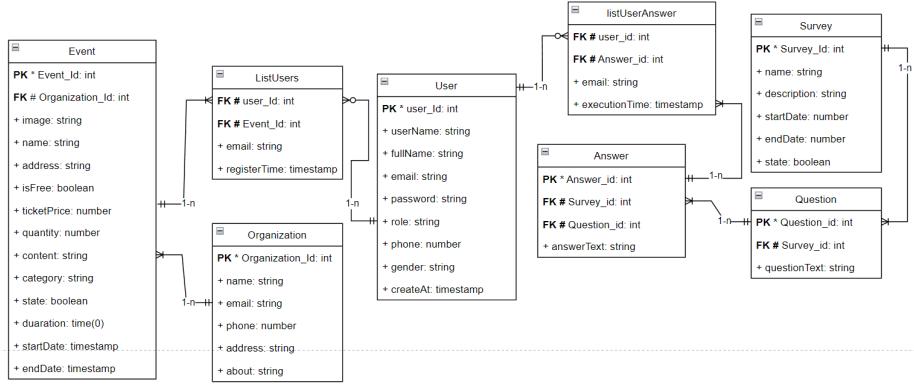


Figure 5: Physical Database Schema.

### 3 CHAPTER 3. SYSTEM IMPLEMENTATION

Our team has chosen **MongoDB**, **HTML/CSS**, and **Node.js** as the primary technologies for the development of the "Customer Event Management and Notification System". This decision was not made lightly, but rather after careful consideration of the advantages these technologies offer.

**MongoDB** is a source-available cross-platform document-oriented database program. It is classified as a NoSQL database program, which allows for high volume data storage. This makes it a good choice for systems that need to handle large amounts of data, such as our event management system. Furthermore, MongoDB is known for its flexibility, as it doesn't require a unified data structure across all objects, making it easier to develop and evolve our system over time.

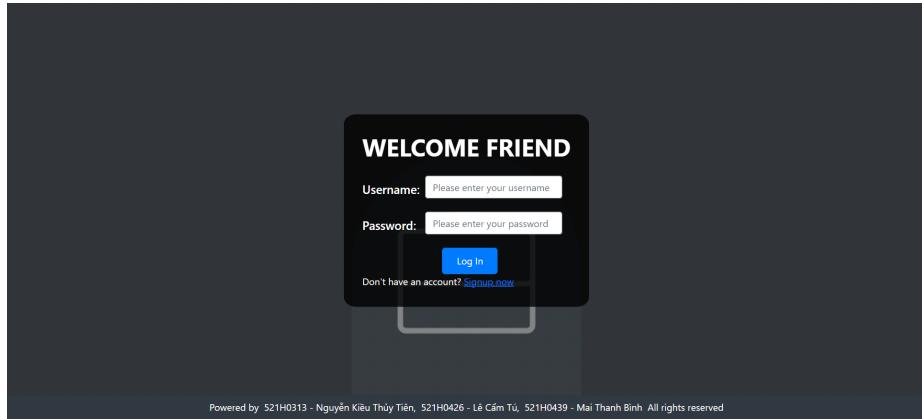
**HTML (HyperText Markup Language)** and **CSS (Cascading Style Sheets)** are cornerstone technologies for building web pages and web applications. HTML provides the structure of the page, while CSS styles the content. HTML and CSS are essential for creating the user interface of our system. They are widely used, well-documented, and supported by all modern web browsers, making them a practical choice for our project.

**Node.js** is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js is perfect for developing server-side and networking applications. It's designed to build scalable network applications, which is crucial for our system as it needs to handle multiple user requests simultaneously. Moreover, since JavaScript is used for both front-end and back-end, it enhances development speed and efficiency.

One of the main reasons for choosing these technologies is their practicality and ease of use. All team members are already familiar with these technologies, which eliminates the need for extensive learning or research. This familiarity allows us to leverage these technologies effectively, accelerating the development process and enhancing the system's performance. Furthermore, these technologies are widely used and supported, which means we'll find a wealth of resources and community support available, should we encounter any issues or require advanced functionalities. This combination of technologies is well-suited to create a robust and efficient "Customer Event Management and Notification System".

## 4 CHAPTER 4. DEMO

### 4.0.1 Login Interface



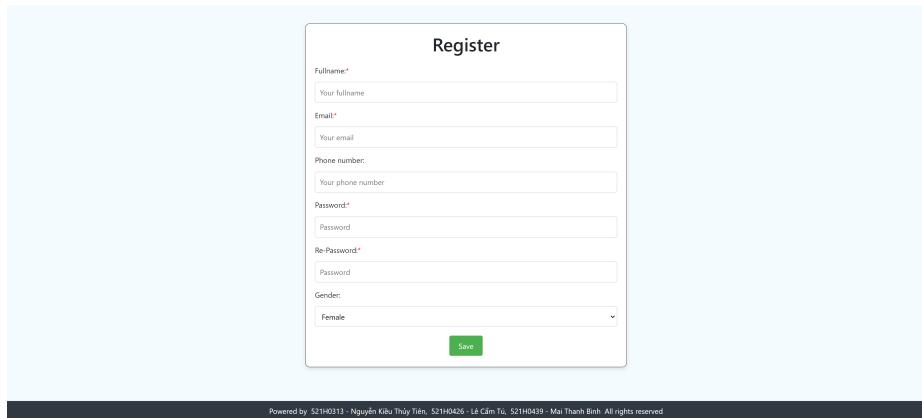
The screenshot shows a dark-themed login form titled "WELCOME FRIEND". It contains fields for "Username" and "Password", both with placeholder text "Please enter your username" and "Please enter your password" respectively. Below these fields is a blue "Log In" button. At the bottom of the form, there is a link "Don't have an account? [Signup now](#)". The background of the page is dark, and at the very bottom, there is a footer bar with the text "Powered by 521H0313 - Nguyễn Kiều Thùy Tiên, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình All rights reserved".

**URL:** <http://localhost:3000/auth/login>

The Login screen is used for both roles: Admin and User to log into the system

### 4.1 For User

#### 4.1.1 Register Interface

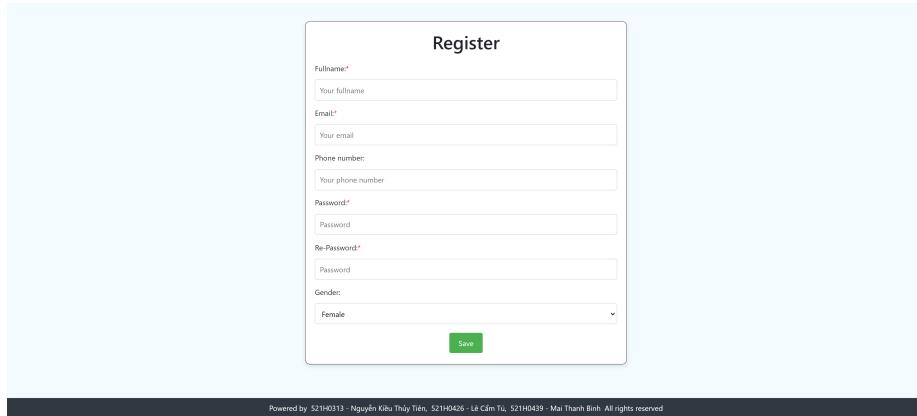


The screenshot shows a registration form titled "Register". It includes fields for "Fullname\*", "Email\*", "Phone number", "Password\*", "Re-Password\*", "Gender", and a dropdown menu showing "Female". There is also a "Save" button at the bottom right. The background is light blue, and the footer bar at the bottom contains the text "Powered by 521H0313 - Nguyễn Kiều Thùy Tiên, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình All rights reserved".

**URL:** <http://localhost:3000/auth/register>

If the user does not have an account to access the system, the user needs to register by filling in the information fields, then register an account and perform the login step to access the system

#### 4.1.2 Profile Interface



Register

Fullname\*  
Your fullname

Email\*  
Your email

Phone number:  
Your phone number

Password\*  
Password

Re-Password\*  
Password

Gender:  
Female

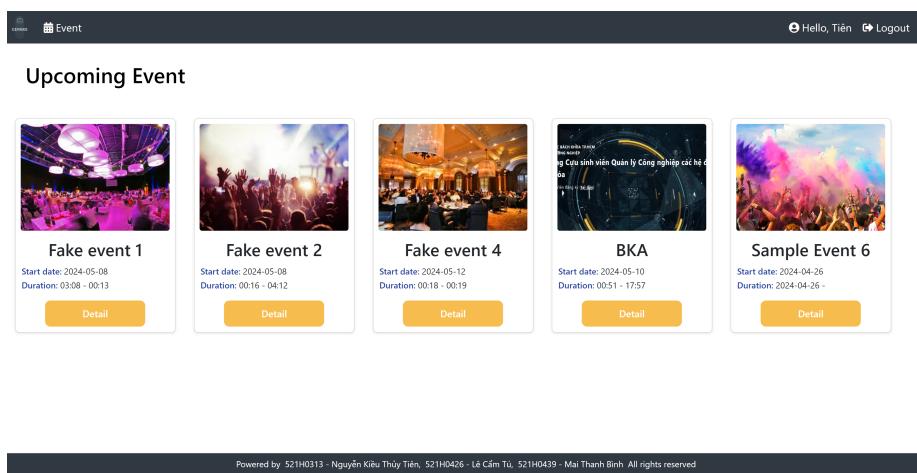
Save

Powered by 521H0313 - Nguyễn Kiều Thúy Tiên, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình. All rights reserved

**URL:** <http://localhost:3000/auth/register>

If the user does not have an account to access the system, the user needs to register by filling in the information fields, then register an account and perform the login step to access the system

#### 4.1.3 Event Interface



Event

Hello, Tiên Logout

### Upcoming Event

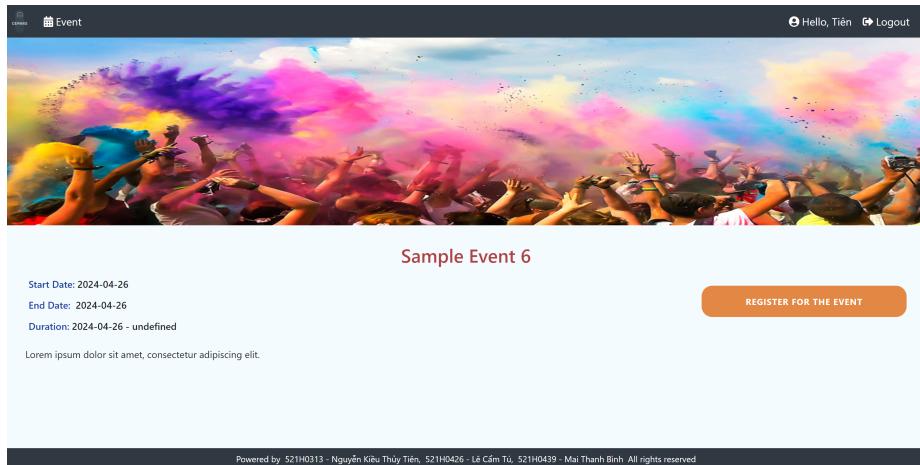
Event thumbnail	Event Name	Start date	Duration	Action
	Fake event 1	2024-05-08	03:08 - 00:13	<a href="#">Detail</a>
	Fake event 2	2024-05-08	00:16 - 04:12	<a href="#">Detail</a>
	Fake event 4	2024-05-12	00:18 - 00:19	<a href="#">Detail</a>
	BKA	2024-05-10	00:51 - 17:57	<a href="#">Detail</a>
	Sample Event 6	2024-04-26	Duration: 2024-04-26 -	<a href="#">Detail</a>

Powered by 521H0313 - Nguyễn Kiều Thúy Tiên, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình. All rights reserved

**URL:** <http://localhost:3000/userEvent>

Users can monitor open events here, from there users can select the event they like and view details of that event.

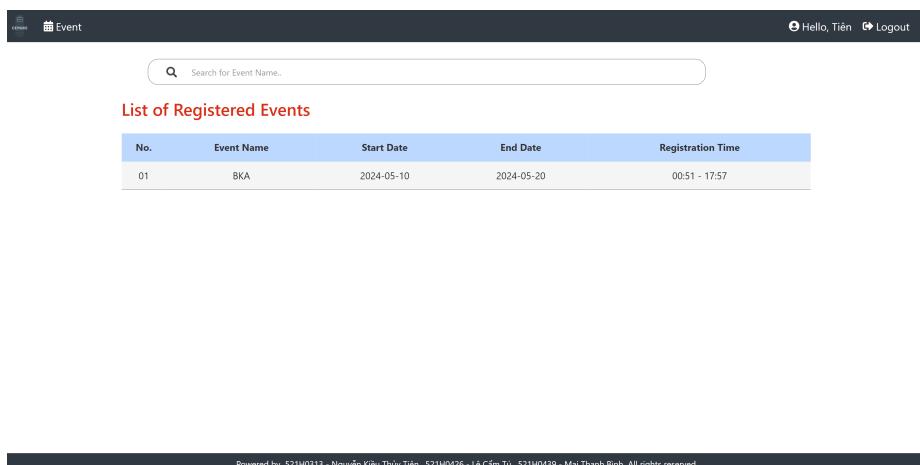
#### 4.1.4 Detail Event Interface



**URL:** <http://localhost:3000/detail/EventId>

Users can view detailed information of the event they are interested in. After that, users can also register to attend the event if needed. Event information will also be sent via the user's email

#### 4.1.5 Participated Events Interface



**URL:** <http://localhost:3000/listEvents/UserId>

Users can view the list of events and times that they have registered for

## 4.2 For Admin

### 4.2.1 Manage Event Interface

No.	Event Name	Start Date	End Date	State	Action
01	Fake event 1	2024-05-08	2024-05-09	Open	<button>Edit</button> <button>Detail</button> <button>Delete</button>
02	Fake event 2	2024-05-08	2024-05-09	Open	<button>Edit</button> <button>Detail</button> <button>Delete</button>
03	Fake event 3	2024-05-10	2024-05-11	Closed	<button>Edit</button> <button>Detail</button> <button>Delete</button>
04	Fake event 4	2024-05-12	2024-05-13	Open	<button>Edit</button> <button>Detail</button> <button>Delete</button>
05	BKA	2024-05-10	2024-05-20	Open	<button>Edit</button> <button>Detail</button> <button>Delete</button>
06	Sample Event 6	2024-04-26	2024-04-26	Open	<button>Edit</button> <button>Detail</button> <button>Delete</button>

Powered by 521H0313 - Nguyễn Kiều Thúy Tiên, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình. All rights reserved

URL: <http://localhost:3000/>

The admin will initiate events, send notifications inviting users or emails that have not registered a system account to register to attend the events that the admin wants. In addition, admins can manipulate CRUD events that they have initialized.

### 4.2.2 Participants In Events Interface

No.	Fullname	Email	Phone number	Registration Time
01	thanhbinhbinhmai	maithanhbinh13102003@gmail.com	12345	2024-05-06T17:16:49.368Z

Powered by 521H0313 - Nguyễn Kiều Thúy Tiên, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình. All rights reserved

URL: <http://localhost:3000/listUser/EventId>

Admin can see how many users registered to attend that event, the time they registered to attend, and the necessary information of that user so that the admin can easily contact them.

#### 4.2.3 Manage Account Login Interface

The screenshot shows a web application interface for managing user accounts. At the top, there is a navigation bar with links for 'Manage Event' and 'Manage Users'. On the right side of the navigation bar is a 'Logout' button. Below the navigation bar is a search bar with the placeholder text 'Search for names..'. The main content area displays a table with two rows of user data. The columns are labeled 'No.', 'Fullname', 'Email', 'Gender', 'Phone number', and 'Action'. The first row (No. 01) has data: fullname 'thanhhbinhmai', email 'maithanhbinh13102003@gmail.com', gender 'male', phone number '12345', and action buttons 'Edit', 'Detail', and 'Delete'. The second row (No. 02) has data: fullname 'Tiến', email 'thuytien2452003@gmail.com', gender 'female', phone number '113', and action buttons 'Edit', 'Detail', and 'Delete'. At the bottom of the page, there is a footer with the text 'Powered by 521H0313 - Nguyễn Kiều Thúy Tiến, 521H0426 - Lê Cẩm Tú, 521H0439 - Mai Thành Bình. All rights reserved'.

No.	Fullname	Email	Gender	Phone number	Action
01	thanhhbinhmai	maithanhbinh13102003@gmail.com	male	12345	<button>Edit</button> <button>Detail</button> <button>Delete</button>
02	Tiến	thuytien2452003@gmail.com	female	113	<button>Edit</button> <button>Detail</button> <button>Delete</button>

**URL:** <http://localhost:3000/account>

Admin can track the list of accounts created by users for convenient contact. In addition, it is possible that the user's account RUD and the admin cannot create a user account, this will be done by the user themselves.

#### 4.2.4 Participated Events Interface

**URL:** <http://localhost:3000/listEvents/UserId>

Admins can view the list of events that users have registered to participate in and track the time that they have registered for those events.

The screenshot shows a web application interface. At the top, there is a navigation bar with icons for 'CEMANS' (a logo), 'Manage Event' (with a camera icon), 'Manage Users' (with a user icon), and 'Logout'. Below the navigation bar is a search bar with the placeholder 'Search for Event Name..'. The main content area is titled 'List of Registered Events' and contains a table with four rows of data. The table has columns for 'No.', 'Event Name', 'Start Date', 'End Date', and 'Registration Time'. The data in the table is as follows:

No.	Event Name	Start Date	End Date	Registration Time
01	Fake event 2	2024-05-08	2024-05-09	00:16 - 04:12
02	Fake event 4	2024-05-12	2024-05-13	00:18 - 00:19
03	BKA	2024-05-10	2024-05-20	00:51 - 17:57
04	Sample Event 6	2024-04-26	2024-04-26	2024-04-26 - undefined

At the bottom of the page, there is a footer bar with the text 'Powered by 521H0313 - Nguyễn Kiều Thúy Tiên, 521H0426 - Lê Cảnh Tú, 521H0439 - Mai Thành Bình. All rights reserved.'

## References

- [1] BKASim.vn. (n.d.). Retrieved April 01, 2024, from <https://bkasim.vn/Event/EventAnother>