



Palestine Technical University – Kadoorie  
College of Engineering and Technology  
Department of Computer Systems Engineering

**Course name:**  
Software Engineering



**Project title:**  
**DEVELOPING AN ONLINE SERVICE FOR  
STUDENTS CLUB**

**By:**  
Huda Sawalha -202110714 - Section(3)  
Jad Atout-202110490 - Section (1)  
Razan Hamad - 202110754 - Section(3)  
Shahd Eid - 202110734 (3)

**Supervisor:**  
Dr. Nael Salman  
Dr. Osama Hamed

Tulkarm, Palestine  
Spring 2023-2024

---

# TABLE OF CONTENTS

---

Page	Topic
4	ABSTRACT
5	CHAPTER 1: INTRODUCTION
6	1.1 PROBLEM STATEMENT
7	1.2 OBJECTIVES
7	1.3 THE TARGET GROUPS OF THE PROJECT
8	CHAPTER 2: REQUIREMENTS
9	2.1 CUSTOMIZED
9	2.2 METHODOLOGY
10	2.3 PROCESS MODEL
11	2.4.1 FUNCTIONAL REQUIREMENTS
31	2.4.2 NON-FUNCTIONAL REQUIREMENTS
33	CHAPTER 3: SYSTEM ARCHITECTURE
35	CHAPTER 4: SYSTEM MODELS
36	4.1.1 USER USE CASE MODEL
37	4.1.2 ADMIN USE CASE MODEL
38	4.1.3 ADMIN/USER USE CASE MODEL

---

## TABLE OF CONTENTS

---

<b>39</b>	4.2 ADMIN ACTIVITY DIAGRAM
<b>40</b>	4.2 USER ACTIVITY DIAGRAM
<b>43</b>	4.3 CLASS DIGRAM
<b>44</b>	4.4 ER-DIAGRAM
<b>45</b>	4.5 SEQUENCE DIAGRAM
<b>45</b>	4.5.1 User Log in Sequence Diagram
<b>45</b>	4.5.2 Admin Log in Sequence Diagram
<b>46</b>	4.5.3 User subscription process
<b>47</b>	CHAPTER5:SYSTEM TESTING
<b>49</b>	5.1 TESTING ADMIN VIEW
<b>50</b>	5.2 TESTING USER VIEW
<b>58</b>	PROJECT TIMELINE
<b>59</b>	CONCLUSION

## Abstract

This project describes an international student organization website from both user and administrator perspectives. Users can create accounts, log in, register for events, and provide feedback. Administrators manage events, memberships, and financial operations efficiently and securely.

Initially, we developed a billing system as a DESKTOP APPLICATION to handle invoicing, ensuring faster and more secure management of subscriptions and billing. This desktop application served as a prototype. Later, due to client updates, we transitioned to a web-based solution, integrating the billing system with the website to provide a comprehensive and streamlined solution for the club's administrative needs.

يقوم هذا المشروع بوصف موقع نادي الطالبي العالمي يساعد الطلاب على إنشاء حساباتهم، التسجيل في الفعاليات، وتقديم تقييمات. أما المسؤولين فيساعدهم في إدارة الفعاليات، متابعة العضويات، والأمور المالية بكفاءة وأمان.

في البداية، قمنا بإنشاء نظام فواتير كتطبيق سطح المكتب لمعالجة الفوترة بسرعة وأمان أكبر. كان هذا التطبيق المكتبي بمثابة نموذج أولي. لاحقاً، وبسبب تحديثات من قبل العميل، قمنا بالانتقال إلى حل يعتمد على الويب، مدمجين نظام الفواتير مع موقع الويب لتوفير حل شامل وسلس لاحتياجات إدارة النادي.

# **CHAPTER ONE**

## **INTRODUCTION**

# INTRODUCTION

## 1.1 PROBLEM STATEMENT:

The student club faces various administrative and organizational challenges, impacting operational efficiency and member satisfaction. To address these issues, the objective is to develop a dedicated website tailored for students, aiming to streamline processes and enhance efficiency. The key problems include:

### Student Challenges:

1. **Event Registration:** Registering for events is time-consuming due to the need for direct communication with event coordinators, leading to unnecessary delays.
2. **Subscription Payments:** Manual subscription payments require direct interaction with coordinators, making the process inconvenient and prone to errors and delays.
3. **Event Tracking:** Searching for and tracking events in a specific area poses a challenge for students, reducing their participation and hindering their ability to benefit from available activities.

### Club Coordinator Challenges:

1. **Subscription Invoice Printing:** Manually printing subscription invoices consumes significant time and effort, increasing the risk of errors.
2. **Event and Participant Tracking:** Manually tracking events and participants, and communicating with them, increases the workload on coordinators and reduces event management efficiency.

## 1.2 OBJECTIVES:

1. **Simplify Event Registration:** We're developing a system that simplifies event registration, reducing the need for direct communication with event coordinators and minimizing unnecessary delays.
2. **Streamline Subscription Payments:** Our system streamlines subscription payments, eliminating the need for direct interaction with coordinators and reducing the likelihood of errors and delays.
3. **Effortless Event Tracking:** Tracking and managing events becomes effortless with our system, providing better organization and saving time while preventing unnecessary costs and delays.
4. **Enhanced Data Protection:** We prioritize data security, ensuring that invoices and data are kept secure to alleviate concerns about data loss, even in the event of technical issues.

## 1.3 THE TARGET GROUPS OF THE PROJECT:

A student club and the coordinators of the club as well. Club members are students who engage in the reception of club services and attendances of club activities, and coordinators are officials overseeing the club's activities and events subscription. Thus, by fulfilling needs of the both groups, it is possible to increase the effectiveness of club functioning and to improve the quality of offered services for members.

## **CHAPTER TWO**

### **REQUIREMENTS**

**Every new system should has new properties, in order to add some advantages for existing problems in the target area. Here in this chapter, we will make a comparison between the current system and our proposed system, to show the main characteristic for both systems. After reading this chapter, the reader will be able to distinguish the differences between the old system and the proposed system with complete understanding.**

# REQUIREMENTS

## 2.1 customized

Customer Type: Customized.

The customer: Kun team

## 2.2 Methodology

In order to get the sufficient requirements for the to be developed student club system, we have held several meetings and surveys. undefined

### **Initial Meetings with Stakeholders:**

First of all, we discussed with the main student club committee members to define the initial goal and figure out the main requirements for the system.

### **Requirement Analysis:**

In this initial stage the team collected all the requirements, then to be more precise the team analyzed the specifications so as to include all

# REQUIREMENTS

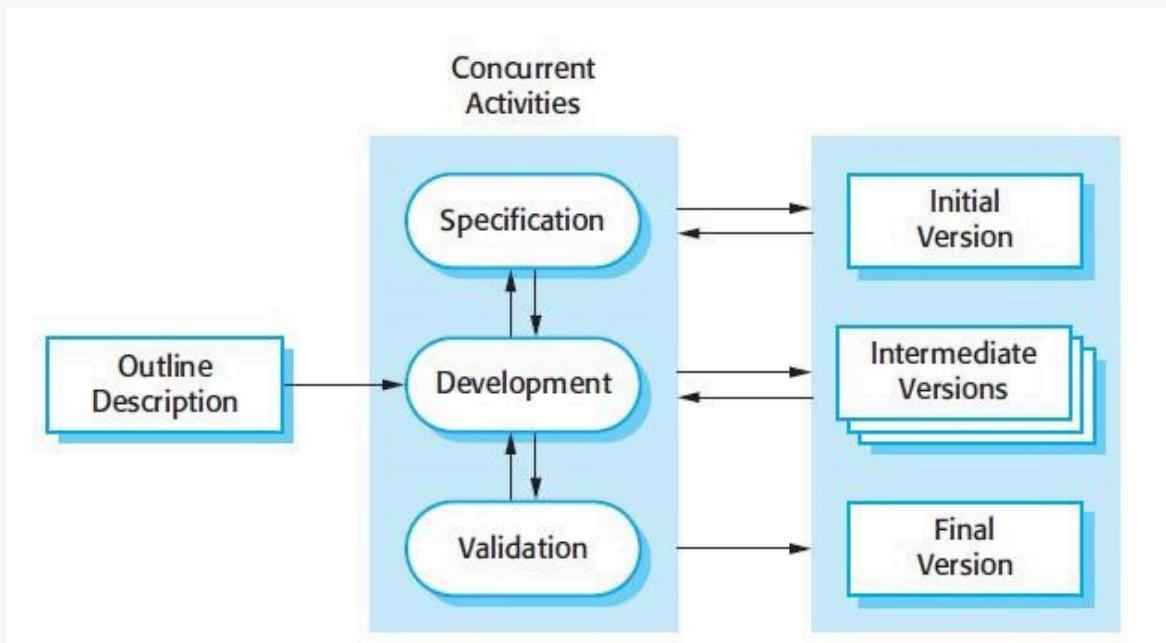
## 2.3 Process Model

### **Incremental Model.**

The incremental approach is chosen for the development of this system for many reasons.

### **Why Using This Model:**

- R1: The flexibility of this approach allows for adjusting future increments as we understand more about our user's needs.
- R2: By delivering early usable increments we can get early and frequent feedback from users and stakeholders which will improve future increments.
- R3: By focusing on a small amount of work at a time if it happens to have a problem, it will affect the increment itself, not the entire system.
- R4: If the system is not complete the users can start using the system early as increments are deployed.



# REQUIREMENTS

## 2.4 FUNCTIONAL REQUIREMENTS

### 2.4.1 User Registration

#### Description and Priority

Registering a new user on our website is designed to be simple secure and efficient. It allows the student to be a member of the club by simply providing the necessary information.

#### Stimulus/Response Sequences

**Stimulus:** Browser to the website and select “Register”

**Response:** The system will present a page where the user can enter in all the required

information which is First Name, Last Name, Student-Email, and Password, then the system will display a summary of all entered information to ensure accuracy, then a confirmation of registration will be displayed and finally a verification page will appear to tell the user to verify registration by confirming an email.

#### Functional Requirements

**REQ-1:** All the information, including the password, must be entered on a single page, the first page of registration.

**REQ-2:** The registration confirmation page must clearly show whether the registration state is successful and if there is an error message should be displayed.

**REQ-3:** The required information fields must be mandatory so the user won't be able to register without filling them out

**REQ-4:** The email must be for an educational institution and must be checked if it already exists so duplications won't occur.

**REQ-5:** The length of the password must be at least 8 characters including a mix of uppercase and lowercase letters, numbers, and special characters

**REQ-6:** Upon successful submission, the data is stored securely in our MySQL database.

# REQUIREMENTS

## 2.4 FUNCTIONAL REQUIREMENTS

### 2.4.1 User Registration

#### Description and Priority

Registering a new user on our website is designed to be simple secure and efficient. It allows the student to be a member of the club by simply providing the necessary information.

#### Stimulus/Response Sequences

**Stimulus:** Browser to the website and select “Register”

**Response:** The system will present a page where the user can enter in all the required

information which is First Name, Last Name, Student-Email, and Password, then the system will display a summary of all entered information to ensure accuracy, then a confirmation of registration will be displayed and finally a verification page will appear to tell the user to verify registration by confirming an email.

#### Functional Requirements

**REQ-1:** All the information, including the password, must be entered on a single page, the first page of registration.

**REQ-2:** The registration confirmation page must clearly show whether the registration state is successful and if there is an error message should be displayed.

**REQ-3:** The required information fields must be mandatory so the user won't be able to register without filling out

**REQ-4:** The email must be for an educational institution and must be checked if it already exists so duplications won't occur.

**REQ-5:** The length of the password must be at least 8 characters including a mix of uppercase and lowercase letters, numbers, and special characters

**REQ-6:** Upon successful submission, the data is stored securely in our MySQL database.

## chapter 2

# REQUIREMENTS

### 2.4.2 Admin Registration

#### Description

Admin registration is a crucial feature that allows the creation of an admin account on our website. This process is handled by a super admin or system administrator due to high-level access and control associated with admin accounts.

#### Stimulus/Response Sequences

**Stimulus:** The system administrator clicks on “create Admin” from his dashboard.

**Response:** The system is designed to generate a dedicated page for the creation of an Admin account. This page will feature all the necessary fields required for account creation, similar to those found in the user registration process. However, it will also include additional information, such as a ‘Rank’ field which has particular importance as it determines the level of access and control the new admin will have. It determines whether the new admin will have the authority to create more admin accounts. Once all the required fields are filled out, the system will direct the user to a confirmation page. This page serves to verify and confirm the action of creating a new admin account.

#### Functional Requirements

**REQ-1:** The email must be marked as belonging to the administration department and checked if it already exists so duplications won’t occur.

**REQ-2:** The length of the password must be changed by the real owner of the account after two login sections.

**REQ-3:** Upon successful submission, the data is stored securely in our MySQL database.

**Other Requirements:** Same as the User Registration Requirements

## chapter 2

# REQUIREMENTS

### 2.4.3 User/Admin Login

#### Description

The login feature on our website is a secure gateway to access user-specific or admin-specific resources by entering their emails and passwords

#### Stimulus/Response Sequences

**Stimulus:** Browser to the website and select “log in”

**Response:** Upon accessing the system, a uniform page will be presented to both users and admins. This page will prompt users to input their email and password. Upon clicking "continue," the system will either grant access to the user or admin's respective dashboard based on their role associated with the entered email or prompt them to reenter their information or change their password in case of a failed login.

#### Functional Requirements

**REQ-1:** In the user interface, when users enter their email address, the system should not differentiate between uppercase and lowercase letters. However, for the password field, the system should be case-sensitive, meaning it should distinguish between uppercase and lowercase letters

**REQ-2:** The system should validate the entered data against the stored one. If there is an error message should be displayed.

**REQ-3:** Upon successful login, the system should create a session for the user/admin, which shall expire after a period of inactivity.

**REQ-4:** The system will continue a “Forgot Password” option that will allow the user/admin to reset their passwords (shall be discussed in Forget Password Feature).

## chapter 2

# REQUIREMENTS

### 2.4.4 admin/user Forgot Password

#### Description

The “Forgot Password” option is a feature found on the login page. It allows the users to rest their passwords to regain access to the website

#### Stimulus/Response Sequences

**Stimulus:** On the login page Click “Forgot Password”.

**Response:** Upon accessing the system, a uniform page will be presented to both users and admins. This page will prompt users to input their email and password. Upon clicking "continue," the system will either grant access to the user or admin's respective dashboard based on their role associated with the entered email or prompt them to reenter their information or change their password in case of a failed login.

#### Functional Requirements

**REQ-1:** The “forgot Password” option should be visible and user-friendly to the users on the login page

**REQ-2:** The system should securely handle the rest password process to prevent unauthorized access to the website.

**REQ-3:** The system must verify that the entered email address is associated with an existing user.

**REQ-4:** The system should be able to send a code in an email to the users.

**REQ-5:** the code is valid for a specified amount of time.

**REQ-6:** the system should monitor the reset password attempts to set a limit on the number of password resets for a specified amount of time to prevent abuse.

## chapter 2

# REQUIREMENTS

### 2.4.4 admin/user Logout

#### Description

The “Logout” feature is an essential part of our website which is designed to allow users to securely end their current session and log out of their account which aims to protect user's data and prevent unauthorized access

#### Stimulus/Response Sequences

**Stimulus:** End of the session or the dashboard page Click the “Logout button”.

**Response:** Upon the end of the user session or clicking on the logout button the system will return the user to the login page so he can sign in again or just end his usage of the website

#### Functional Requirements

REQ-1: The system should manage user's sessions and log them out by the end of their sessions.

REQ-2: The “Logout” button should be visible for the website users

REQ-3: Upon clicking the “Logout” button, the user's session must be ended immediately, so, the session ID will be invalid.

REQ-4: The logout action must be secured to prevent attacks like Cross-Site Request Forgery by using safe tokens

REQ-5: Any sensitive data stored on the client side must be cleared when logging out.

REQ-6: The system should display a message to the user to notify him when his session ends before logging him out of the account.

## chapter 2

# REQUIREMENTS

### 2.4.5 admin/user Search for Events

#### Description

The search feature on an event will display the events of the organization and will also allow users to quickly navigate to the page of the event they are looking for. When displaying results that contain multiple organization types, the search will be performed upon the event's ID or name.

#### Stimulus/Response Sequences

**Stimulus:** On the Event's page Click "Search".

**Response:** The system will generate a page that contains a list of events that match the ID or name. The result list can be filtered based on various parameters like date, location, and type of the event

#### Functional Requirements

**REQ-1:** The website should have a visible search bar where the users can enter the name or the ID of an event.

**REQ-2:** The system should have a reliable search algorithm that can match user's events and handle typos and other search challenges.

**REQ-3:** The system should have a filter that enables users to filter their results based on various parameters such as date, location, price, and type.

**REQ-4:** The system should have a sorting option to sort the events by date, popularity (participants/likes), and price.

**REQ-5:** If there are many results the system should divide them into pages to improve user experience.

**REQ-6:** the search should be user-friendly even for first-time users. The design should aim to ease reading and understanding.

## chapter 2

# REQUIREMENTS

### 2.4.6 admin /Add Event

#### Description

The authorized admins are enabled to create and publish events on the website by the "Add Event" function. It serves as a tool in the organization and promotion of the club activities which are scheduled in the future.

#### Stimulus/Response Sequences

**Stimulus:** The clicked-in user goes to the event management area and chooses "Add Event".

**Response:** The system provides a form for the user to fill in all the event details such as title, description, date, time, location, and any other specific information. After the input, the system processes the request and either the event creation is successfully done or it displays an error message if there are any problems.

#### Functional Requirements

**REQ-1:** The "Add Event" choice should be clearly shown and be able to be reached easily in the event management section of the website.

**REQ-2:** Only the users with the right permissions should be authorized to use the "Add Event" feature.

**REQ-3:** The system should be able to check all the input fields to confirm the data is correct and to avoid any mistakes.

**REQ-4:** The event date and time fields should have a user-friendly calendar and a time picker interface that will make it easier for the user to select the date and time.

**REQ-5:** The system should be capable of the inclusion of images or multimedia content to create situations when people will be able to have the best presentations.

**REQ-6:** After the successful creation of the event, the system should send out a notification to the users who are subscribed to the event's updates and the event will be displayed in the club's event calendar.

## chapter 2

# REQUIREMENTS

### 2.4.7 admin/ Edit Event

#### Description

The selected participants are being given the chance to utilize the "Edit Event" function to amend the existing events available on the site. This action assists in the modification as well as the update of event details that help the club activities listed for the future to be organized and promoted effectively.

#### Stimulus/Response Sequences

**Stimulus:** The identified user, while being logged in, gets directed to the event designation and chooses "Modify Event" among others.

**Response:** The system provides a form to the user which is pre-filled with the existing event details and the user can apply changes accordingly to the title, description, event dates, time, etc. The system validates the request once the user submits the form, and if successful it updates the event otherwise, it displays the error message.

#### Functional Requirements

**RREQ-1:** The "Edit Event" option should be complex integrated and annotated within the event management area of the website.

**REQ-2:** Only authorized individuals with the proper role of editors should be permitted to utilize the "Edit Event" feature.

**REQ-3:** The system should be set up in a way that checks all the input fields (to confirm the right input) and also to avoid errors.

**REQ-4:** When setting the event date and time, a friendly calendar and time picker interface should be one of the features to aid easy selection.

**REQ-5:** The system should be built with the capability to show images or multimedia files to improve the performance of event presentations.

**REQ-6:** However, this system should notify users who have subscribed to receive updates about the changes, and the club event calendar should be updated to reflect the changes.

## chapter 2

# REQUIREMENTS

### 2.4.8 admin/ Delete Event

#### Description

Authorized administrators have the capability to utilize the "Cancel Event" function to revoke or cancel existing events listed on the website. This action facilitates the removal of events from the schedule, ensuring effective management of club activities.

#### Stimulus/Response Sequences

**Stimulus:** person responsible for supervising the event, while logged in to the system, goes to the event management section and selects "Cancel Event" from among the available options displayed.

**Response:** The cancellation request prompts the administrator to enter the corresponding process that confirms their intent to cancel the event. With the coordination being accomplished, the system will process the request, will remove the conference from its schedule, and change the event description to reflect the updated information.

#### Functional Requirements

**REQ-1:** the "Cancel Event" option should be prominently integrated within the event management area of the website, with clear annotations for easy identification.

**REQ-2:** Only authorized individuals with administrative privileges should be allowed to utilize the "Cancel Event" feature.

**REQ-3:** Errors in processing cancellation requests prompt the system to be built in a way that verifies and validates all input fields and then eliminates errors there.

**REQ-4:** When canceling an event, the system should prompt the administrator to confirm their action to prevent accidental cancellations.

**REQ-5:** Upon successful cancellation of an event, the system should notify subscribed users about the cancellation and update the club's event calendar to reflect the changes.

**REQ-6:** Additionally, the system should provide an option for administrators to add notes or reasons for the cancellation, which other administrators can view for reference.

## chapter 2

# REQUIREMENTS

### 2.4.9 Print receipt and send It to users admin

#### Description

Authorized administrators have the power to print invoices in addition to sending them to the users for the processing of payments. This response simplifies the invoice process and ensures prompt payment collection.

#### Stimulus/Response Sequences

**Stimulus:** The administrator logs into the system and navigates to the invoicing section. Then, he or she selects the invoice to be printed and sent by choosing the "Print & Send" option.

**Response:** The system creates the invoice, converts it into a printable form, and sends it to the designated recipient. The information is delivered with the assurance of successful transmission.

#### Functional Requirements

**REQ-1:** The "Print & Send" option should be conveniently situated in the invoicing area with a distinctive label that is easily comprehensible.

**REQ-2:** Only managers with the "Print Invoice & Send" capability should be allowed to use this function to maintain control over invoicing processes.

**REQ-3:** The system should be structured in a way that validates all the input fields to ensure that the invoicing information is correct and errors are eliminated.

**REQ-4:** Before sending an invoice, the system should ask the administrator to check the action to be sure it is not mistakenly sent.

**REQ-5:** Following a successful invoice transmission, the system should notify the recipient via an update of the invoice status.

**REQ-6:** Moreover, the system should enable managers to include memos or reasons for sending the invoice, which is accessible to other managers for reference.

## chapter 2

# REQUIREMENTS

### 2.4.10 View all students admin

#### Description

The "View all students" option on our website enables administrators to see a list of all registered students with their details.

#### Stimulus/Response Sequences

**Stimulus:** The administrator goes to the website and clicks on "View all students".

**Response:** After visiting the "View all students" page, the administrator sees a list of all the registered students with their details such as username, email, and role.

#### Functional Requirements

**REQ-1:** The "View all students" page should show a list of all the registered students with their details.

**REQ-2:** The system should enable administrators to look for particular students by username or email.

**REQ-3:** The system should offer pagination for the student list to handle the large number of registered students.

**REQ-4:** If there are many results the system should divide them into pages to improve user experience.

## chapter 2

# REQUIREMENTS

### 2.4.11 Search for User/ admin

#### Description

The search function on our website enables administrators to locate particular users by entering their usernames or email addresses.

#### Stimulus/Response Sequences

**Stimulus:** Admin goes to the website and chooses “Search for User”.

**Response:** The administrator is shown the "Search for User" page, which is a search form where they can enter the username or email address of the user they are searching for. Once the search query is submitted, the system will show the user's details if they are found.

#### Functional Requirements

**REQ-1:** The system should have a search form for the administrators to fill in the username or email address of the user they want to locate.

**REQ-2:** The system should enable administrators to search for users by username or email address.

**REQ-3:** If the user is located, the system should show his/her details.

## chapter 2

# REQUIREMENTS

### 2.4.12 Delete User admin

#### Description

The "Delete User" feature on our website allows administrators to remove specific users from the system.

#### Stimulus/Response Sequences

**Stimulus:** The administrator navigates to the website and selects “Delete User”

**Response:** Upon accessing the "Delete User" page, the administrator is presented with a form or a confirmation dialog to confirm the deletion of the selected user. After confirming, the system will remove the user from the system.

#### Functional Requirements

**REQ-1:** The system should provide a way for administrators to select the user they want to delete.

**REQ-2:** The system should require confirmation from the administrator before deleting the user.

**REQ-3:** Upon confirmation, the system should remove the user from the system.

## chapter 2

# REQUIREMENTS

### 2.4.13 Subscribing to an event user

#### Description

The option of subscribing allows the users to join a prearranged event by simply clicking on the “subscribe” button while having enough balance in their accounts.

#### Stimulus/Response Sequences

**Stimulus:** after navigating to the desired event click on the “subscribe” button.

**Response:** The system will check both the user's balance and availability to join the event. Once both are confirmed to meet the requirements the system will make a reservation and withdraw fees from the user's balance and if there isn't enough balance or available seats the system will present a message to the user.

#### Functional Requirements

**REQ-1:** The “Subscribe” button should be visible and user-friendly to the users on the event's page.

**REQ-2:** The subscription process should be user-friendly.

**REQ-3:** The system should present all the details about the event such as location, date, price, etc.

**REQ-4:** The payment process should be accurate, error-free, and secure.

**REQ-5:** An option for cancellation and refund should be available under the Organization Policy rules.

## chapter 2

# REQUIREMENTS

### 2.4.14 Show Subscribers

#### Description

The "Show Subscribers" option provides User/Admin with the ability to see a list of individuals who have subscribed to a specific event, along with details of their number. This feature enables the User/Admin to easily track event subscribers.

#### Stimulus/Response Sequences

**Stimulus:** after navigating to the desired event click on the "Show Subscribers" button.

**Response:** The system will generate a window that contains the names and emails of the users who had subscribed to this particular event with their number.

#### Functional Requirements

**REQ-1:** The "Show Subscribers" button should be visible and user-friendly to the users on the event's page.

**REQ-2:** The System should generate a page that contains the names, emails, and the number of users.

**REQ-3:** The page should present the required information in a user-friendly way

**REQ-4:** The page should contain a filter with sorting options.

**REQ-5:** The information presented on the page should meet with the Organization's Privacy Policy

## chapter 2

# REQUIREMENTS

### 2.4.15 Payment

#### Description

Payment is a page that provides the user with the necessary information and actions about their payments

#### Stimulus/Response Sequences

**Stimulus:** navigating to the user's payment page from the dashboard

**Response:** The system will present a page that contains information about the user's balance and withdrawal history plus some features like Add Balance. This page contains an option to navigate to history that shows the events that the user had subscribed in

#### Functional Requirements

**REQ-1:** The “Payment” page should be user-friendly to the users with the ability to load quickly to enhance user experience.

**REQ-2:** The page should have the ability to handle errors and provide clear and understandable error messages.

**REQ-3:** The page should contain an option for adding balance with support for multiple payment methods.

**REQ-4:** The page should contain an option to display payment history.

## chapter 2

# REQUIREMENTS

### 2.4.16 Add Balance

#### Description

The system contains a payment feature that enables the user to feed their accounts so they can benefit and join the events organized by the club/organization

#### Stimulus/Response Sequences

**Stimulus:** On the user's Payment page. The user clicks on "Add Balance"

**Response:** The system will show a window where the user has different payment options, including Visa, Mastercard, PalPay, and Reflect. The user chooses the amount that he/she intends to add then by clicking confirm the system with the API should do the required actions.

#### Functional Requirements

**REQ-1:** The Payment method information fields showed be mandatory.

**REQ-2:** The system Should contain a Payment Gateway and Multi-Currency Support

**REQ-3:** The system should contain high-level encryption and a security standard like PCI-DSS to ensure customer safety and privacy

**REQ-4:** Before sending an invoice, the system should ask the administrator to check the action to be sure it is not mistakenly sent.

**REQ-5:** The system should provide APIs for integrating with other systems like accounting or CRM2.

**REQ-6:** The system should notify the user about the state of their transactions.

## chapter 2

# REQUIREMENTS

### 2.4.17 User/ Participate in Event

#### Description

The "Participate in Event" feature on our website allows users to sign up for events selecting the event they wish to participate in.

#### Stimulus/Response Sequences

**Stimulus:** User navigates to the website and selects “Participate in Event”.

**Response:** The user is presented with the "Participate in Event" page, which includes a form where they can choose the event they want to join. Upon submitting the form, the system will confirm their participation and display a confirmation message.

#### Functional Requirements

**REQ-1:** The system should have a participation form for users to select the event they want to participate in.

**REQ-2:** The system should validate the availability of the selected event.

**REQ-3:** Upon successful registration, the system should confirm the user's participation and display a confirmation message.

**REQ-4:** If there are any errors, the system should display appropriate error messages.

## chapter 2

# REQUIREMENTS

### 2.4.18 User/ Cancel Event for User

#### Description

In the present work, one of our main features, “Cancel Event,” To initiate the cancellation process, users are required to provide their user identification and select the event they wish to cancel. Upon successful cancellation, users receive a confirmation message indicating the successful cancellation of their participation

#### Stimulus-Response Patterns

**Stimulus:** User ends up on the app and clicks “Cancel Event”.

**Response:** The user is shown the “Cancel Event” page, where at the bottom, there will be a form that allows the user to select the event they wish to cancel. After filling and submitting the form, the system will return a success message informing the user that the cancellation was successful.

#### Functional Requirements

**REQ-1:** There should be a cancellation form that the user is supposed to input their credentails and select the event they wish to cancel.

**REQ-2:** The system should be able to verify the uniqueness of the user ID, and also define whether the user participating in the selected event.

**REQ-3:** In case of successful cancellation, the system should give a feedback that the event has been cancelled and what follows is an acknowledgement message.

**REQ-4:** Along the same lines, when citing sources, if there are any mistakes (e. g. If, for example, such input corresponds to an empty string, designation with an invalid user ID (meaning this user is not registered for the event), the system should output the following error messages.

# REQUIREMENTS

## 2.5 NON-FUNCTIONAL REQUIREMENTS

### 2.5.4 Performance Requirements

- 1- Request time: The system should be able to process any request within 2 seconds.
- 2-Number of users: The system should handle the concurrent access by at least 100 users without significantly degrading performance.
- 3- Shutdown time: The shutdown time for maintenance should be scheduled during peak-off time.
- 4- Availability: The system should be available 99.9 of the time (excluding shutdown time)

### 2.5.2 Safety Requirements

No safety requirements have been identified.

### 2.5.3 Security Requirements

**SEC-1** The system shall allow users to access the Login, Forgot Password, and Register page.

**SEC-2** The system shall require a login and password from the user and the admin to gain access to their specified interfaces

**SEC-3** The system shall comply with industry-standard security, including data encryption and address potential security risks.

### 2.5.4 Reliability

**1- Data Integrity:** Data integrity should be contained to prevent Data loss and corruption

**2- High Availability:** reducing downtime and maintenance and ensuring that the system can be accessed by the users at any time.

### **2.5.5 Usability**

The system will be used by individuals of varying skill levels and technical competence. The system shall be intuitive to use and easy for clients but for admins, it may require a few hours of training depending on their technical skills.

### **2.5.4 Reliability**

The code and the design shall be well documented so that a junior project team with the same academic experience can ramp up on the project within 3 weeks.

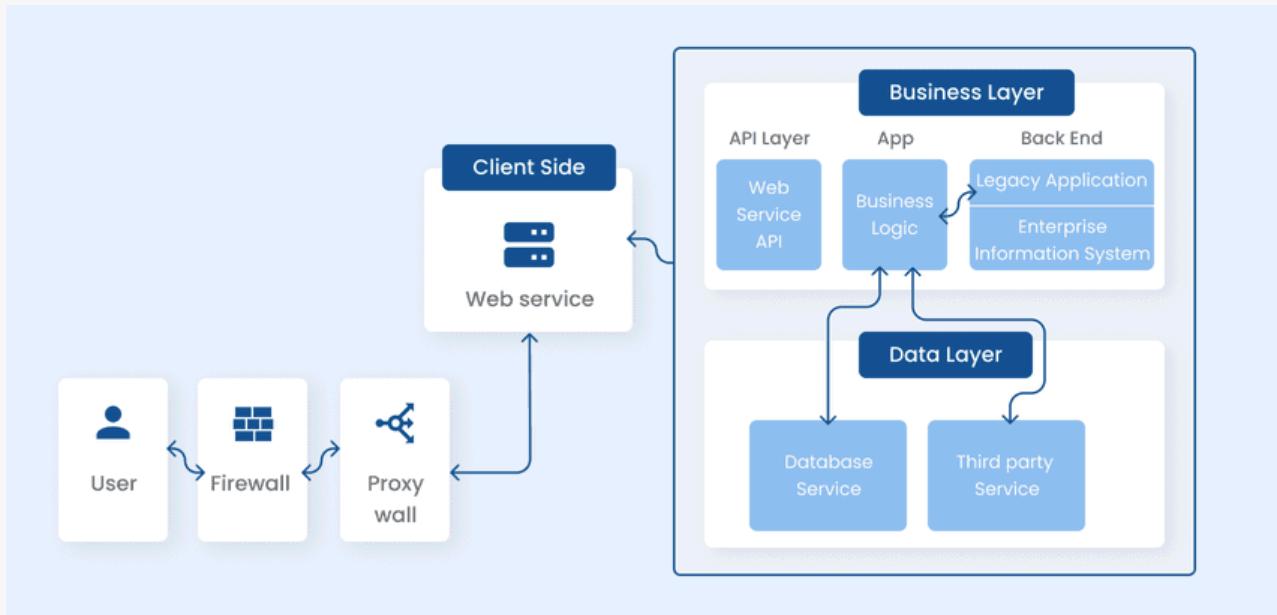
# **CHAPTER THREE**

## **SYSTEM ARCHITECTURE**

# chapter4

## SYSTEM MODELS

### SYSTEM ARCHITECTURE



## **CHAPTER FOUR**

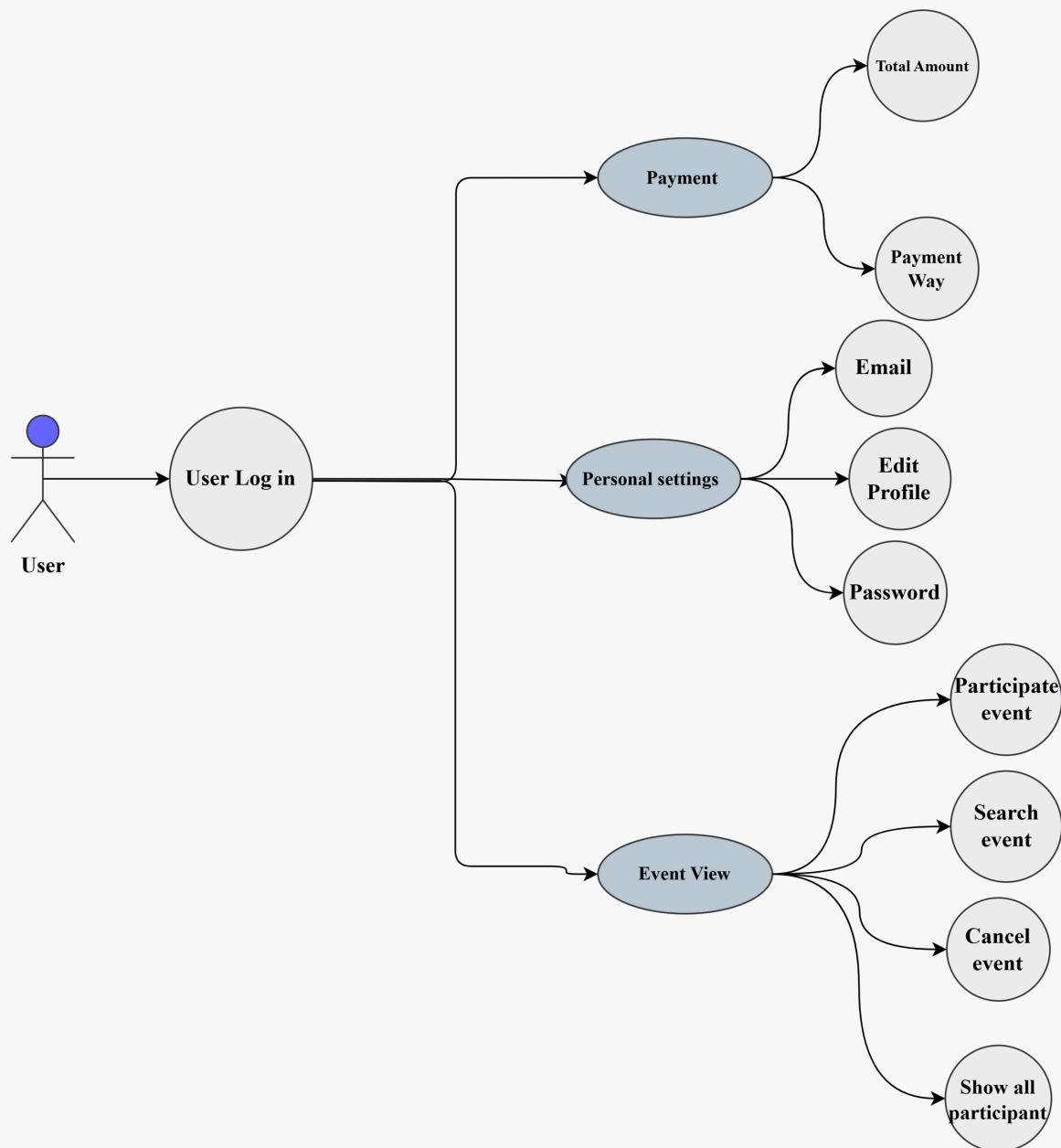
## **SYSTEM MODELS**

## 4. SYSTEM MODELS

### 4.1 USE CASE MODEL

It provides a high-level description of how user and admin interact with the system.

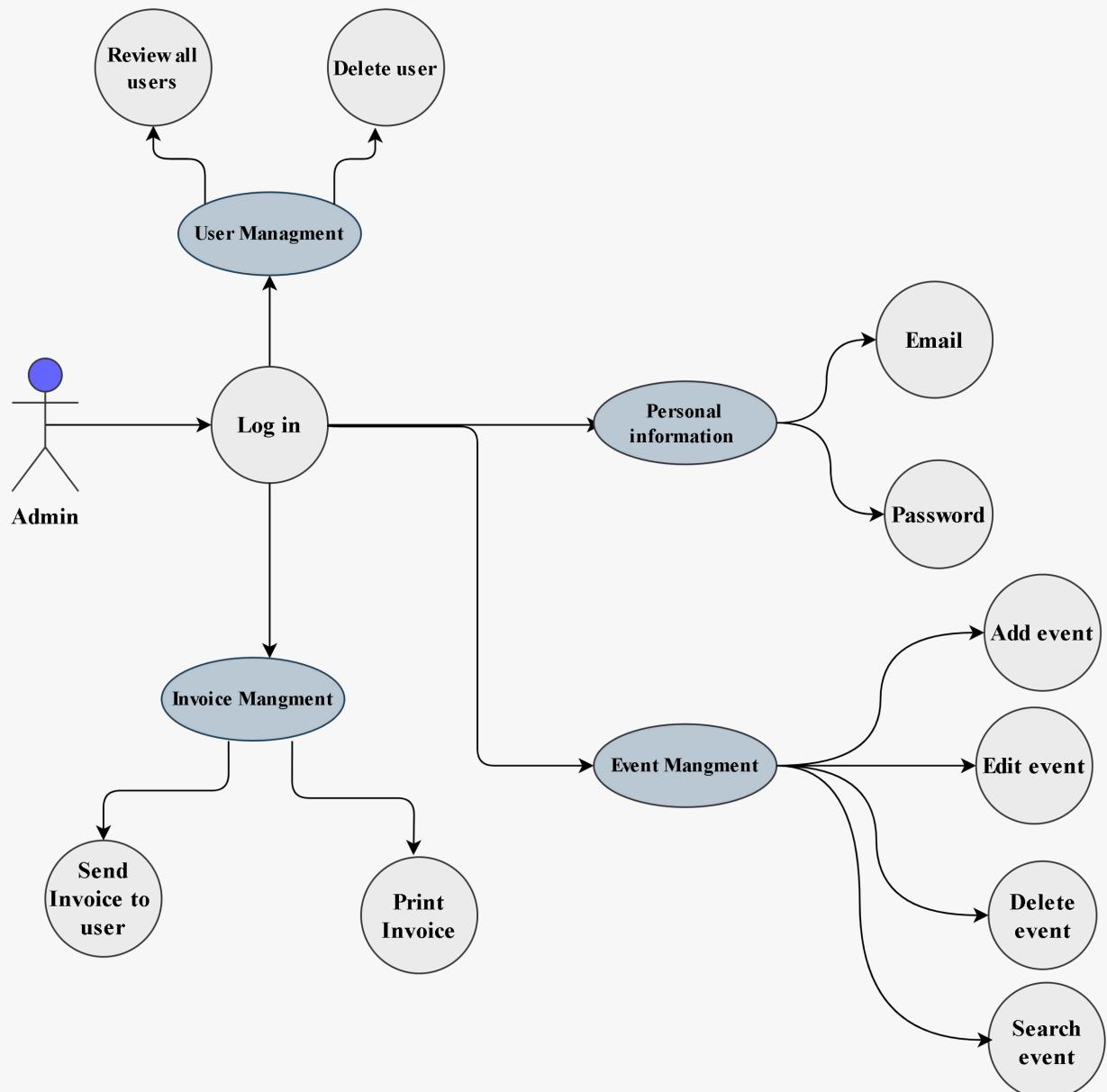
#### 4.1.1 USER USE CASE MODEL



# chapter4

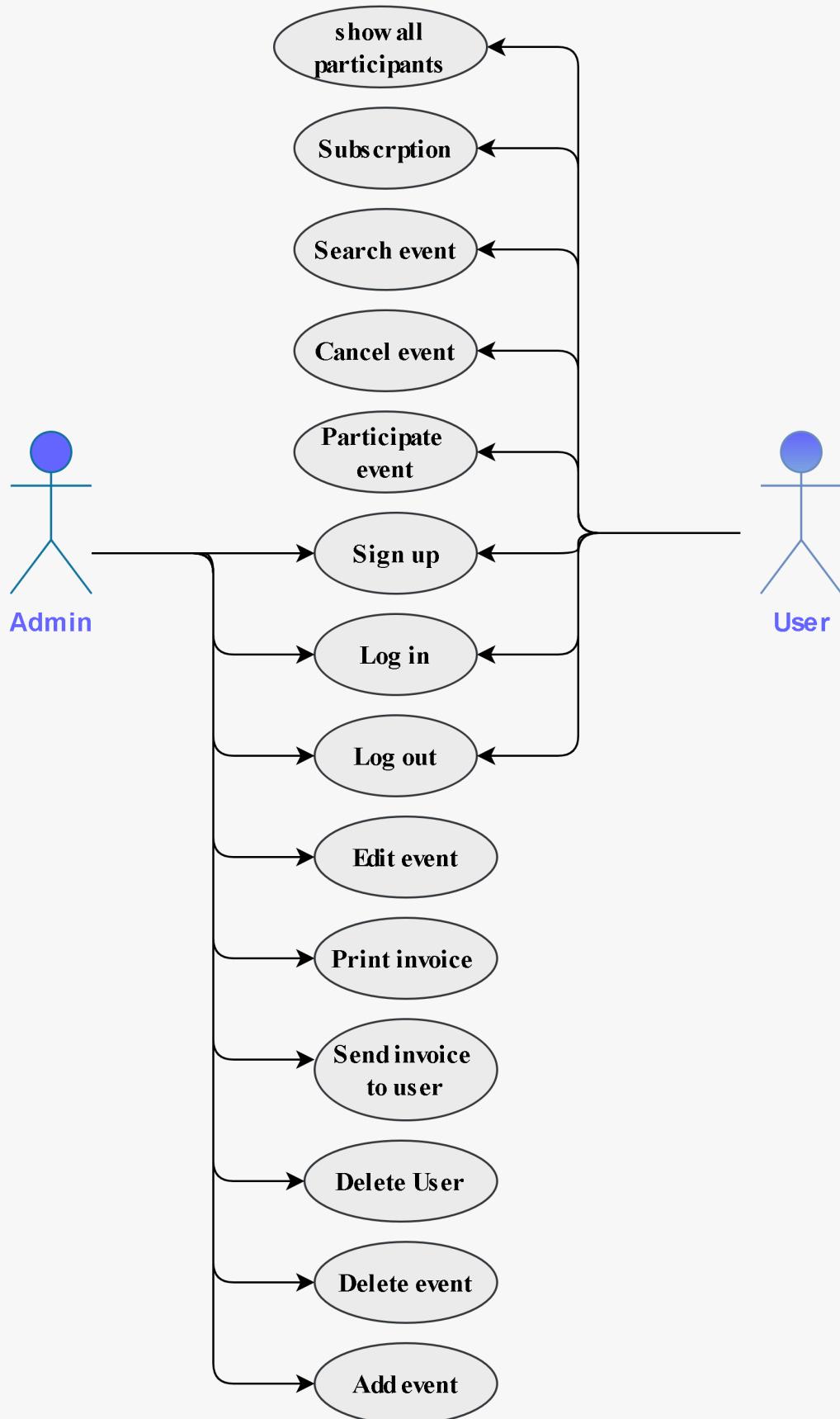
## SYSTEM MODELS

### 4.1.2 ADMIN USE CASE MODEL



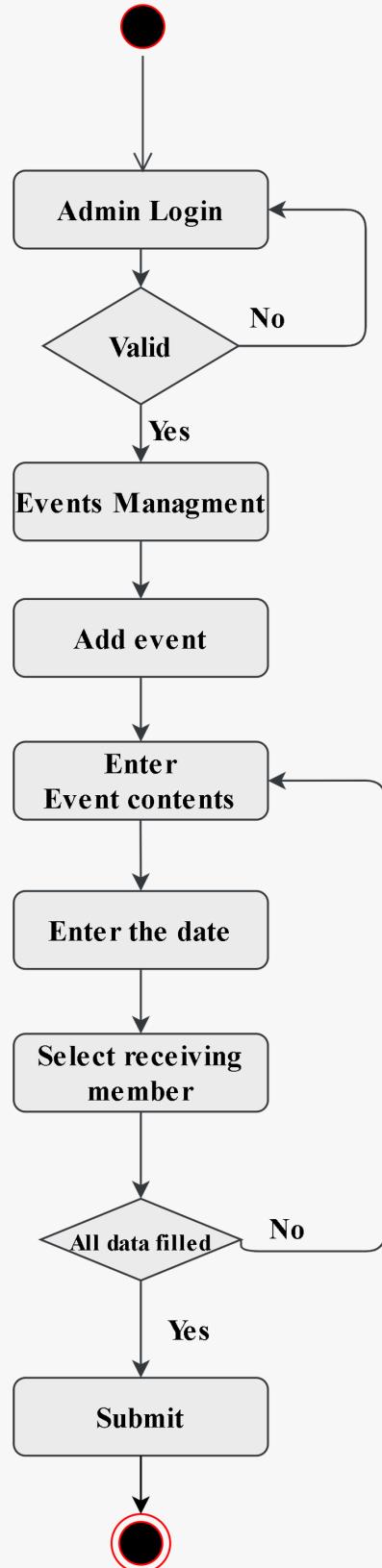
## chapter4 SYSTEM MODELS

### 4.1.3 ADMIN/USER USE CASE MODEL



## 4.2 ADMIN ACTIVITY DIAGRAM

- Add Event

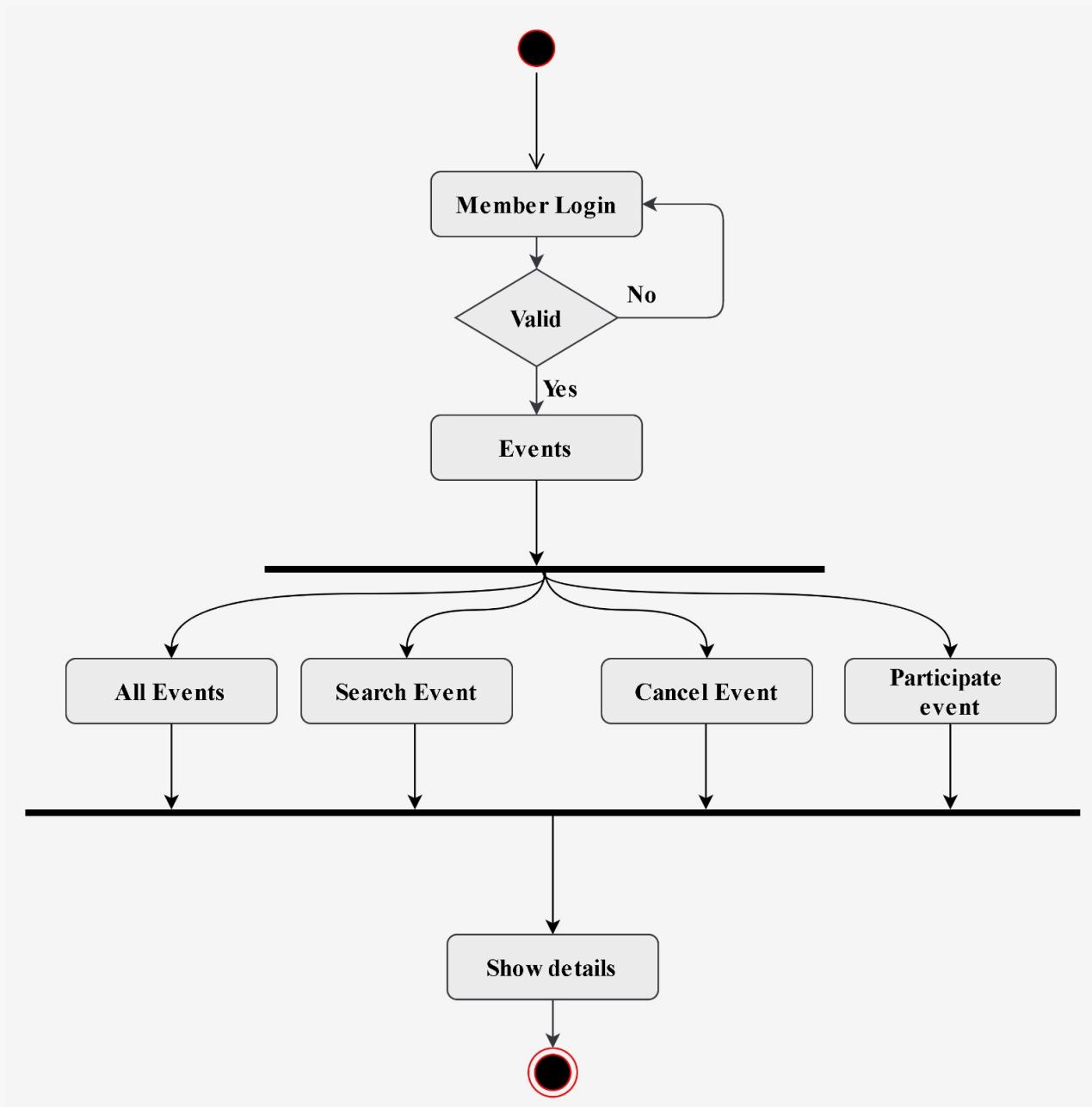


# chapter4

## SYSTEM MODELS

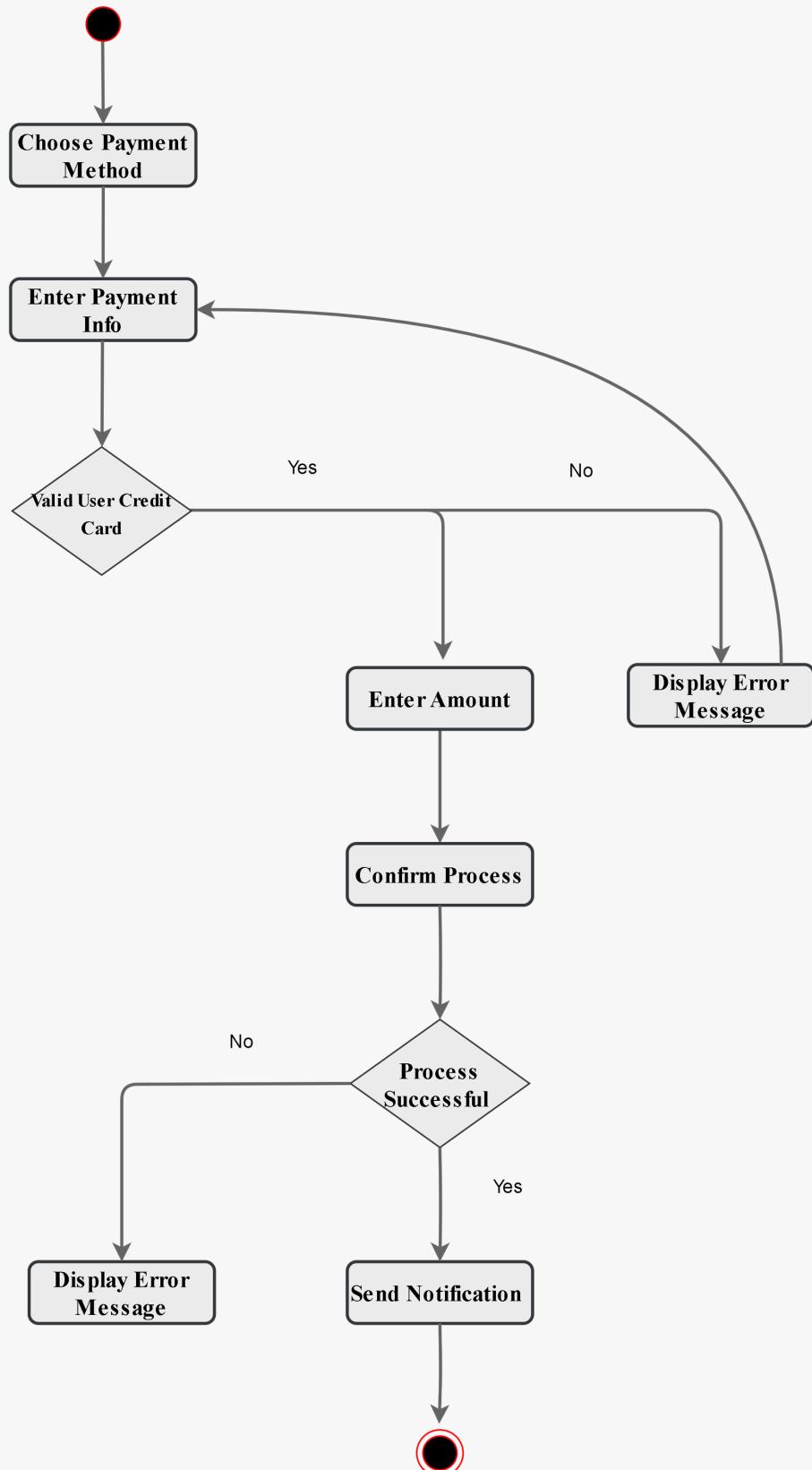
### 4.2 USER ACTIVITY DIAGRAM

- Show Event



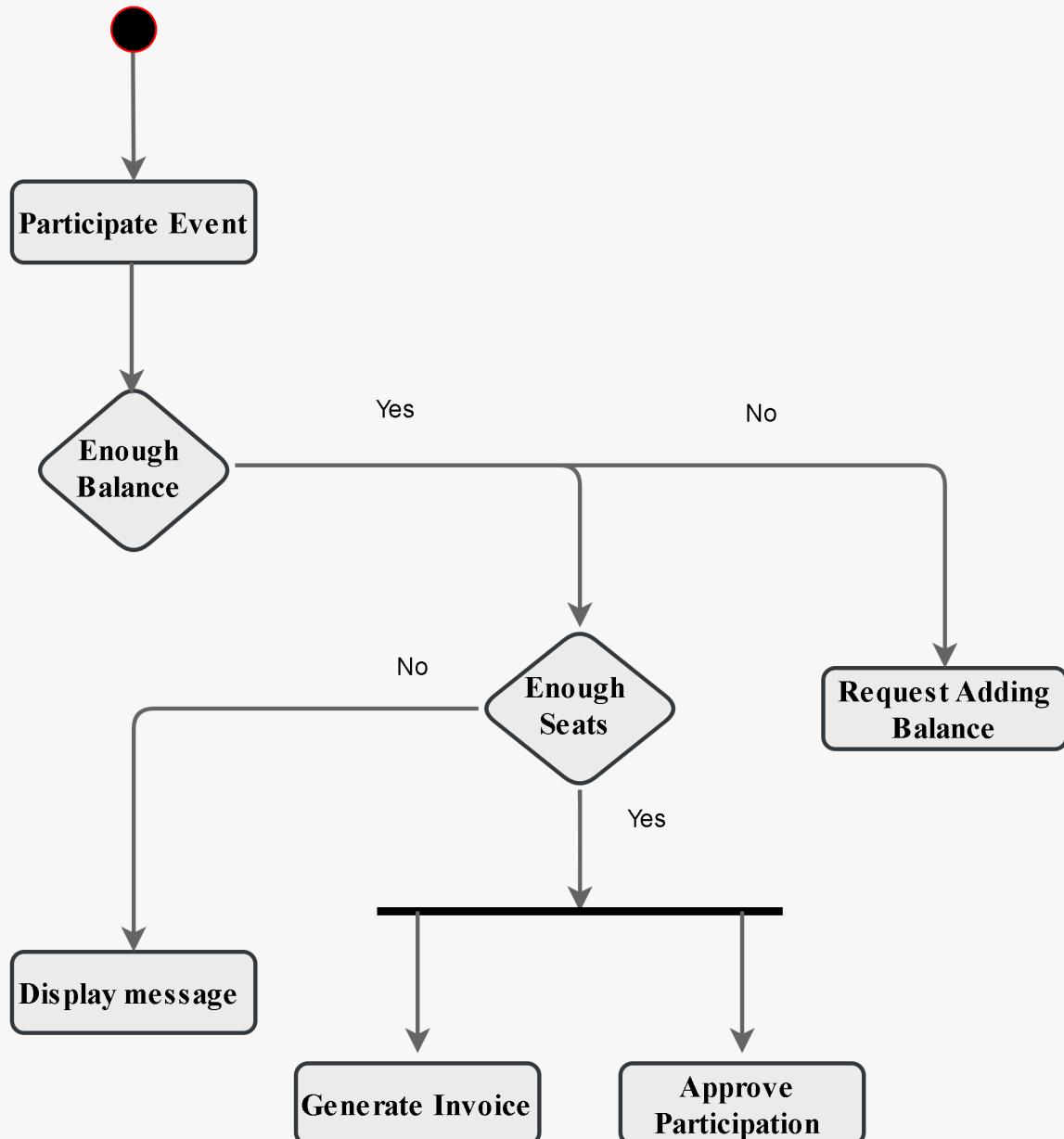
chapter4  
SYSTEM MODELS

## 4.2 USER ACTIVITY DIAGRAM



chapter4  
SYSTEM MODELS

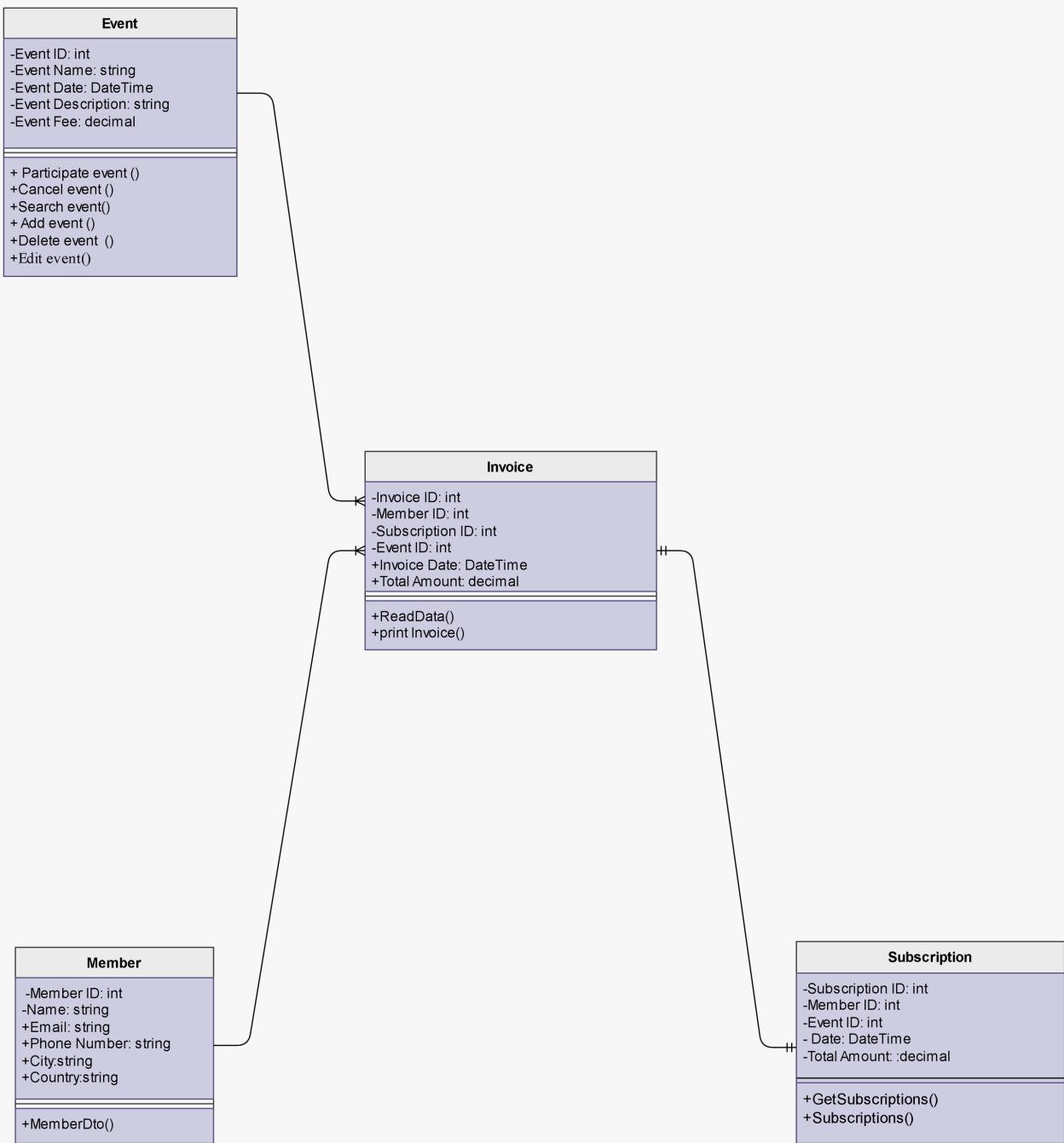
## 4.2 USER ACTIVITY DIAGRAM



# chapter4

## SYSTEM MODELS

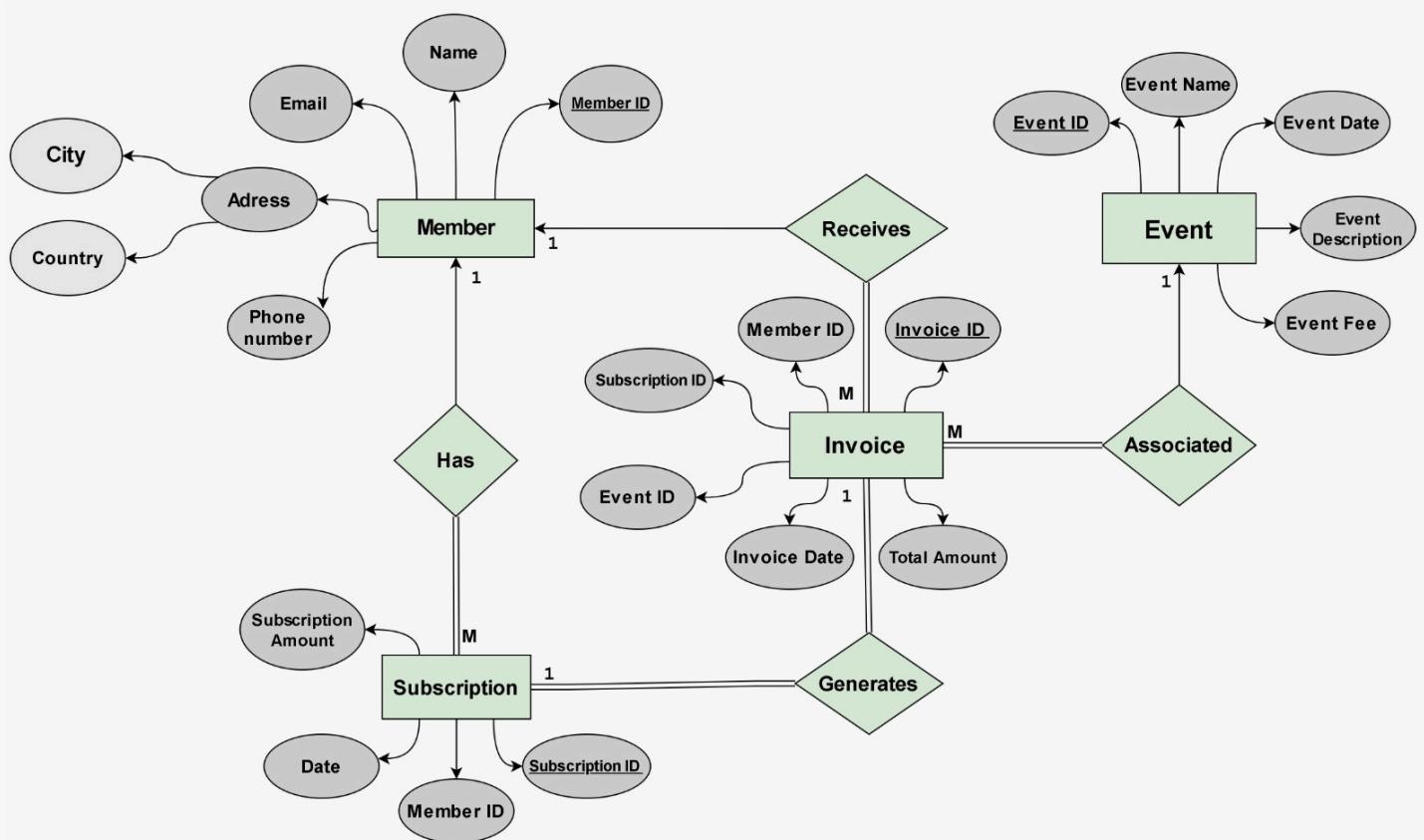
### 4.3 CLASS DIAGRAM



# chapter4

## SYSTEM MODELS

### 4.4 ER-DIAGRAM

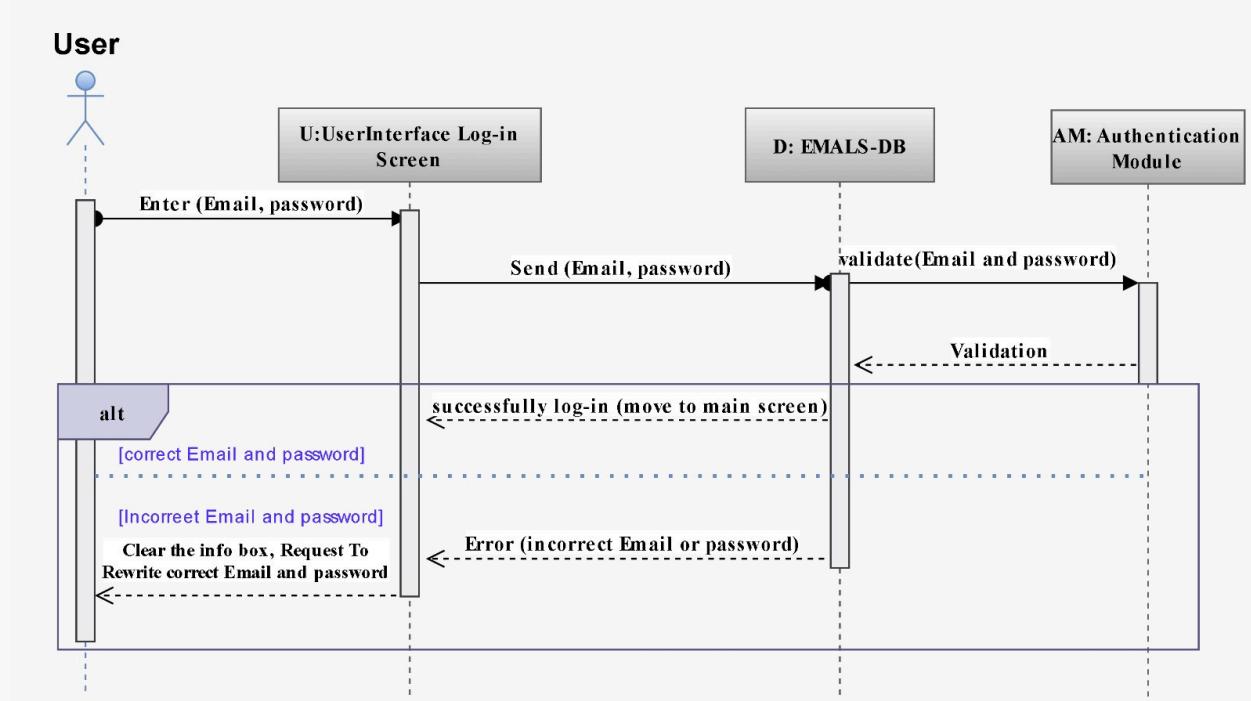


# chapter4

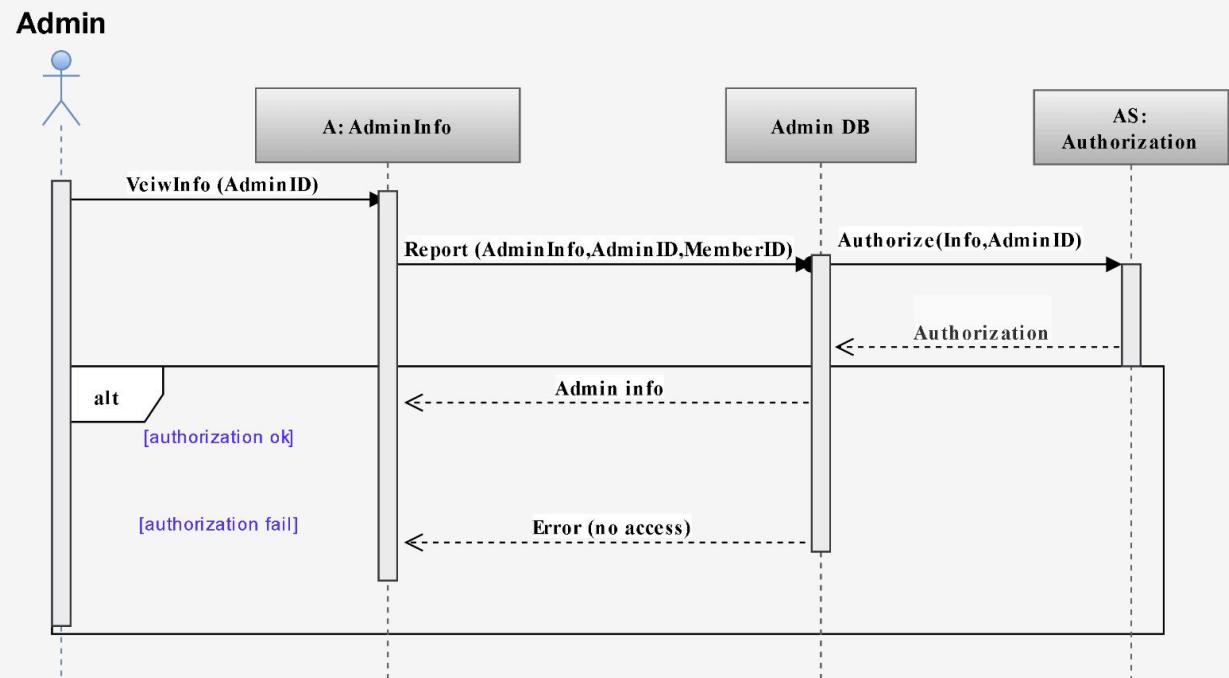
## SYSTEM MODELS

### 4.5 SEQUENCE DIAGRAM

#### 4.5.1 User Log in Sequence Diagram



#### 4.5.2 Admin Log in Sequence Diagram

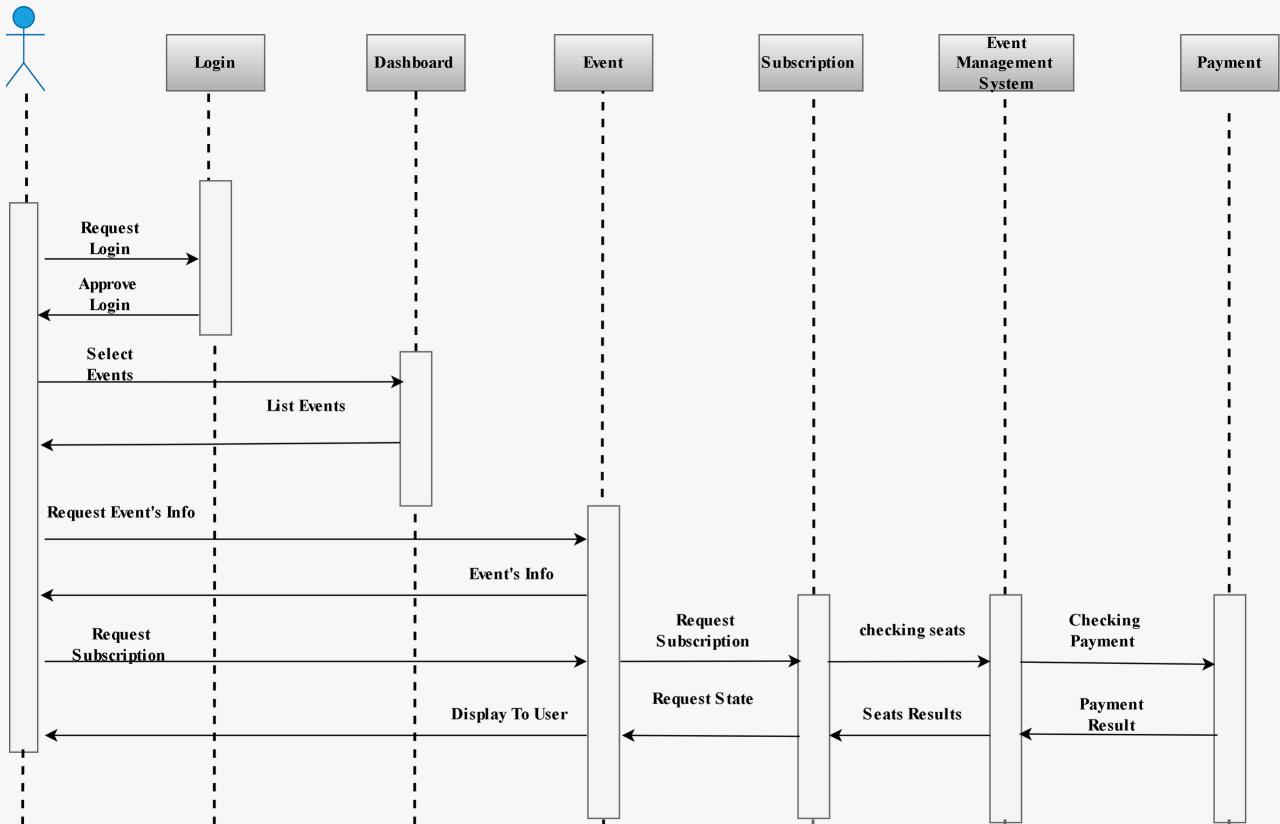


# chapter4

## SYSTEM MODELS

### 4.5 SEQUENCE DIAGRAM

#### 4.5.3 User subscription process



# **CHAPTER FIVE**

## **SYSTEM TESTING**

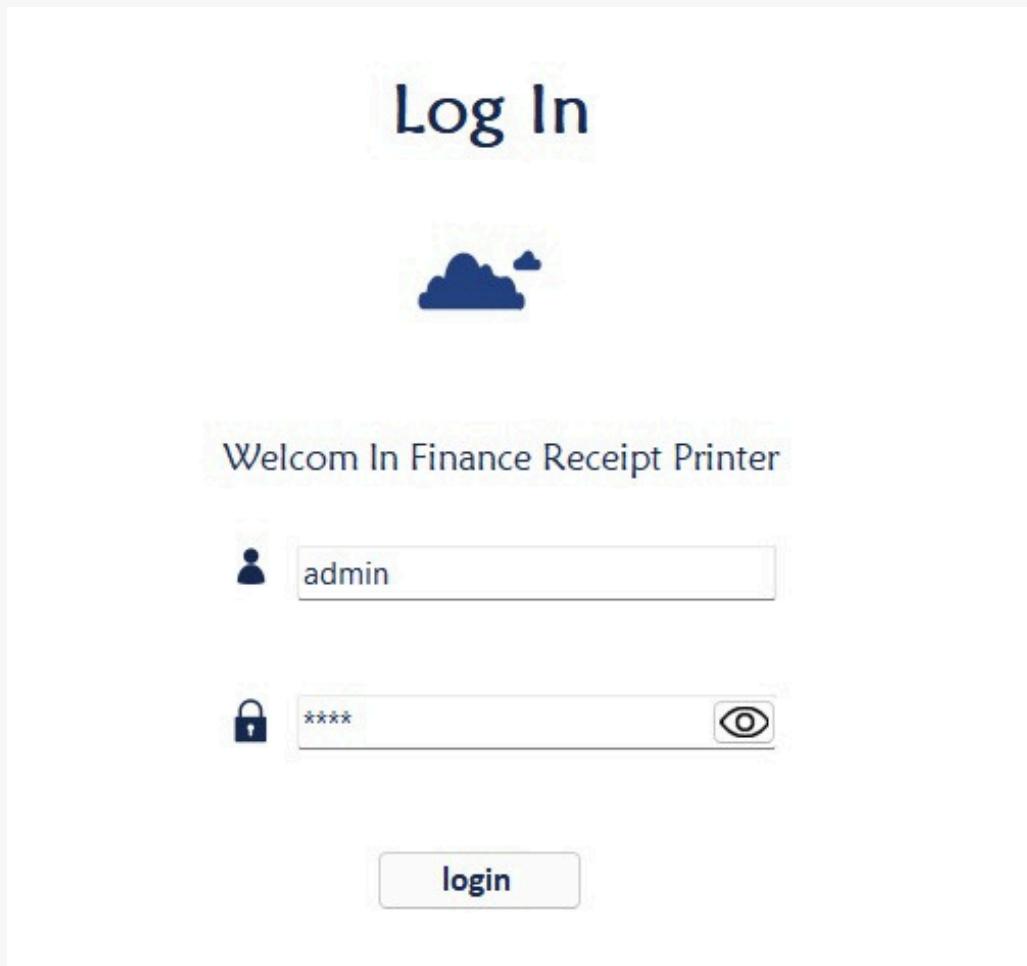
**GITHUB LINK:**

**<https://github.com/razan0r/software-project>**

## chapter 5

### SYSTEM tESTING

#### 5.1 TESTING ADMIN VIEW:



# chapter 5

## SYSTEM tESTING

### 5.1 TESTING ADMIN VIEW:

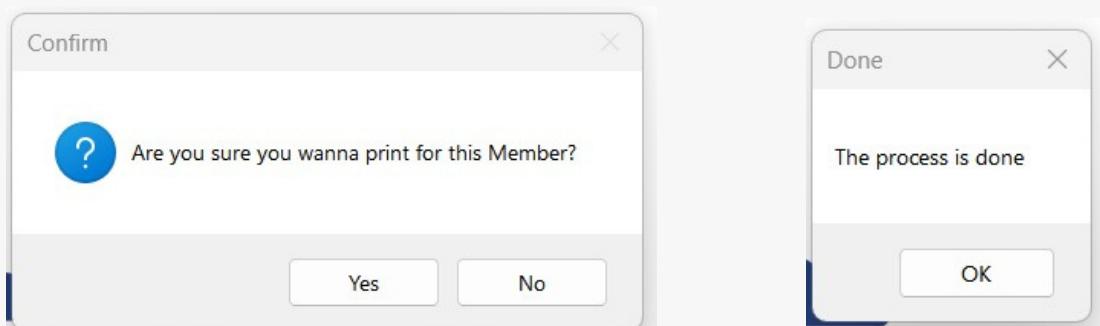
**payment receipt**

Please upload the excel file:

Please choose the city:

Please enter the Member ID here:

Id: 202110754, Name: Razan  
Id: 202110714 1074, Name:  
Huda  
Id: 202110734, Name: Shahd  
Id: 202110490, Name: Jad



**سند قبض**

Nablus-Palestine الصندوق: 20/5/2023 التاريخ:

paypal طريقة الدفع: Razan الاسم:

10 المبلغ:

2024 لسنة:

00000007

# chapter 5

## SYSTEM tESTING

### TESTING:

## 5.2 TESTING USER VIEW:

### 5.2.1 main page

The screenshot shows a dark blue header with white text and buttons. From left to right, the buttons are: تسجيل الدخول (Login), إنشاء حساب (Create Account), من نحن (About Us), الفعاليات (Activities), الرئيسية (Home), طلاب من أجل المستقبل (Students for the Future). Below the header, there is a large white area containing Arabic text and a group of stylized human figures.

شبكة طلابية عابرة تسعى إلى بناء مساحات  
جامعة وبناءة وبيئات تفاعلية وتبادلية للطلاب  
من مختلف التخصصات.

الأهداف:

1. خلق بيئة تفاعلية تشاركية استثمارية للطلاب .
2. تعزيز تبادل المعرفة والمهارات والخبرة.
3. التدريب والتطوير.

At the bottom of the page, there are social media icons for Instagram, Facebook, and TikTok.

# chapter 5

## SYSTEM tESTING

### TESTING:

#### 5.2.1 main page

The screenshot shows a dark blue header bar with several navigation links in Arabic: "تسجيل الدخول" (Login), "إنشاء حساب" (Create Account), "من نحن" (About Us), "الفعاليات" (Events), "الرئيسية" (Home), and "طلاب من أجل المستقبل" (Students for the Future). Below the header is a light gray section containing a "Create Account" form. The form fields are labeled in Arabic: "الاسم" (Name) with an input field, "البريد الإلكتروني" (Email) with an input field, and "إنشاء كلمة مرور" (Create Password) with an input field. At the bottom of the form is a large, rounded rectangular button labeled "إنشاء حساب" (Create Account) in white text.



# chapter 5

## SYSTEM tESTING

### TESTING:



### تسجيل الدخول

اسم المستخدم أو البريد الإلكتروني

كلمة مرور

سجل

هل نسيت كلمة المرور؟



# chapter 5

## SYSTEM tESTING

### TESTING:



استرجاع كلمة المرور

البريد الإلكتروني

إرسال رابط الاسترجاع



# chapter 5

## SYSTEM tESTING

### TESTING:

طلاب من أجل  
ال المستقبل

الرئيسية الفعاليات الاشتراك من نحن

Search icon X

اسم الفعالية: صالون نقاشي  
المنطقة: ZOOM

عدد المسجلين: 30

إلغاء الانضمام اكتمل العدد

اسم الفعالية: ورشة معرفية  
المنطقة: رام الله

عدد المسجلين: 7

\$ 15

اسم الفعالية: بوك كاست  
المنطقة: نابلس

عدد المسجلين: 15

انضمام

Instagram icon Facebook icon TikTok icon

# chapter 5

## SYSTEM tESTING

### TESTING:

طلاب من أجل  
المستقبل

الرئيسية الفعاليات الاشتراك من نحن

صالون نقاشي

وصف الفعالية: نقاش حول موضوع معين

حالة الفعالية: ممتنعة

عدد المسجلين: 30

قسم التعليقات:

سفا الحادة: shahd

فعالية اكتر من رائعة: Razan

# chapter 5

## SYSTEM tESTING

### TESTING:

The screenshot shows a mobile application interface for a sports club. At the top, there is a dark blue header bar with a menu icon (three horizontal lines) on the left. To the right of the menu are four items: 'الاشتراك' (Subscription), 'الفعاليات' (Activities), 'الرئيسية' (Home), and 'طلاب من أجل المستقبل' (Students for the Future). Below the header, a message in Arabic reads: 'شكراً للتسجيل في عضوية النادي! إذا أردت تثبيت عضويتك والمشاركة في الفعاليات المختلفة. رجاءً قم بدفع الاشتراك السنوي.' (Thank you for registering in the club! If you want to renew your membership and participate in various activities. Please pay the annual subscription.). To the right of this message is a button labeled 'دفع الاشتراك' (Pay subscription). Below the message, there are three payment method buttons: 'JAWAL PAY', 'APPLE PAY', and 'PAYPAL'. To the right of these buttons is a link labeled 'تغذية الحساب' (Feed the account). Further down, there are two input fields: one for 'الاسم' (Name) and one for 'رقم البطاقة' (Card number). At the bottom, there is a button labeled 'PAY BY CARD' and a row of social media icons for Instagram, Facebook, and TikTok.

الاشتراك الفعاليات الرئيسية طلاب من أجل المستقبل

شكراً للتسجيل في عضوية النادي! إذا أردت تثبيت عضويتك والمشاركة في الفعاليات المختلفة. رجاءً قم بدفع الاشتراك السنوي.

دفع الاشتراك

JAWAL PAY APPLE PAY PAYPAL تغذية الحساب

الاسم رقم البطاقة

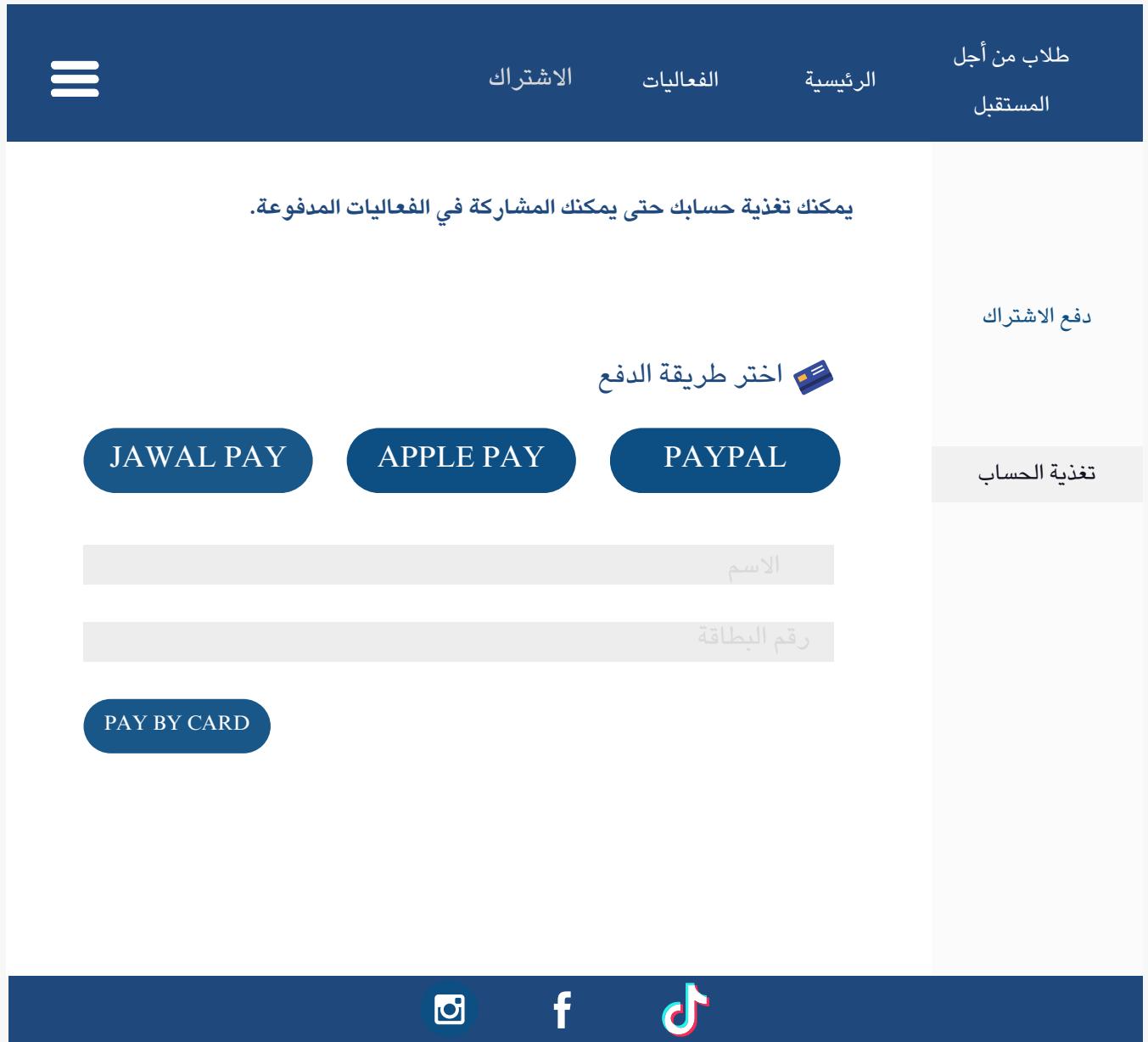
PAY BY CARD

Instagram Facebook TikTok

# chapter 5

## SYSTEM tESTING

### TESTING:

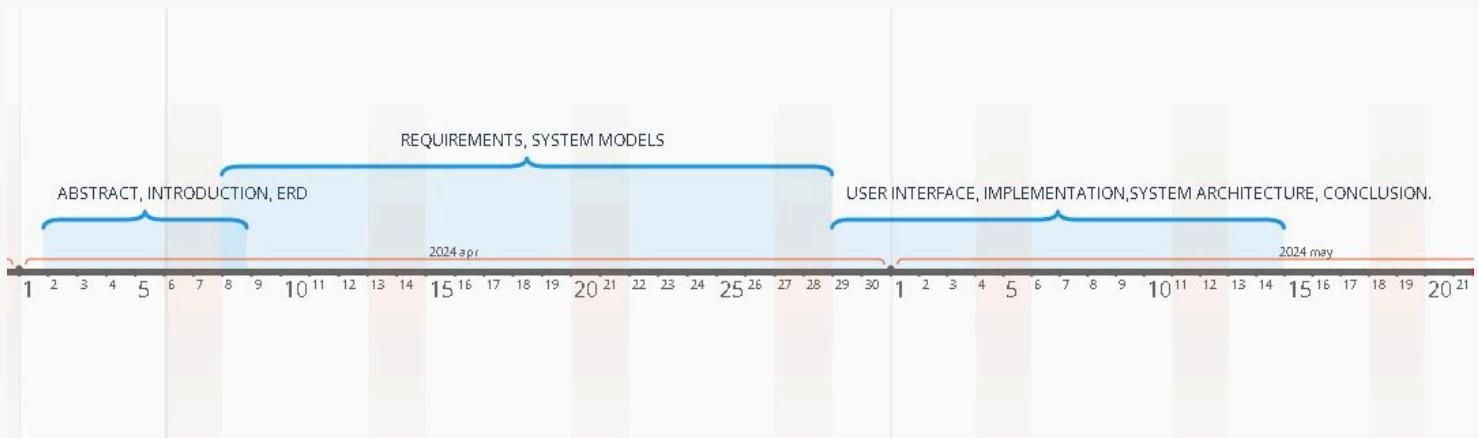


# chapter4

## SYSTEM MODELS

### PROJECT TIMELINE

<b>1 Week</b>	<b>ABSTRACT, INTRODUCTION, ERD.</b>
<b>2-4 Week</b>	<b>REQUIREMENTS, SYSTEM MODELS.</b>
<b>5 -7 Week</b>	<b>USER INTERFACE, IMPLEMENTATION, SYSTEM ARCHITECTURE, CONCLUSION.</b>



## **CONCLUSION:**

In conclusion, our website enhance organizational operations and user satisfaction by providing user-friendly features for account creation, event registration, and feedback reporting, and equipping administrators with tools for event, membership, and financial management. A well-designed interface ensures that all activities are conducted smoothly, fostering a positive and engaging environment for all participants.

Working on this project has been extremely valuable. At the outset, we struggled to come up with a project idea and to gather the user and system requirements. Our project is more complex than it appears, requiring substantial effort to address all the challenges.

We all consider ourselves to be beginners in the field of web development, lacking extensive experience with web technologies. Therefore, we plan to work collaboratively and learn from each other.