**SOFTWARE DESIGN DOCUMENT**

**FOR**

**TIME MANAGEMENT SYSTEM**  
  
<Nhóm 8>

# 1. Introduction

## 1.1 Phạm vi

Trong báo cáo này, chúng tôi đang lên kế hoạch cung cấp thông tin cho các lập trình viên để viết mã. Do đó, tài liệu bao gồm các yêu cầu của các lập trình viên cho việc phát triển mã.

## 1.2 Mục đích

Tài liệu này mô tả thiết kế khái niệm của Dự án Calendy, theo hướng dẫn trong tài liệu mô tả của IEEE 1016-1998 Recommended Practice for Software Design (SDD). SDD cho thấy cách hệ thống phần mềm sẽ được cấu trúc để đáp ứng các yêu cầu được xác định trong mô tả yêu cầu phần mềm. Nó là một bản dịch của các yêu cầu thành một mô tả về cấu trúc phần mềm, các thành phần phần mềm, giao diện và dữ liệu cần thiết cho giai đoạn triển khai. Theo bản chất, SDD trở thành một bản thiết kế chi tiết cho hoạt động triển khai. Trong một SDD hoàn chỉnh, mỗi yêu cầu phải có thể được theo dõi đến một hoặc nhiều thực thể thiết kế.

## 1.3 Tổng quan

Mục đích của tài liệu này là giúp người đọc hình dung được giải pháp cho dự án được trình bày. Tài liệu này xác nhận cách thiết kế đáp ứng các yêu cầu được quy định trong tài liệu SRS thông qua các quan điểm thiết kế. Các quan điểm thiết kế sẽ bao gồm tất cả các yếu tố thiết kế đã được trình bày trước đó. Bằng cách sử dụng thông tin từ IEEE 1016-1998, tài liệu này sẽ cung cấp một cách tiếp cận trực tiếp đến việc phát triển dự án này, từ đó giảm thiểu sự lan rộng tính năng không kiểm soát và xác định một cách cụ thể chất lượng của thiết kế.

## 1.4 Tài liệu tham khảo

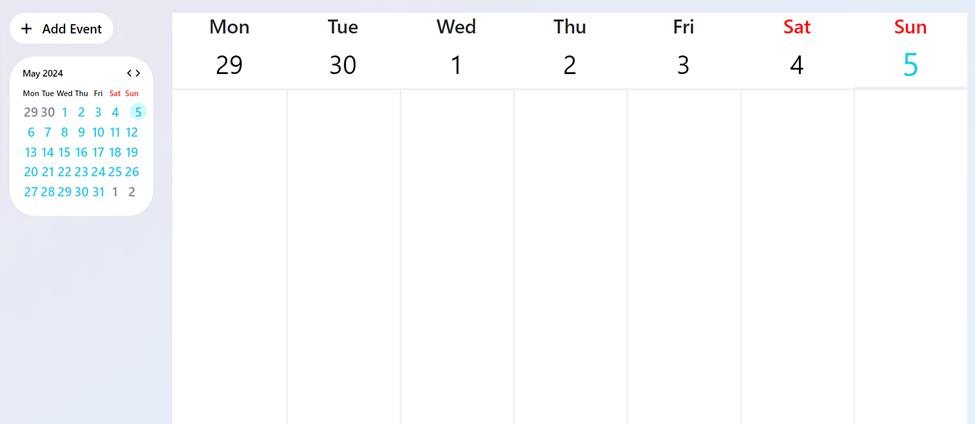
IEEE, *IEEE Std 1016-1998 Recommended Practice for Software Design Descriptions*, 1998-  
 09-23, The Institute of Electrical and Electronics Engineers, Inc., (IEEE )  
 IEEE, *IEEE 1016* Software Design Document (SDD) Template for CENG491

## 1.5 Định nghĩa và viết tắt

# 2. Đặc tả yêu cầu

## 2.1. Yêu cầu Chức năng:

### 2.1.1. Hiển thị lịch

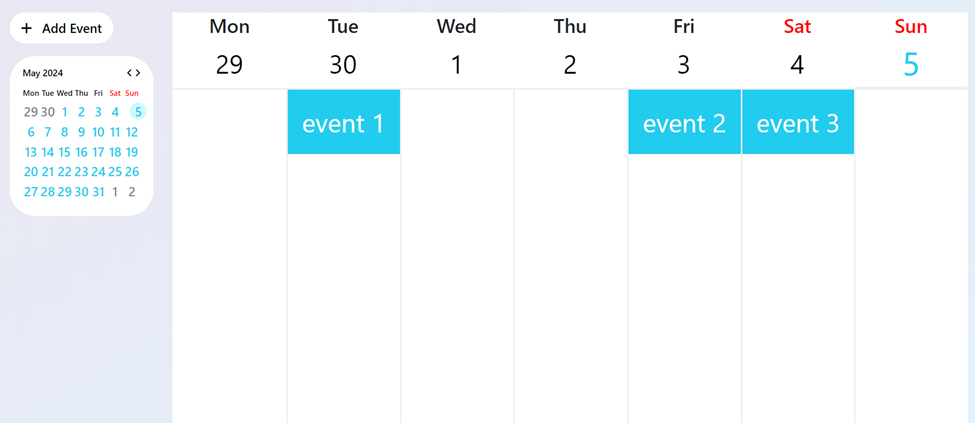


Hệ thống sẽ lấy ngày tháng năm hiện tại làm mốc, sử dụng thuật toán để lấy ra những ngày còn lại trong tuần, tháng (tháng hiện tại, tháng trước và sau) để tạo ra một bảng lịch hoàn chỉnh hiển thị ra màn hình cho người dùng.

### 2.1.2. Thay đổi lịch

Người dùng có thể sử dụng các nút điều hướng để chuyển đến lịch của tháng trước, sau hoặc sử dụng tính năng nhập vào ngày tháng năm theo định dạng ‘dd/mm/yyyy’ chuyển đến lịch theo mong muốn

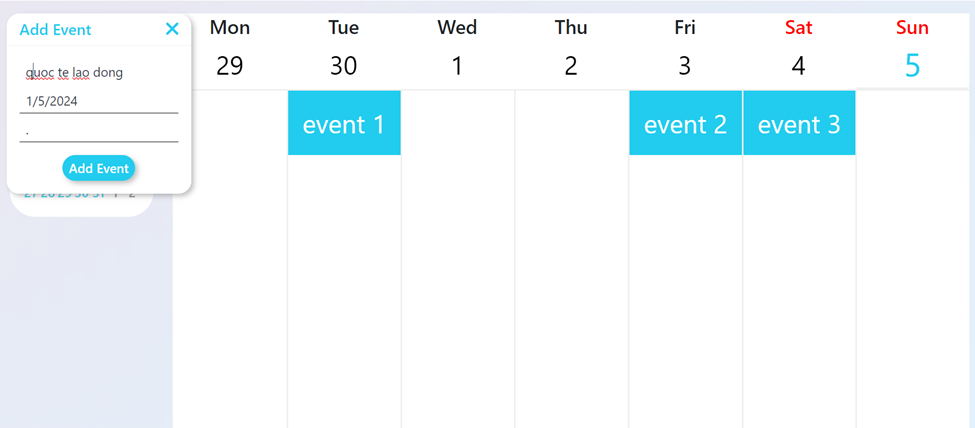
### 2.1.3. Hiển thị sự kiện

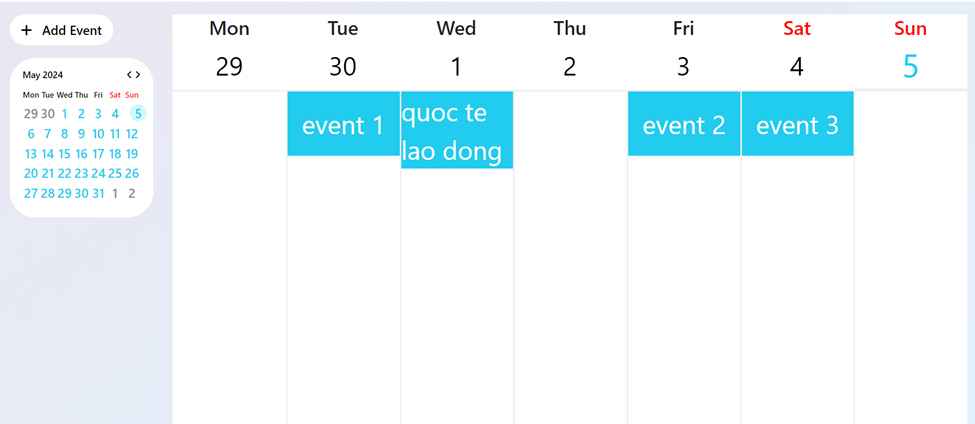


Các sự kiện sẽ được hiển thị trong bảng lịch tuần gồm có 7 cột tương ứng với 7 ngày trong tuần, mỗi cột trong bảng tuần sẽ có một id riêng tương ứng với ngày tháng năm của cột đó, hệ thống sẽ sử dụng api có sẵn để gửi yêu cầu lên server lấy về danh sách sự kiện trong database, sau đó hệ thống sẽ so sánh thuộc tính date của các sự kiện với id của các cột để xắp xếp các sự kiện vào vị trí tương ứng

Bên phía server, hệ thống sẽ gọi đến phương thức getEvents của event model để lấy về danh sách các sự kiện từ database và trả về cho client

### 2.1.4. Thêm sự kiện

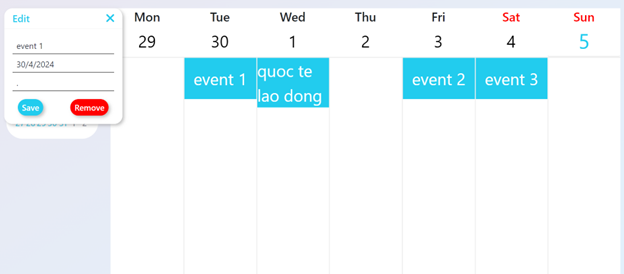


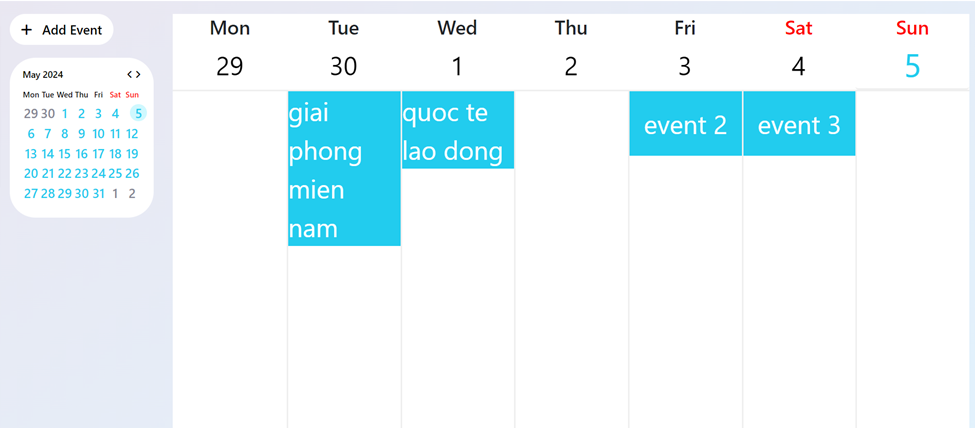
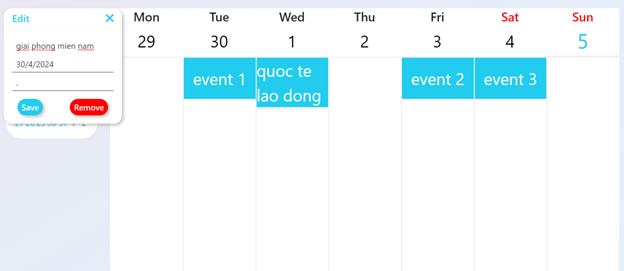


Nếu muốn thêm mới sự kiện, người dùng có thể sử dụng nút add event, sau khi click một form nhập thông tin sẽ được hiện ra, sau khi điền đầy đủ thông tin người dùng có thể ấn nút add ở cuối form, khi đó hệ thống sẽ sử dụng api có sẵn để gửi request với method post lên server yêu cầu nhập them sự kiện mới vào database với các thuộc tính tương ứng với những dữ liệu người dùng đã nhập trong form.

Bên phía server, hệ thống sẽ gọi đến phương thức addEvent của event model với tham số là một object gồm 3 thuộc tính được lấy từ client: title, date, description, model sẽ gửi query lên database để thêm mới một sự kiện với các thuộc tính có giá trị tương ứng.

### 2.1.5. Sửa, xóa sự kiện



Người dùng có thể click vào một sự kiện bảng sự kiện, một edit form sẽ được hiện ra với các thông tin của sự kiện đã được chọn, người dùng có thể chỉnh sửa thông tin, sau khi hoàn tất người dùng có thể chọn 1 trong 2 hành động là lưu hoặc xóa sự kiện, sau khi người dùng chọn hành động, hệ thống sẽ sử dụng các api tương ứng để gửi request lên server để cập nhật hoặc xóa sự kiện hiện tại khỏi database.

Bên phía server, hệ thống sẽ gọi đến phương thức updateEvent hoặc deleteEvent tương ứng của event model với tham số là một object gồm 4 thuộc tính được lấy từ client: id, title, date, description, model sẽ gửi query lên database để tìm ra sự kiện với id tương ứng để sửa hoặc xóa theo yêu cầu của client

## 2.2. Yêu cầu Phi chức năng:

2.2.1. Giao diện Người dùng:

Giao diện người dùng phải được thiết kế đơn giản và trực quan để người dùng dễ dàng tương tác và sử dụng.

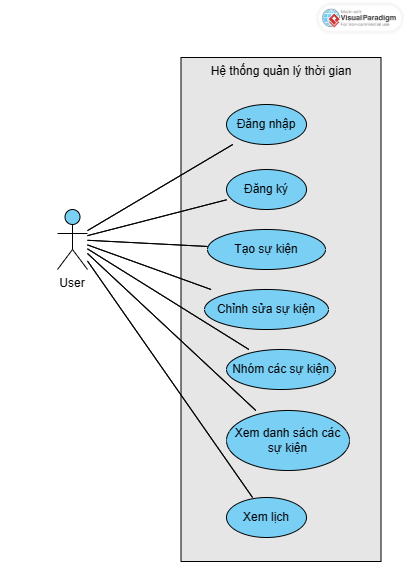
Phải có các hướng dẫn rõ ràng và hỗ trợ sẵn sàng để giúp người dùng làm quen và sử dụng ứng dụng một cách hiệu quả.

2.2.2. Bảo mật:

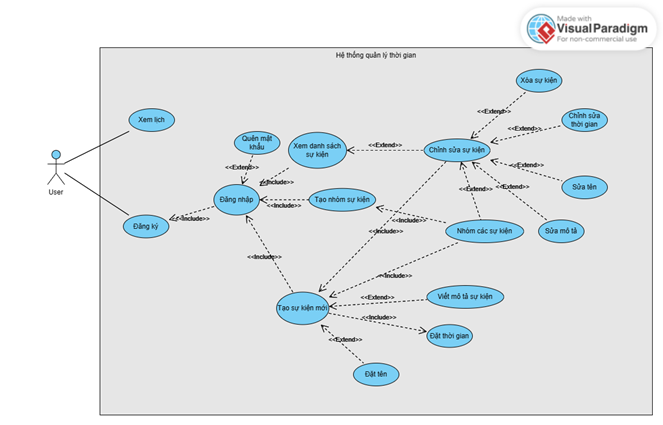
Dữ liệu người dùng phải được bảo vệ an toàn và tuân thủ các tiêu chuẩn bảo mật như mã hóa dữ liệu và xác thực hai yếu tố.

Cần có các biện pháp bảo vệ chống lại các cuộc tấn công mạng và lỗ hổng bảo mật để đảm bảo tính bảo mật của hệ thống và thông tin người dùng.

## 2.3 Mô hình ca sử dụng tổng quan

.

## 2.4. Mô hình ca sử dụng phân rã



# 3. Phân tích kiến trúc và hành vi của hệ thống

## 3.1. Phân tích kiến trúc

### 3.1.1. Các Tầng:

* Giao diện Người dùng (User Interface Layer): Đây là phần tương tác trực tiếp với người dùng, cung cấp giao diện để họ nhập liệu, xem và chỉnh sửa lịch sự kiện.
* Logic Ứng dụng (Application Logic Layer): Tầng này chứa logic ứng dụng, bao gồm các hàm xử lý sự kiện, quản lý người dùng và sự kiện, và tương tác với các tầng dưới cùng.
* Dữ liệu (Data Layer): Đây là nơi lưu trữ dữ liệu của ứng dụng, bao gồm thông tin về người dùng, sự kiện, nhóm và cài đặt.

### 3.1.2. Các Khái niệm Cốt lõi:

* Người dùng (User): Đại diện cho người dùng của ứng dụng Calendy.
* Sự kiện (Event): Đại diện cho các sự kiện trong lịch cá nhân của người dùng, bao gồm tiêu đề, thời gian, địa điểm và mô tả.
* Nhóm (Group): Đại diện cho các nhóm sự kiện mà người dùng có thể tạo để tổ chức và phân loại sự kiện.

### 3.1.3. Analysis Mechanisms (Các Cơ chế Phân tích):

* Tích hợp và Tương tác: Cơ chế này giúp Calendy tích hợp và tương tác với các dịch vụ và ứng dụng khác, như Google Calendar, Outlook hoặc các ứng dụng di động.
* Quản lý Dữ liệu và Cơ sở dữ liệu: Cơ chế này xác định cách Calendy quản lý và lưu trữ dữ liệu người dùng và sự kiện trong cơ sở dữ liệu.

## 3.2. Phân tích hành vi:

### 3.2.1. Use case realization:

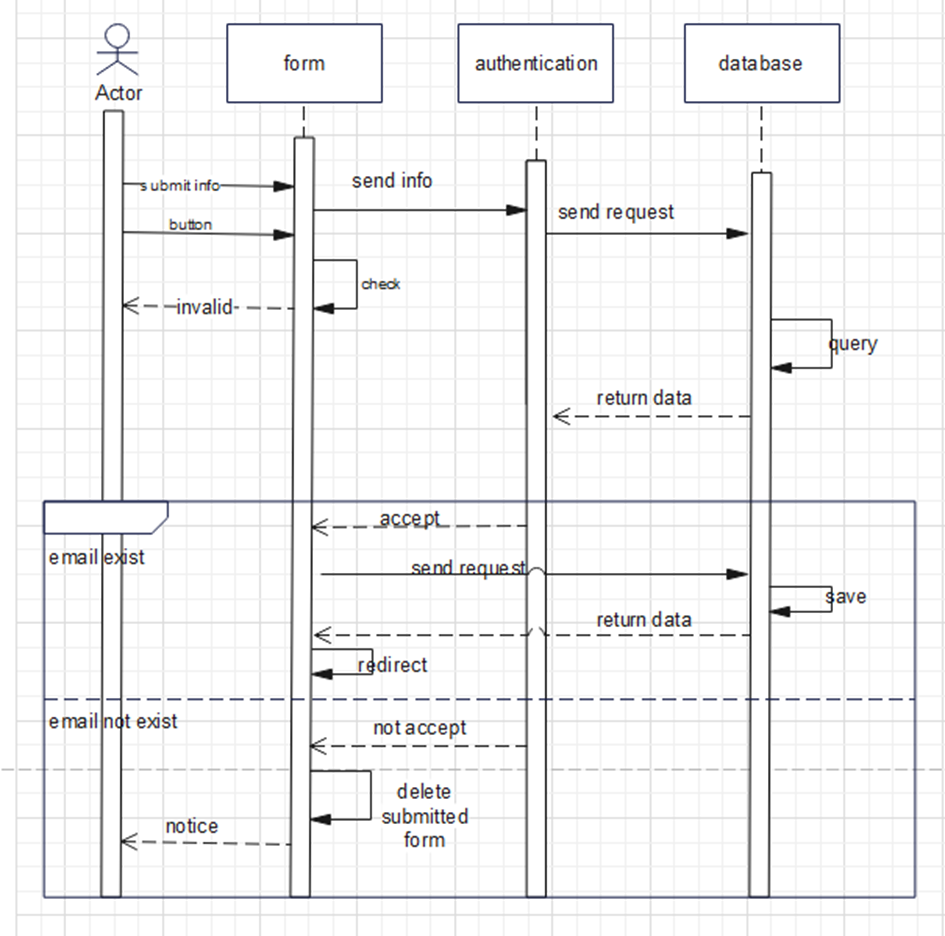


Figure 3.1: Biểu đồ tuần tự của chức năng “Đăng ký”

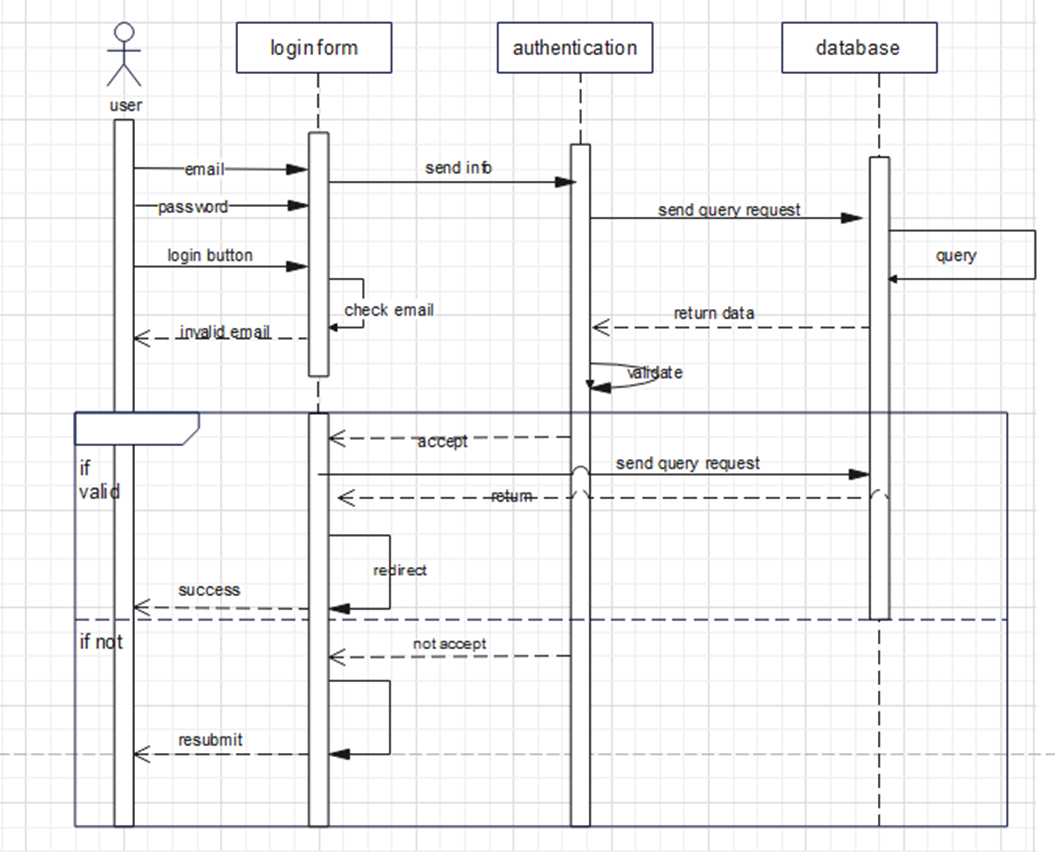


Figure 3.2: Biểu đồ tuần tự của chức năng “Đăng nhập”

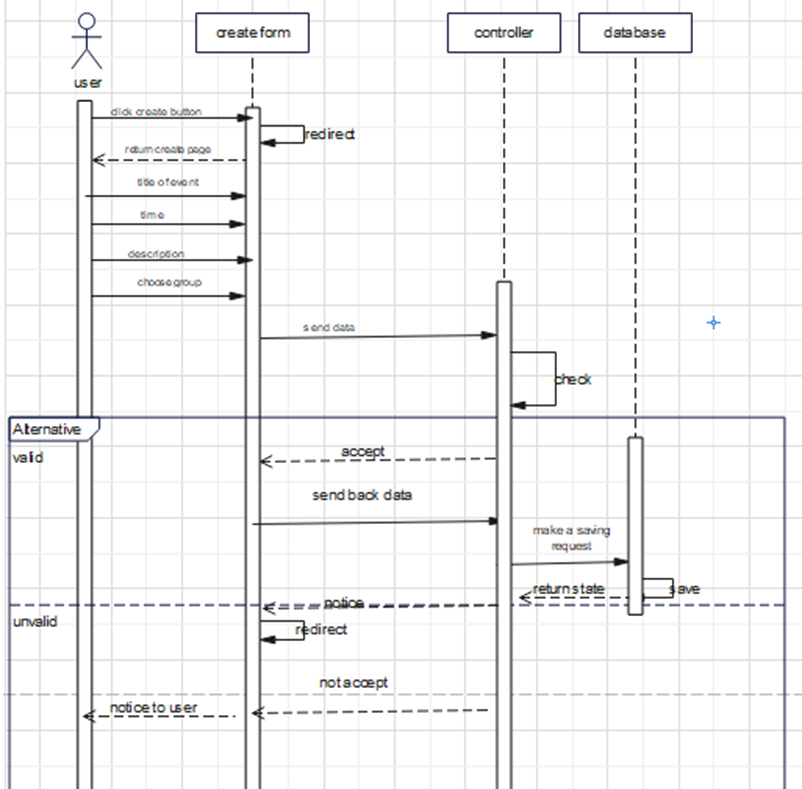


Figure 3.3: Biểu đồ tuần tự của chức năng “Tạo sự kiện”

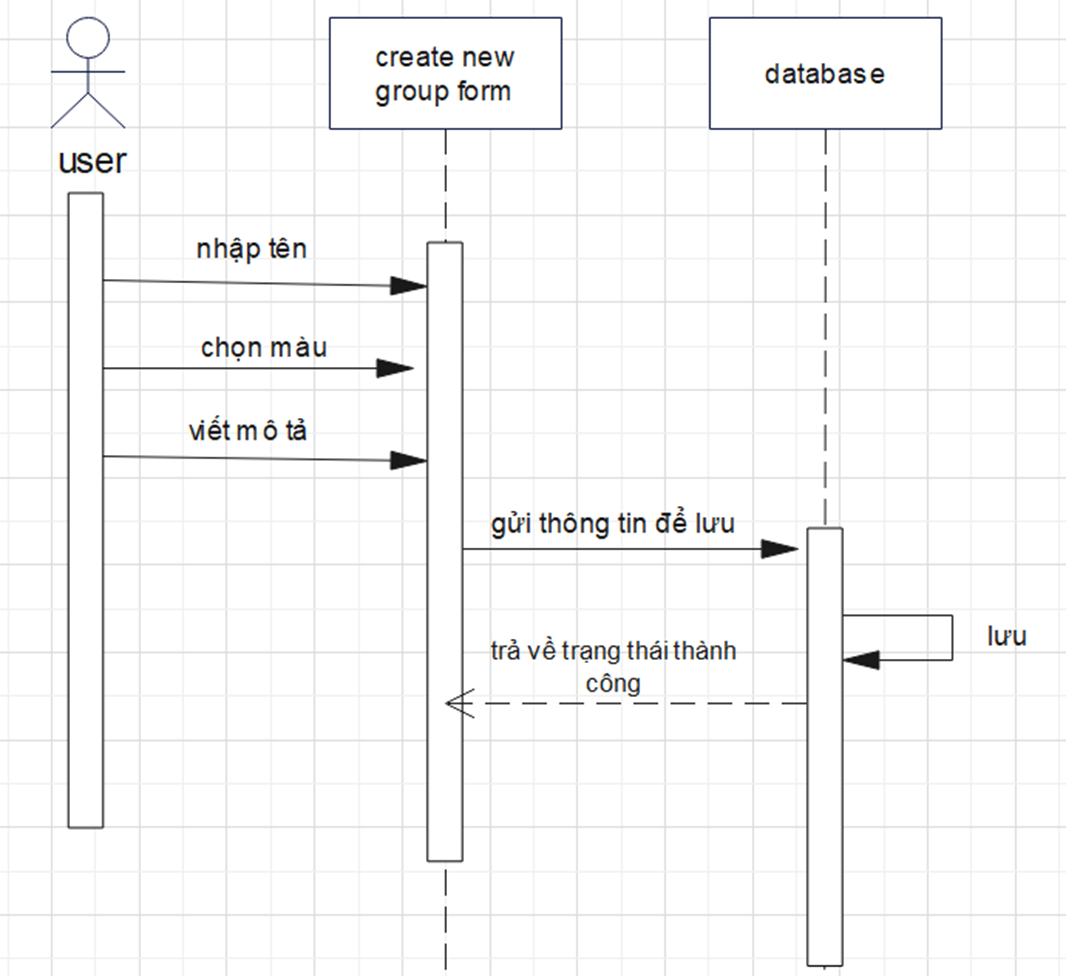


Figure 3.4: Biểu đồ tuần tự của chức năng “Tạo nhóm sự kiện”

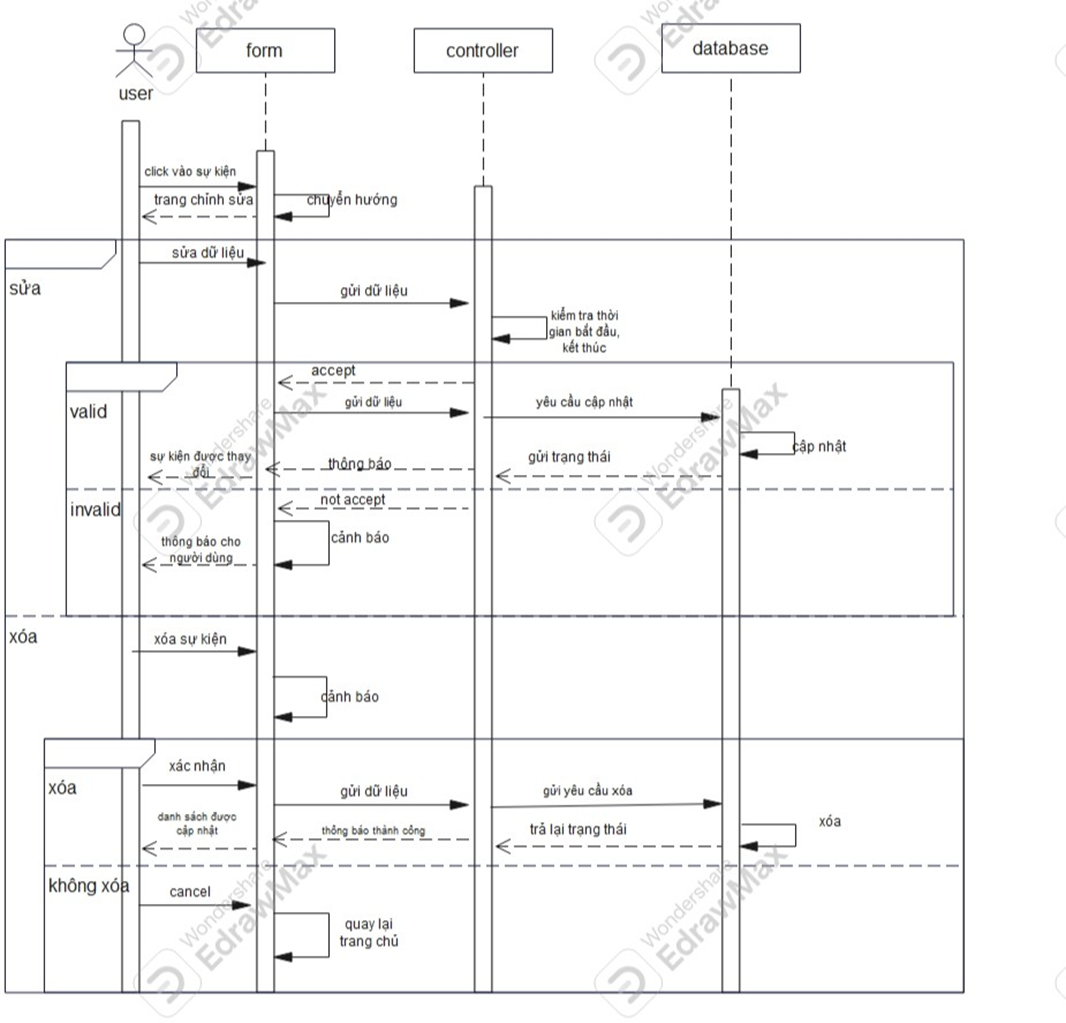
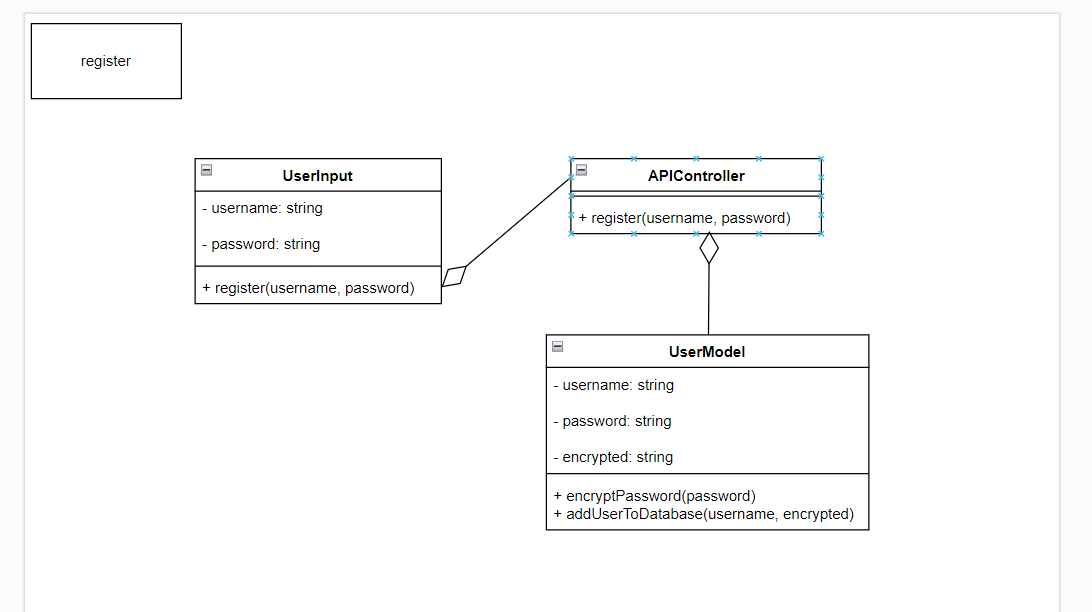
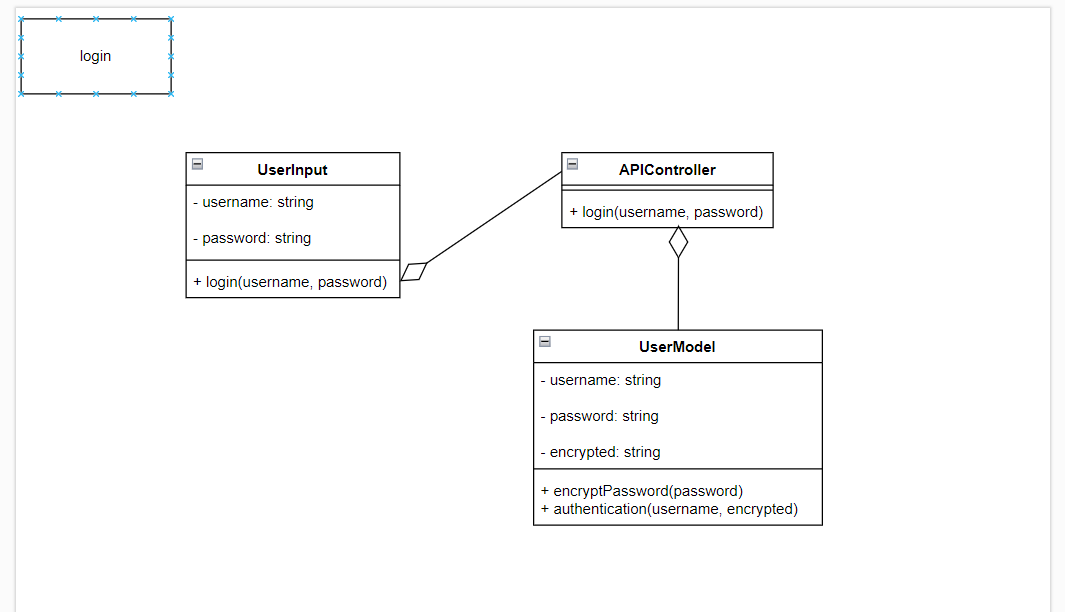


Figure 3.5: Biểu đồ tuần tự của chức năng “Chỉnh sửa sự kiện”

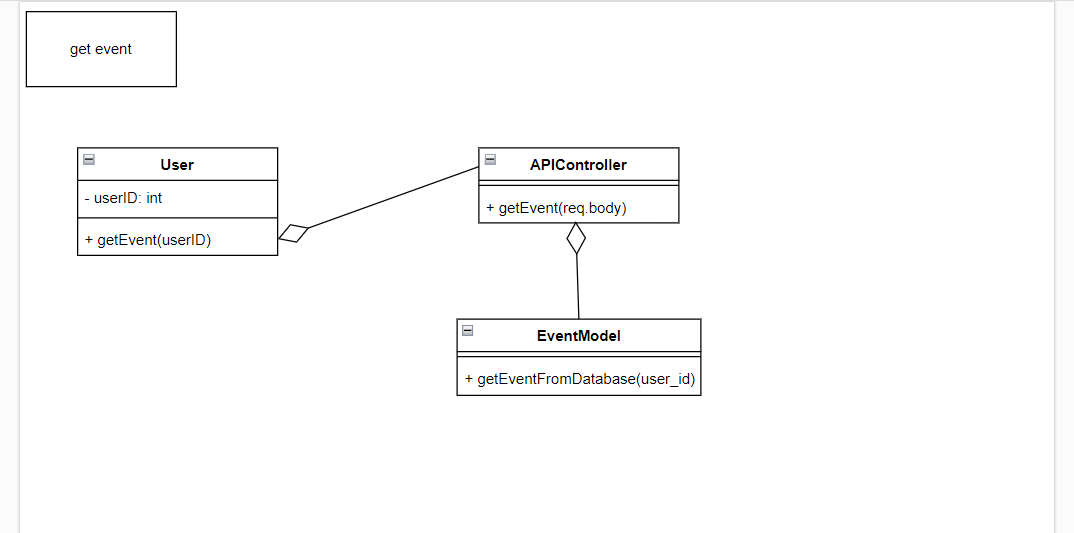
### 3.2.2. Mô hình phân tích



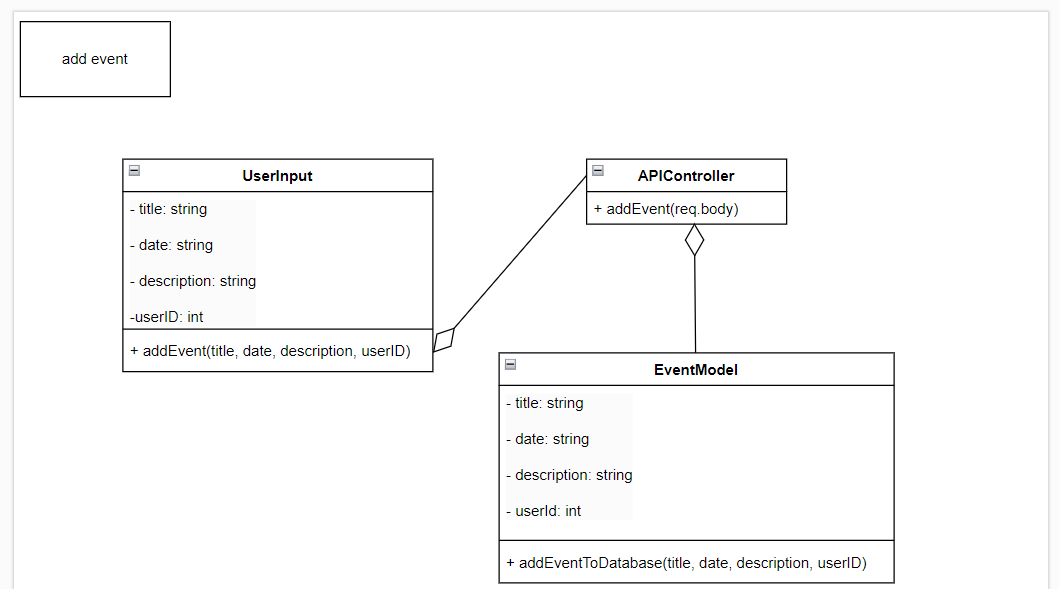
1. Đăng ký



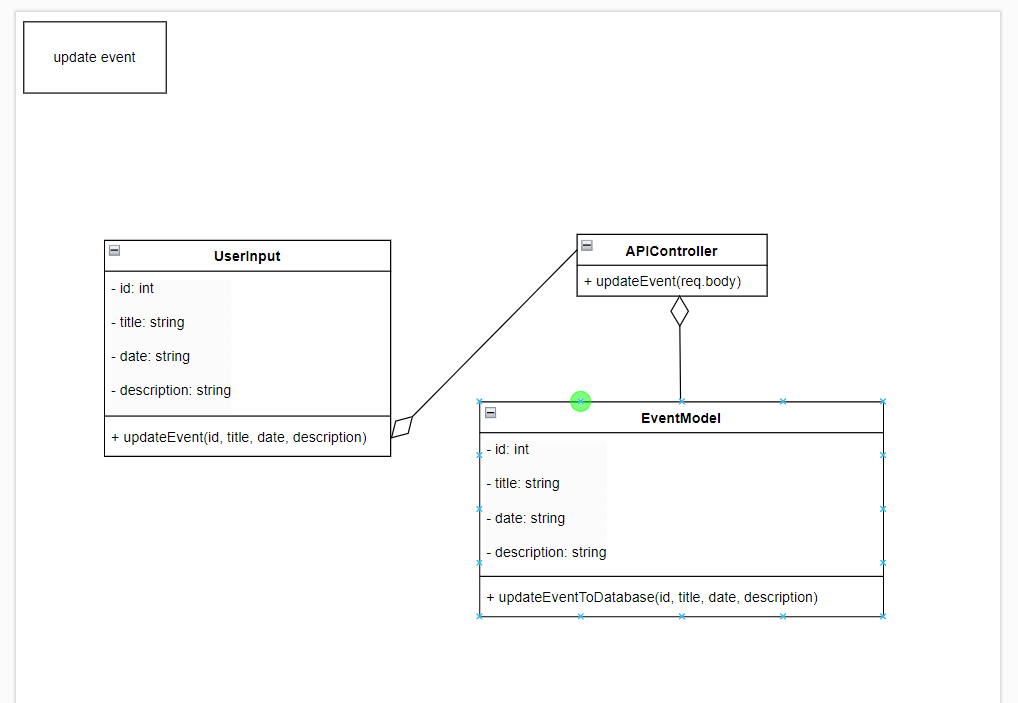
1. Đăng nhập



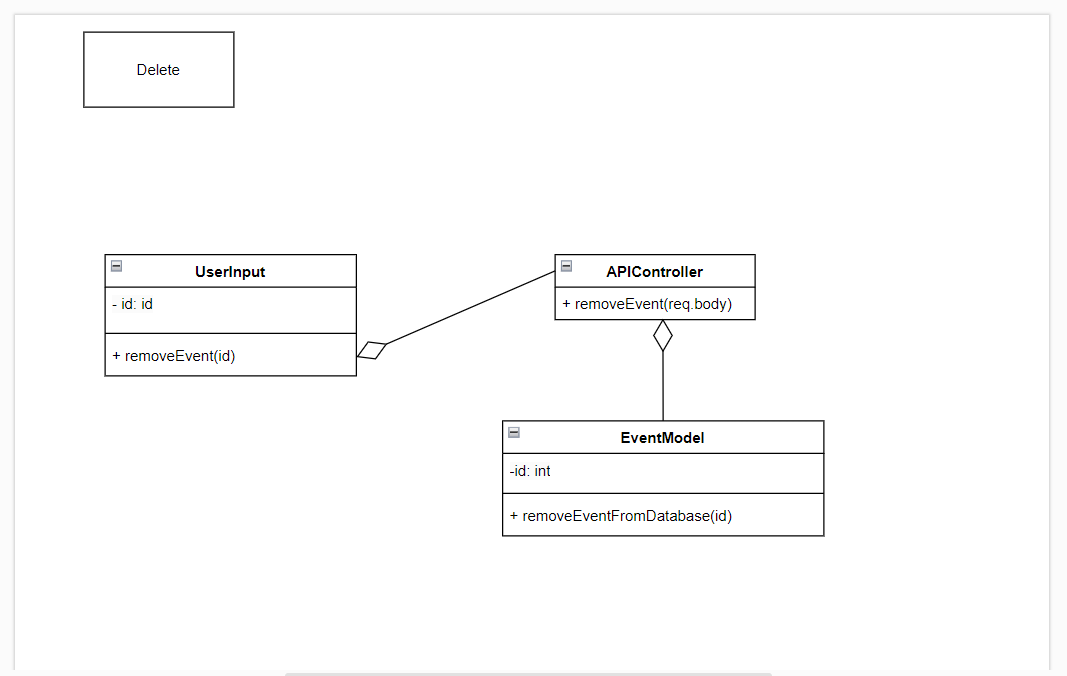
4) Lấy sự kiện từ database



5) Thêm mới sự kiện



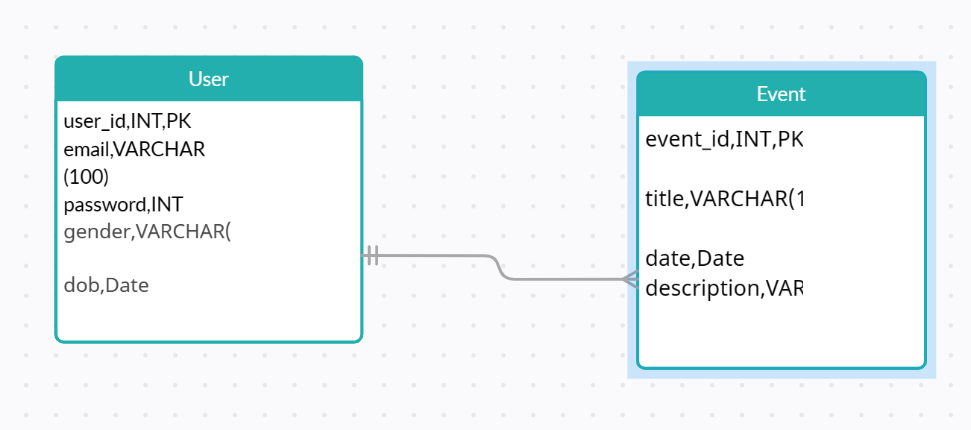
6) Sửa sự kiện trong database



7) Xóa sự kiện khỏi database

# 4. Phân tích hệ thống

## 4.1. Lược đồ cơ sở dữ liệu



### 4.2. Biểu đồ Layered Architecture