



# **Software Requirements Specification (SRS)**

**Môn: Phân tích và thiết kế hệ thống**

Dự án: Nền tảng Quản lý Công việc Nhóm “WorkFlow Mini”

Sinh viên thực hiện: Trương Hà Cẩm Linh - MSSV: 122

*Ngày 16 tháng 11 2025*

# Mục lục

CHƯƠNG 1. GIỚI THIỆU .....	1
1.1 Mục đích .....	1
1.2 Phạm vi & Bối cảnh .....	1
1.3 Độc giả và bên liên quan .....	1
1.4 Thuật ngữ .....	1
1.5 Tài liệu tham khảo .....	2
1.6 Vấn đề cụ thể cần giải quyết .....	2
1.7 Phạm vi (In/Out) .....	2
CHƯƠNG 2. MÔ TẢ TỔNG QUAN .....	2
2.1 Bối cảnh sản phẩm .....	2
2.2 Kiến trúc 3 tầng .....	3
2.4 Người dùng .....	5
2.5 Chức năng chính .....	5
2.6 Ràng buộc nghiệp vụ .....	5
2.7 Ràng buộc kỹ thuật .....	6
2.8 Giả định và phụ thuộc .....	6
Chương 3. YÊU CẦU CỤ THỂ .....	6
3.1 Yêu cầu UI .....	6
3.2 Yêu cầu chức năng chi tiết .....	6
3.3 Phi chức năng .....	10
3.4 Thiết kế Cơ sở dữ liệu .....	10
PHỤ LỤC .....	15
A. Biểu đồ Lớp .....	15
B. ác biểu đồ hành vi chi tiết khác .....	16

## CHƯƠNG 1. GIỚI THIỆU

### 1.1 Mục đích

Tài liệu này nêu các yêu cầu cho phiên bản thử nghiệm (MVP) của “WorkFlow Mini” — một ứng dụng giúp nhóm nhỏ tổ chức, phân công và theo dõi công việc hằng ngày. Mục tiêu chính là:

- Mô tả rõ ràng chức năng cần có để phát triển và nghiệm thu MVP.
- Đặt ra các tiêu chí kiểm thử cho QA và hướng dẫn triển khai cho đội Dev/DevOps.
- Làm rõ phạm vi, ràng buộc và giả định nhằm tránh mở rộng chức năng ngoài yêu cầu.

### 1.2 Phạm vi & Bối cảnh

WorkFlow Mini phục vụ cho nhóm nhỏ (ví dụ 3–10 người) cần công cụ quản lý công việc nội bộ: giao task, theo dõi trạng thái, trao đổi nhanh và lưu tài liệu liên quan. Bản MVP tập trung vào:

- Tạo Team, quản lý thành viên, phân quyền cơ bản.
- Tạo Project kèm bảng Kanban 3 cột mặc định.
- Task CRUD, gán assignee, comment, đính kèm file (giới hạn), kéo-thả thay đổi trạng thái.
- Thông báo nội bộ và công cụ báo cáo cơ bản cho trưởng nhóm.
- Không bao gồm: public API mở, tích hợp thanh toán, theo dõi thời gian chi tiết hoặc mobile native app.

### 1.3 Độc giả và bên liên quan

- Chủ sở hữu sản phẩm / Khách hàng: Chị Mai (ví dụ) — người dùng quản lý nhóm, cung cấp yêu cầu và nghiệm thu.
- Người dùng cuối: Quản lý (Manager) và Nhân viên (Member).
- Đội phát triển: Frontend, Backend, DevOps.
- Đội kiểm thử (QA): chịu trách nhiệm test chức năng và phi chức năng.
- Người quản trị hệ thống (SysAdmin): chịu trách nhiệm triển khai và backup.

### 1.4 Thuật ngữ

- Task: thẻ công việc
- Kanban: bảng theo cột trạng thái
- Assignee: người được giao Task

- Attachment: tệp đính kèm

### 1.5 Tài liệu tham khảo

- Bản yêu cầu nghiệp vụ nguồn (Yêu cầu nghiệp vụ.md)
- Template SRS (Template SRS.md)

### 1.6 Vấn đề cụ thể cần giải quyết

- Giao việc phân tán, thiếu lịch sử rõ ràng.
- Không có view tổng quan dễ hiểu cho quản lý.
- Tệp đính kèm và thông tin công việc rải rác.
- Thiếu nhắc nhở dẫn tới bỏ sót deadline.

### 1.7 Phạm vi (In/Out)

#### a. Các chức năng có trong hệ thống (In-scope):

- Authentication: Đăng nhập, phân quyền JWT
- Team & Project management: Tạo team, thêm thành viên, tạo dự án
- Kanban 3-cột: Bảng quản lý công việc với drag-drop
- Task CRUD: Tạo, sửa, xóa task, gán người thực hiện
- Comments: Thảo luận và trao đổi trên task
- Attachments: Đính kèm file (tối đa 10MB)
- In-app notifications: Thông báo nội bộ real-time
- Activity log: Lịch sử hoạt động
- Simple reports: Báo cáo cơ bản cho manager

#### b. Các chức năng không có trong hệ thống (Out-of-scope):

- Hệ thống không có Public API mở cho bên thứ ba
- Hệ thống không tích hợp Payment gateway
- Hệ thống không có Advanced integrations (Slack, Jira, ...)
- Hệ thống không có Time tracking chi tiết
- Hệ thống không có Mobile native app (iOS/Android)

## CHƯƠNG 2. MÔ TẢ TỔNG QUAN

### 2.1 Bối cảnh sản phẩm

WorkFlow Mini là ứng dụng web độc lập nhằm tập trung hoá công việc của nhóm. Có khả năng triển khai trên máy chủ nội bộ hoặc cloud. Thành phần chính: UI (SPA), Backend API, Database quan hệ, Object Storage, Worker để xử lý thông báo.

## 2.2 Kiến trúc 3 tầng

Giữ mô hình Presentation / Application / Data để tách concerns.

### a. Presentation

- SPA (React hoặc Vue)
- Chịu trách nhiệm hiển thị Kanban, forms, dashboard và thực hiện tương tác kéo-thả.

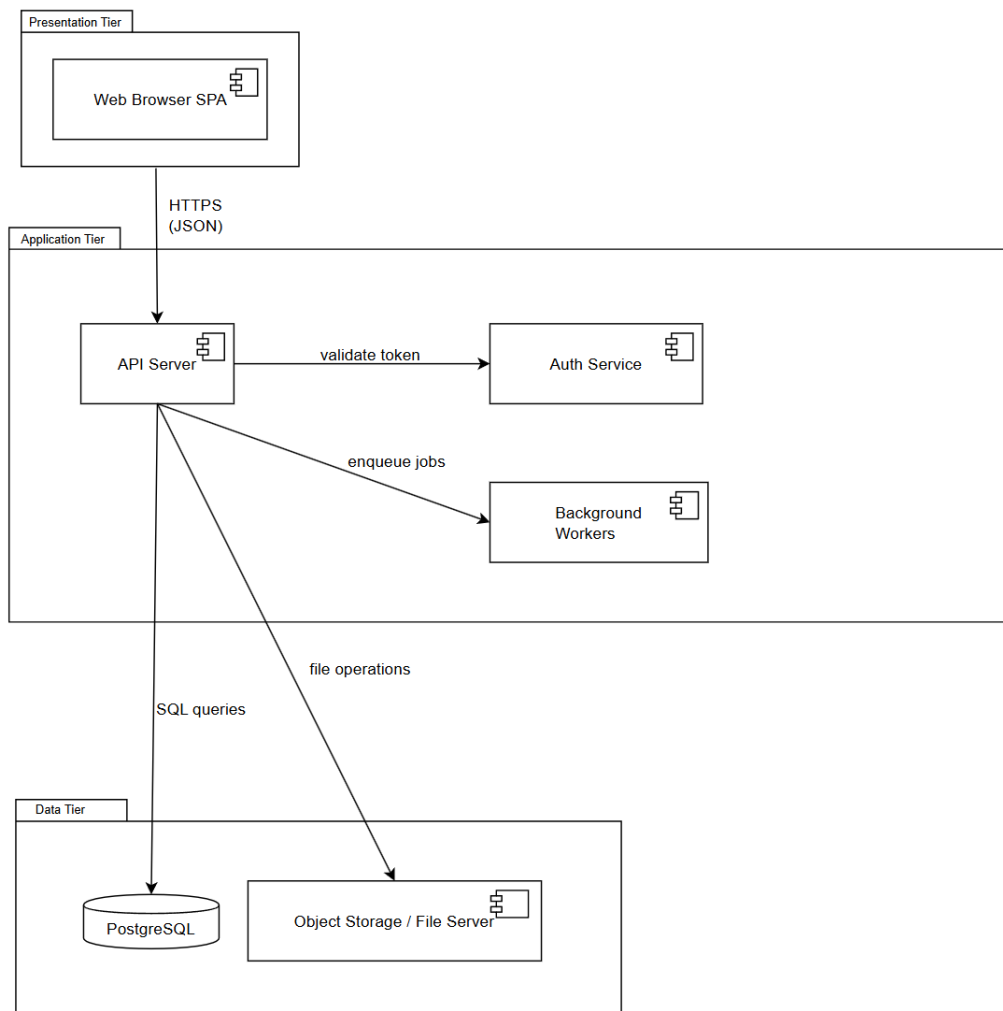
### b. Application

- API server (Node.js/Express hoặc tương đương).
- Xác thực JWT, RBAC, xử lý logic nghiệp vụ, enqueue job cho Worker

### c. Data

- PostgreSQL cho dữ liệu quan hệ.
- Object Storage (hoặc filesystem) cho attachments).

### d. Sơ đồ



*Sơ đồ 3 tầng*

## 2.3 Biểu đồ Use Case đầy đủ

Hệ thống cung cấp 13 use case chính với các quan hệ include, extend và inheritance để thể hiện logic nghiệp vụ và phân quyền.

### a. Sơ đồ Use Case chi tiết



*Sơ đồ Use Case chi tiết*

### b. Mô tả 13 Use Cases

1. **UC-01: Đăng nhập** - Người dùng xác thực bằng email và mật khẩu để truy cập hệ thống
2. **UC-02: Quản lý Nhóm** - Manager tạo team mới, thêm/xóa thành viên, phân quyền

3. **UC-03: Khởi tạo Dự án** - Manager tạo project thuộc team với Kanban 3 cột tự động
4. **UC-04: Xem Dashboard** - Manager xem tổng quan tiến độ toàn bộ dự án
5. **UC-05: Truy cập Kanban** - Thành viên xem bảng công việc theo cột trạng thái
6. **UC-06: Thêm Task mới** - Thành viên tạo task, nhập thông tin, gán người làm
7. **UC-07: Gán người thực hiện** - Chỉ định một hoặc nhiều assignees cho task
8. **UC-08: Di chuyển Task** - Kéo-thả task giữa các cột để cập nhật trạng thái
9. **UC-09: Thảo luận Task** - Thêm comment, trao đổi về task
10. **UC-10: Tải file lên** - Đính kèm tệp tài liệu vào task ( $\leq 10\text{MB}$ )
11. **UC-11: Nhận thông báo** - Người dùng nhận notification khi có sự kiện liên quan
12. **UC-12: Truy xuất Báo cáo** - Manager xem báo cáo tiến độ, tasks quá hạn
13. **UC-13: Cảnh báo Deadline** - Hệ thống tự động gửi cảnh báo task sắp đến hạn

## 2.4 Người dùng

- Manager: quyền quản trị team/project, xem báo cáo.
- Member: thao tác trên task, comment, upload file.
- System/Scheduler: actor tự động sinh cảnh báo.

## 2.5 Chức năng chính

- Auth & RBAC
- Team & Project management
- Kanban board và drag-drop
- Task management, comments, attachments
- Notifications, activity log, simple reporting

## 2.6 Ràng buộc nghiệp vụ

- BR-01: Chỉ Manager tạo/sửa Project
- BR-02: Task thuộc đúng 1 Project
- BR-03: Attachment  $\leq 10\text{MB}$
- BR-04: Data phân vùng theo Team
- BR-05: Hệ thống gửi cảnh báo khi còn  $\leq 24\text{h}$
- BR-06: Project khởi tạo sẵn 3 lists
- BR-07: Member chỉ di chuyển task nếu được phép

## 2.7 Ràng buộc kỹ thuật

- HTTPS, JWT, mật khẩu hash (bcrypt/argon2)
- UI phản hồi nhanh, tối ưu cho nhóm nhỏ
- Backup định kỳ

## 2.8 Giả định và phụ thuộc

- Trình duyệt hiện đại, kết nối internet
- PostgreSQL sẵn có, storage cài đặt

# Chương 3. YÊU CẦU CỤ THỂ

## 3.1 Yêu cầu UI

### a. Nguyên tắc Thiết kế UI/UX

- Responsive Design: Hỗ trợ desktop (1920 \* 1080), table (768 \* 1024), mobile (375 \* 667).
- Performance: First Contentful Paint < 1.5s, drag-drop latency < 100ms.
- Accessibility: WCAG 2.1 Level AA, keyboard navigation, screen reader support.
- Consistency: Material Design System với color palette nhất quán.
- Real-time: WebSocket updates cho collaborative feature.

### b. WifeFrame

[Click vào đây](#)

### c. Figma

[Click vào đây](#)

## 3.2 Yêu cầu chức năng chi tiết

### a. Tóm tắt các Use Cases chính

- UC-01: Đăng nhập
  - Input: Email, mật khẩu.
  - Output: JWT token, redirect đến Dashboard.
  - Actor: Tất cả người dùng.
- UC-02: Quản lý nhóm:
  - Chức năng: Tạo team, chỉnh sửa thông tin, thêm/xoá thành viên, set role.
  - Actor: Manager.
- UC-03: Khởi tạo Dự án
  - Chức năng: Tạo project, gán vào team, tự động sinh 3 lists (To Do, In Progress, Done).
  - Actor: Manager.
- UC-04: Xem Dashboard

- Hiển thị: tổng tasks, % hoàn thành, tasks overdue, biểu đồ tiến độ.
  - Actor: Manger.
  - UC-05: Task CRUD và Assign
    - Chức năng: Tạo/sửa/xoá task, set priority, due date, assign users.
    - Actor: Member.
  - UC-06: Di chuyển Task (Drag-Drop)
    - Chức năng: Comment trên task, upload file.
    - Actor: Member.
  - UC-07: Thảo luận và Đính kèm
    - Chức năng: Comment trên task, upload file.
    - Actor: Member.
  - UC-08: Nhận thông báo
    - Triggers: gán task, comment mới, task sắp hết hạn.
    - Actor: Member
  - UC-09: Truy xuất Báo cáo
    - Nội dung: danh sách tasks theo trạng thái, thống kê theo người, timeline.
    - Actor: Manager
- b. Đặc tả Use Case mở rộng**
- UC-06: Thêm Task mới

Thuộc tính	Nội dung
ID	UC-06
Tên	Thêm Task mới
Actor chính	Thành viên (Member)
Stakeholders	Manager (theo dõi tiến độ), Assignees (nhận việc)
Điều kiện tiên quyết	- User đã đăng nhập- User thuộc team của project- Project đang active
Điều kiện sau	- Task được tạo trong list ‘To Do’- Assignees nhận notification- Activity log ghi nhận
Trigger	User nhấn nút “Tạo Task” trên Kanban board
Luồng chính	1. Hệ thống hiển thị form tạo task 2. User nhập title (bắt buộc), description, chọn assignees, set due date, priority 3. User submit form 4. Backend validate: title không rỗng, assignees hợp lệ 5. Hệ thống tạo task record, lưu task_assignees 6. Hệ thống enqueue notification job

	7. Hệ thống ghi activity log 8. Frontend refresh và hiển thị task mới trong cột 'To Do'
Luồng thay thế	<b>3a. Title trống:</b> 3a1. Hiện thị lỗi "Title là bắt buộc" 3a2. Giữ form mở, focus vào trường title <b>3b. Assignee không thuộc team:</b> 3b1. Hiện thị lỗi "Người dùng không hợp lệ" 3b2. Bỏ assignee đó khỏi danh sách <b>3c. Upload file &gt; 10MB:</b> 3c1. Từ chối file, hiển thị lỗi 3c2. Vẫn cho phép tạo task không có file
Yêu cầu đặc biệt	- Form phải responsive, hoạt động tốt trên mobile- Validation phía client và server- Optimistic UI update cho trải nghiệm mượt
Tần suất	Trung bình 10-50 lần/ngày/team

- UC-08: Di chuyển Task

Thuộc tính	Nội dung
ID	UC-08
Tên	Di chuyển Task giữa các cột
Actor chính	Thành viên (có quyền trên task)
Điều kiện tiên quyết	- User đã đăng nhập- User có quyền edit task (assignee hoặc creator)- Task tồn tại
Điều kiện sau	- Task chuyển sang cột mới- Lịch sử thay đổi được lưu- Notification gửi đến người liên quan
Trigger	User kéo-thả task card sang cột khác
Luồng chính	1. User bắt đầu kéo task card 2. Frontend highlight các cột có thể drop 3. User thả task vào cột mới 4. Frontend update UI ngay lập tức (optimistic) 5. Frontend gửi PATCH /tasks/:id {list_id: new_list_id} 6. Backend validate quyền và cập nhật DB 7. Backend ghi activity_log 8. Backend trả 200 OK 9. Frontend confirm thành công
Luồng thay thế	<b>6a. Không có quyền:</b> 6a1. Backend trả 403 Forbidden

	6a2. Frontend rollback UI về trạng thái cũ 6a3. Hiển thị thông báo lỗi <b>6b. Conflict (race condition):</b> 6b1. Backend trả 409 Conflict 6b2. Frontend fetch trạng thái mới nhất 6b3. Hiển thị thông báo “Task đã được cập nhật bởi người khác”
Yêu cầu đặc biệt	- Drag-drop phải mượt (60fps)- Optimistic update < 100ms- Network latency < 500ms- Hỗ trợ undo trong 5 giây
Tần suất	Cao (50-200 lần/ngày/team)

- UC-02: Quản lý Nhóm

Thuộc tính	Nội dung
ID	UC-02
Tên	Quản lý Nhóm (Team Management)
Actor chính	Manager
Điều kiện tiên quyết	- Manager đã đăng nhập- Manager có quyền quản trị team
Điều kiện sau	- Thành viên được thêm/xóa khỏi team- Notification gửi đến user bị ảnh hưởng- Activity log ghi nhận thay đổi
Trigger	Manager truy cập Team Settings
Luồng chính (Thêm thành viên)	1. Manager mở Team Settings 2. Manager chọn “Thêm thành viên” 3. Manager nhập email của user 4. Hệ thống kiểm tra user có tồn tại 5. Manager chọn role (Member/Manager) 6. Hệ thống tạo team_member record 7. Gửi email/notification mời user 8. Hiển thị user trong danh sách team
Luồng thay thế	<b>4a. Email không tồn tại:</b> 4a1. Hiển thị “User không tìm thấy” 4a2. Đề xuất gửi email mời đăng ký <b>4b. User đã là thành viên:</b> 4b1. Hiển thị “User đã trong team” 4b2. Cho phép thay đổi role
Yêu cầu đặc biệt	- Giao diện drag-drop để sắp xếp members- Bulk actions: thêm nhiều user cùng lúc- Search và filter members

- UC-03: Khởi tạo Dự án

Thuộc tính	Nội dung
ID	UC-03
Tên	Khởi tạo Dự án mới
Actor chính	Manager
Điều kiện tiên quyết	- Manager đã đăng nhập- Manager thuộc ít nhất 1 team
Điều kiện sau	- Project được tạo với 3 lists mặc định- Manager redirect đến Project Kanban view
Trigger	Manager chọn “Tạo dự án mới”
Luồng chính	<ol style="list-style-type: none"> <li>1. Hệ thống hiển thị form tạo project</li> <li>2. Manager nhập: tên, mô tả, chọn team, start_date, due_date</li> <li>3. Manager submit</li> <li>4. Backend validate dữ liệu</li> <li>5. Hệ thống tạo project record</li> <li>6. Hệ thống tự động tạo 3 lists: “To Do”, “In Progress”, “Done”</li> <li>7. Ghi activity log</li> <li>8. Redirect đến Kanban board của project</li> </ol>
Luồng thay thế	<p><b>4a. Tên project trống:</b></p> <p>4a1. Hiển thị lỗi validation</p> <p><b>4b. Manager không có quyền trên team:</b></p> <p>4b1. Trả 403, hiển thị lỗi quyền</p>
Yêu cầu đặc biệt	- Cho phép custom tên 3 lists ban đầu- Template projects (clone structure)

### 3.3 Phi chức năng

- Hiệu năng: thao tác cục bộ phản hồi < 500ms
- Bảo mật: hash mật khẩu, HTTPS, RBAC.
- Khả năng mở rộng: thiết kế cho scale sau này.

### 3.4 Thiết kế Cơ sở dữ liệu

#### a. Mô tả các bảng chính

- **users:** Thông tin người dùng
  - id (PK): integer, auto-increment
  - name: varchar(100), họ tên đầy đủ
  - email: varchar(255), unique, dùng để đăng nhập
  - password\_hash: varchar(255), mã hóa bcrypt
  - role: enum('admin', 'user'), phân quyền toàn hệ thống

- avatar\_url: text, nullable
- created\_at, updated\_at: timestamp

**\*\*teams\*\*** - Nhóm làm việc

- id (PK): integer
  - name: varchar(100), tên team
  - description: text, nullable
  - created\_by: integer (FK -> users.id)
  - created\_at, updated\_at: timestamp
- **team\_members**: Thành viên trong team
    - team\_id (FK -> teams.id)
    - user\_id (FK -> users.id)
    - role\_in\_team: enum('manager', 'member')
    - joined\_at: timestamp
    - PRIMARY KEY (team\_id, user\_id)
- **projects**: Dự án
    - id (PK): integer
    - team\_id (FK -> teams.id)
    - name: varchar(150)
    - description: text, nullable
    - start\_date: date, nullable
    - due\_date: date, nullable
    - status: enum('planning', 'active', 'completed', 'archived')
    - created\_by: integer (FK -> users.id)
    - created\_at, updated\_at: timestamp
- **lists**: Cột trong Kanban
    - id (PK): integer
    - project\_id (FK -> projects.id)
    - name: varchar(50), ví dụ: "To Do", "In Progress", "Done"
    - position: integer, thứ tự hiển thị
    - created\_at: timestamp
- **tasks**: Công việc
    - id (PK): integer

- project\_id (FK -> projects.id)
  - list\_id (FK -> lists.id)
  - title: varchar(255), bắt buộc
  - description: text, nullable
  - priority: enum('low', 'medium', 'high')
  - due\_date: date, nullable
  - position: integer, vị trí trong list
  - created\_by: integer (FK -> users.id)
  - created\_at, updated\_at: timestamp
- **task\_assignees:** Người được gán task
    - task\_id (FK -> tasks.id)
    - user\_id (FK -> users.id)
    - assigned\_at: timestamp
    - PRIMARY KEY (task\_id, user\_id)
- **comments:** Bình luận trên task
    - id (PK): integer
    - task\_id (FK -> tasks.id)
    - user\_id (FK -> users.id)
    - content: text
    - created\_at, updated\_at: timestamp
- **attachments:** Tập đính kèm
    - id (PK): integer
    - task\_id (FK -> tasks.id)
    - filename: varchar(255)
    - original\_name: varchar(255)
    - file\_size: bigint, bytes
    - mime\_type: varchar(100)
    - storage\_path: text, đường dẫn lưu file
    - uploaded\_by: integer (FK -> users.id)
    - uploaded\_at: timestamp
- **activity\_logs :** Lịch sử hoạt động
    - id (PK): integer
    - actor\_id (FK -> users.id), người thực hiện

- action: varchar(50), ví dụ: 'create\_task', 'move\_task', 'assign\_user'
- subject\_type: varchar(50), loại đối tượng: 'task', 'project', 'comment'
- subject\_id: integer, id của đối tượng
- metadata: jsonb, thông tin bổ sung
- created\_at: timestamp
- **notifications:** Thông báo
  - id (PK): integer
  - user\_id (FK -> users.id), người nhận
  - type: varchar(50), 'task\_assigned', 'comment\_added', 'due\_soon'
  - title: varchar(255)
  - content: text
  - is\_read: boolean, default false
  - link: text, đường dẫn tới đối tượng liên quan
  - created\_at: timestamp

**b. Sơ đồ ERD**

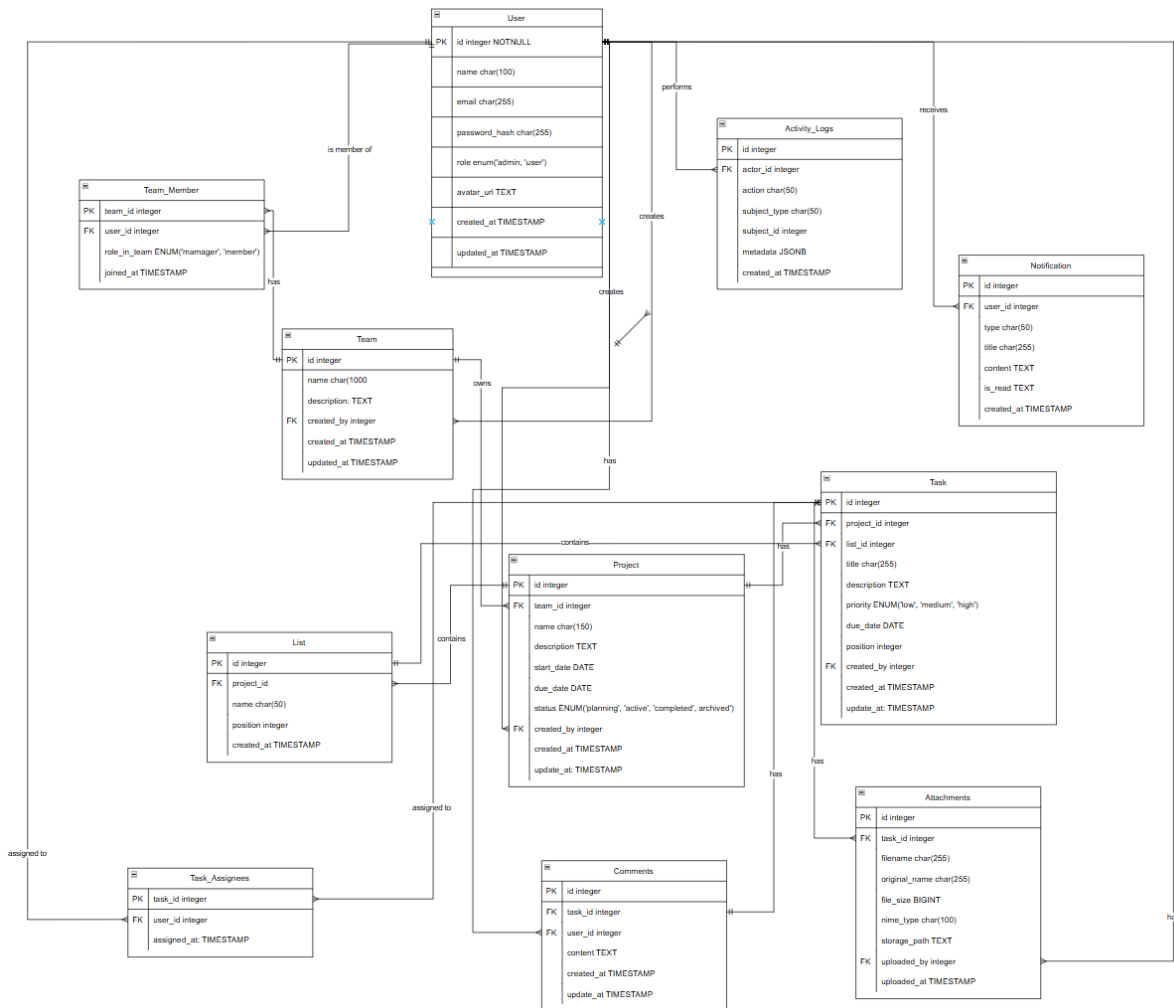


Diagram 1: Sơ đồ ERD

### c. Ràng buộc toàn vẹn

- Cascade delete: khi xóa project → xóa lists và tasks
- Cascade delete: khi xóa task → xóa comments, attachments, task\_assignees
- Check constraints:  $\text{file\_size} \leq 10 * 1024 * 1024$  (10MB)
- Unique constraints: email trong users, (team\_id, user\_id) trong team\_members
- Foreign key constraints: tất cả references phải tồn tại

# PHỤ LỤC

## A. Biểu đồ Lớp

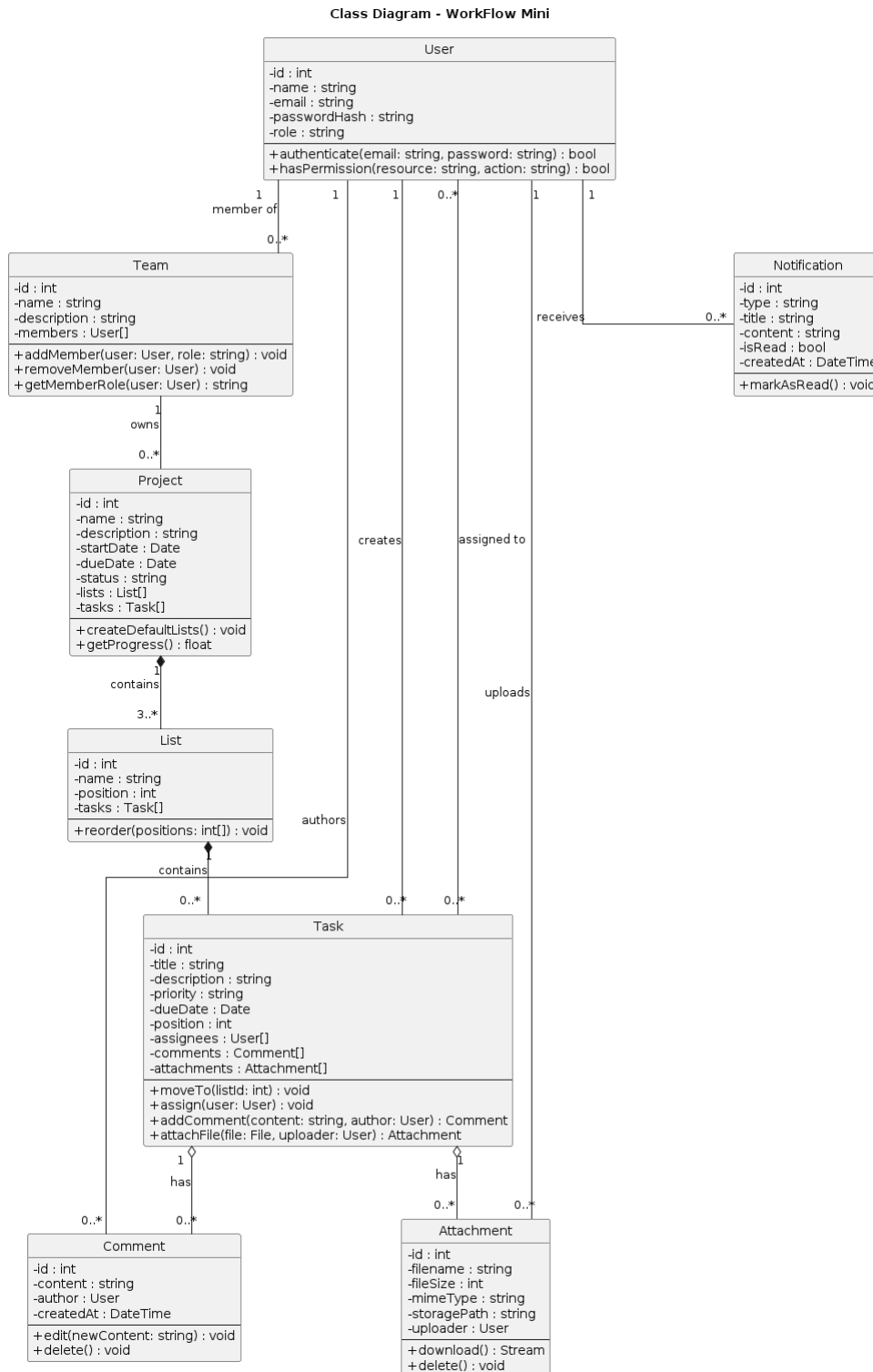


Diagram 2: Class Diagram

## B. ác biểu đồ hành vi chi tiết khác

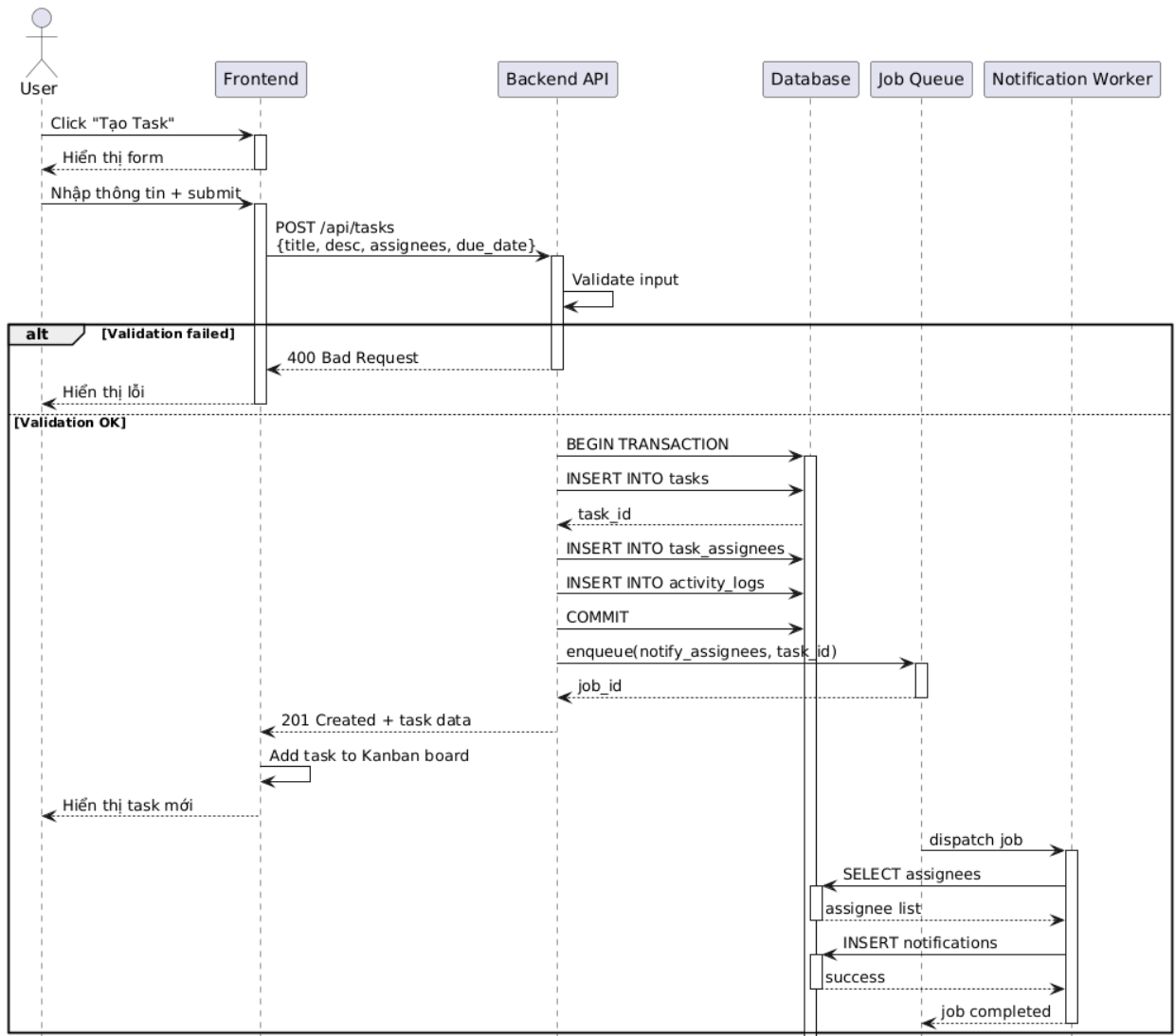


Diagram 3: Sequence Diagram - Tạo Task mới

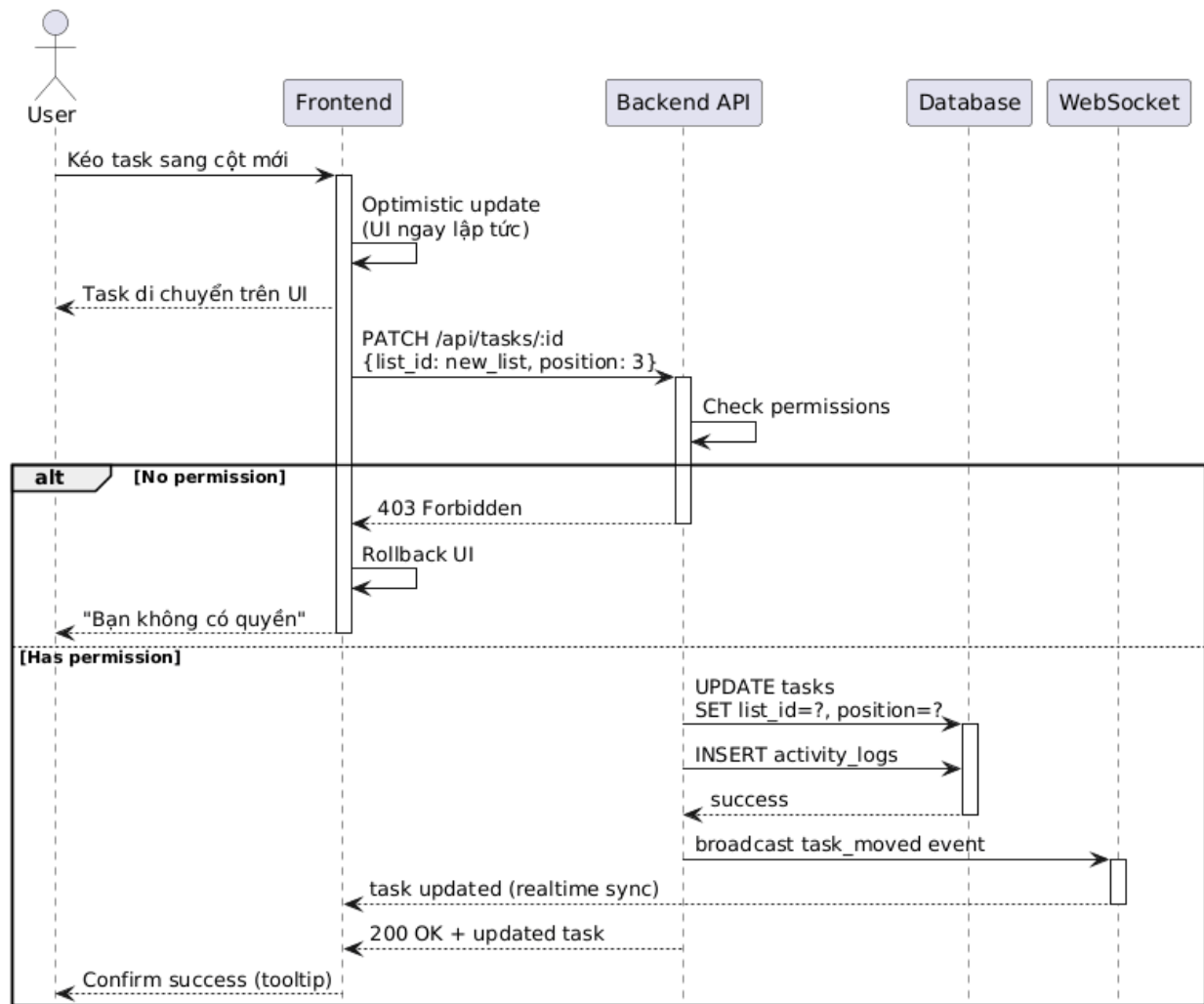


Diagram 4: Sequence Diagram - Kéo-thả Task

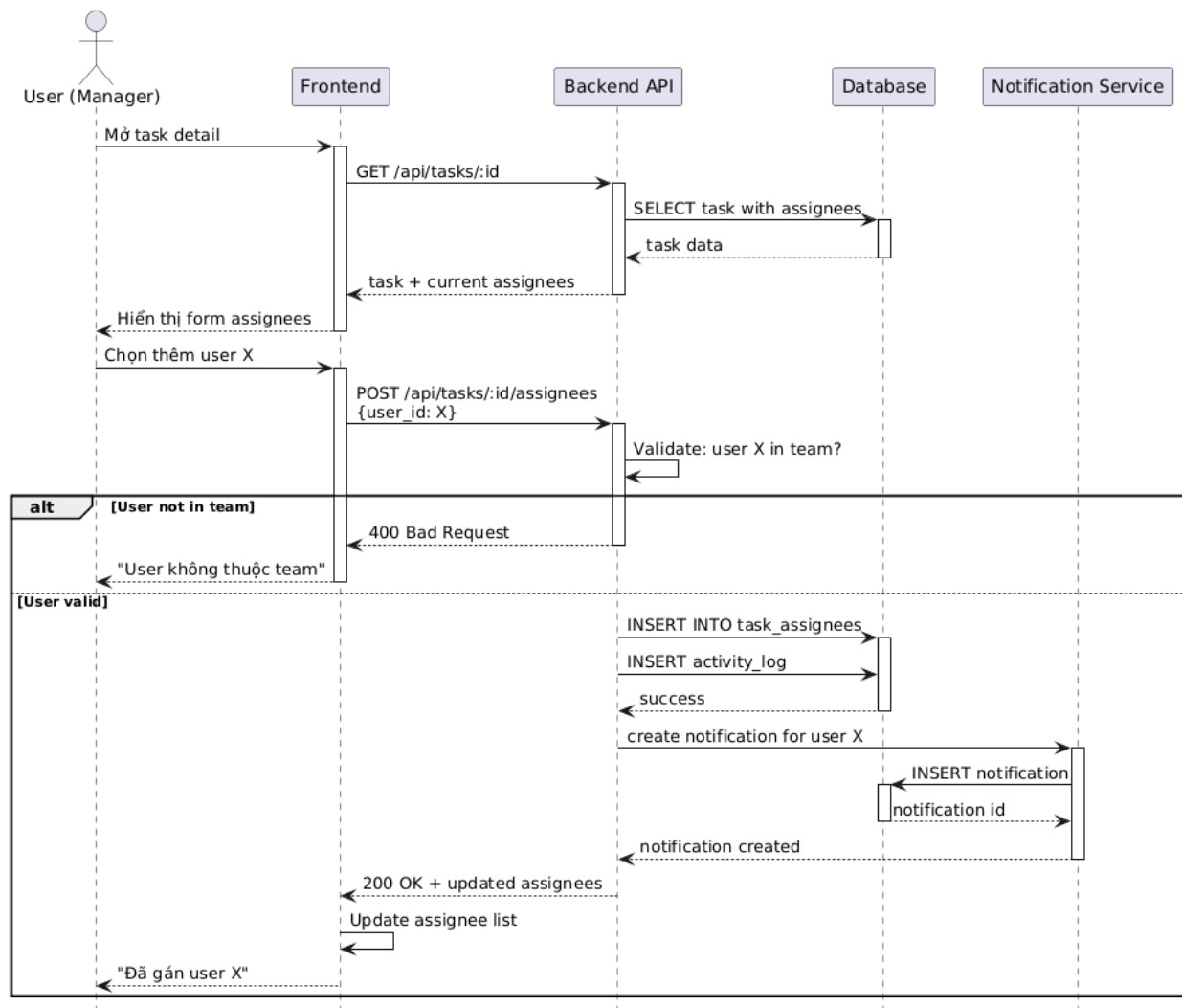


Diagram 5: Sequence Diagram - Gán người thực hiện

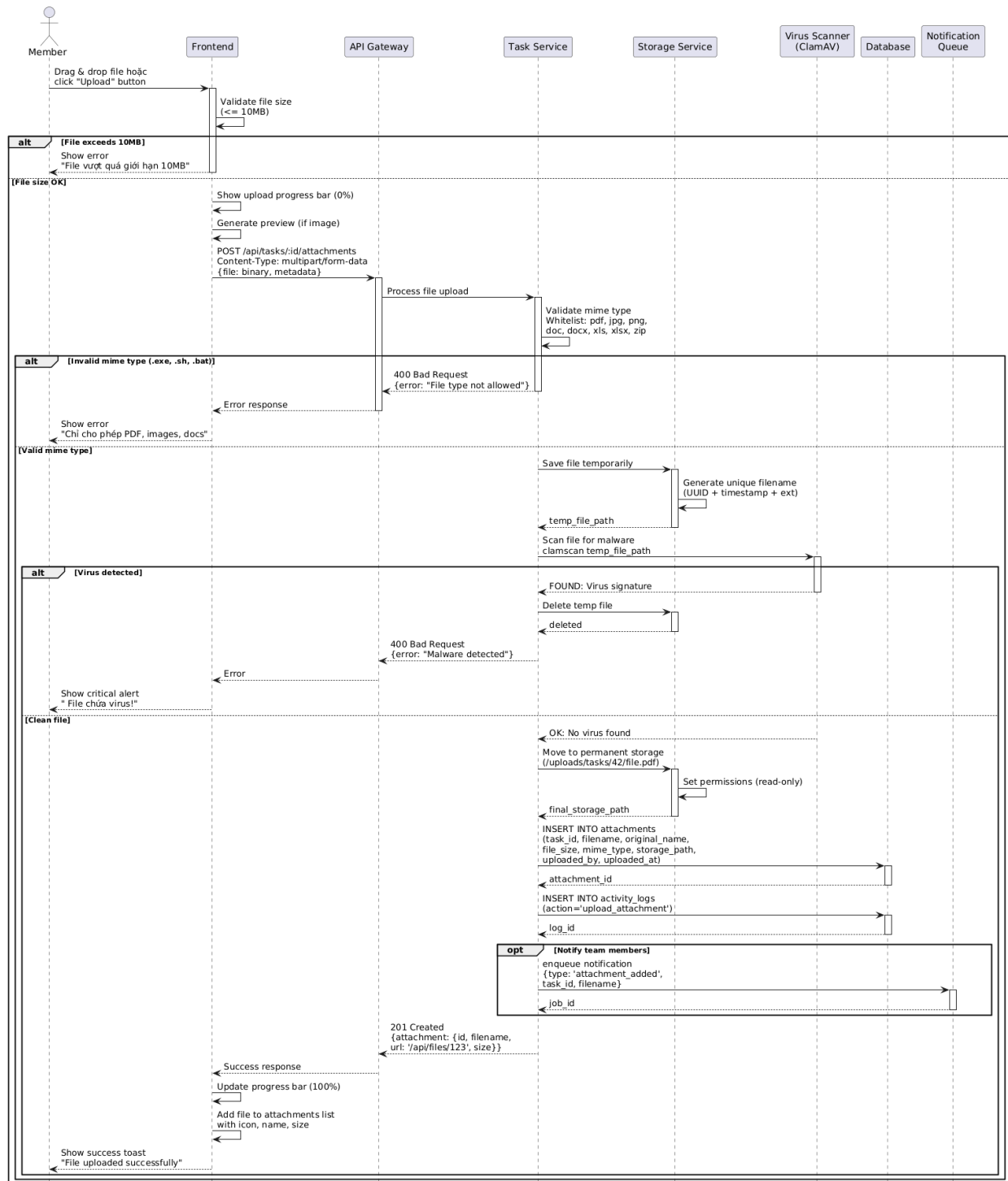


Diagram 6: Sequence Diagram - Upload File và Virus Scan

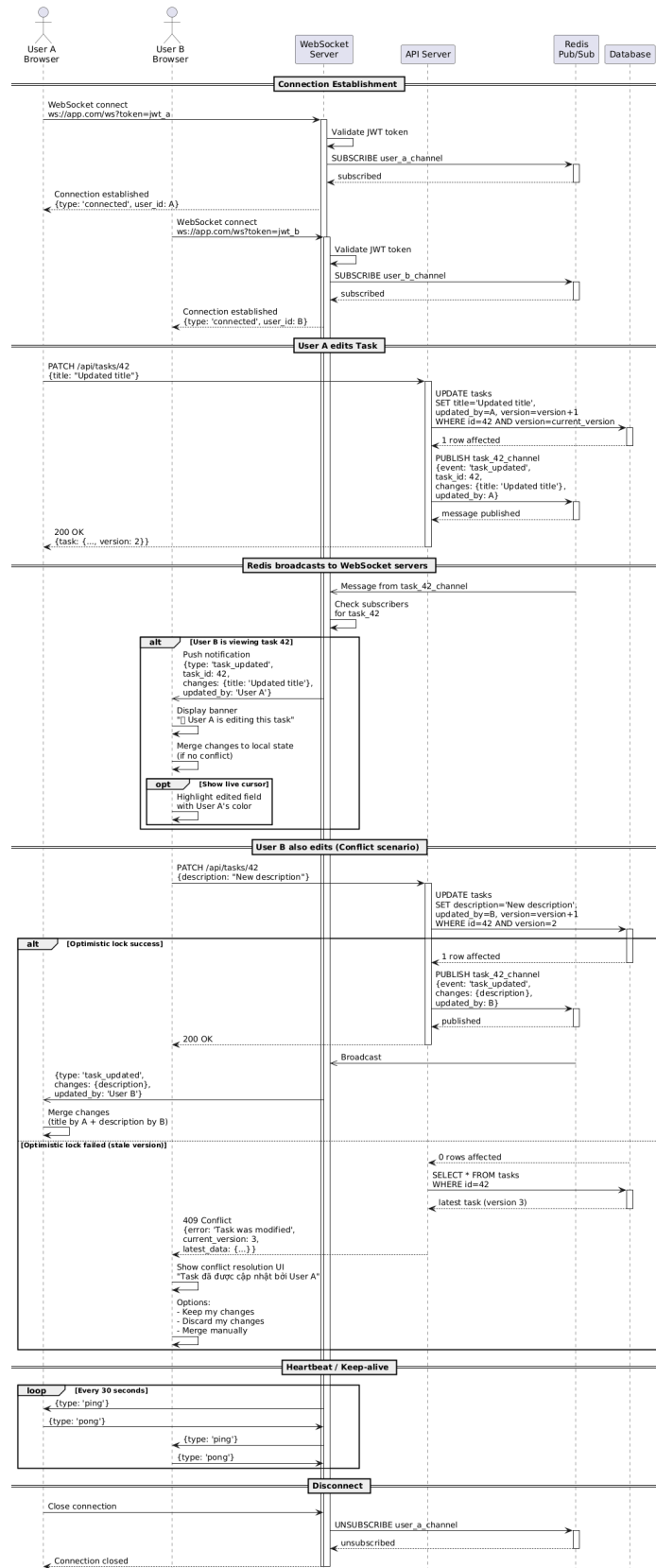


Diagram 7: Sequence Diagram - Real-time Collaboration với WebSocket

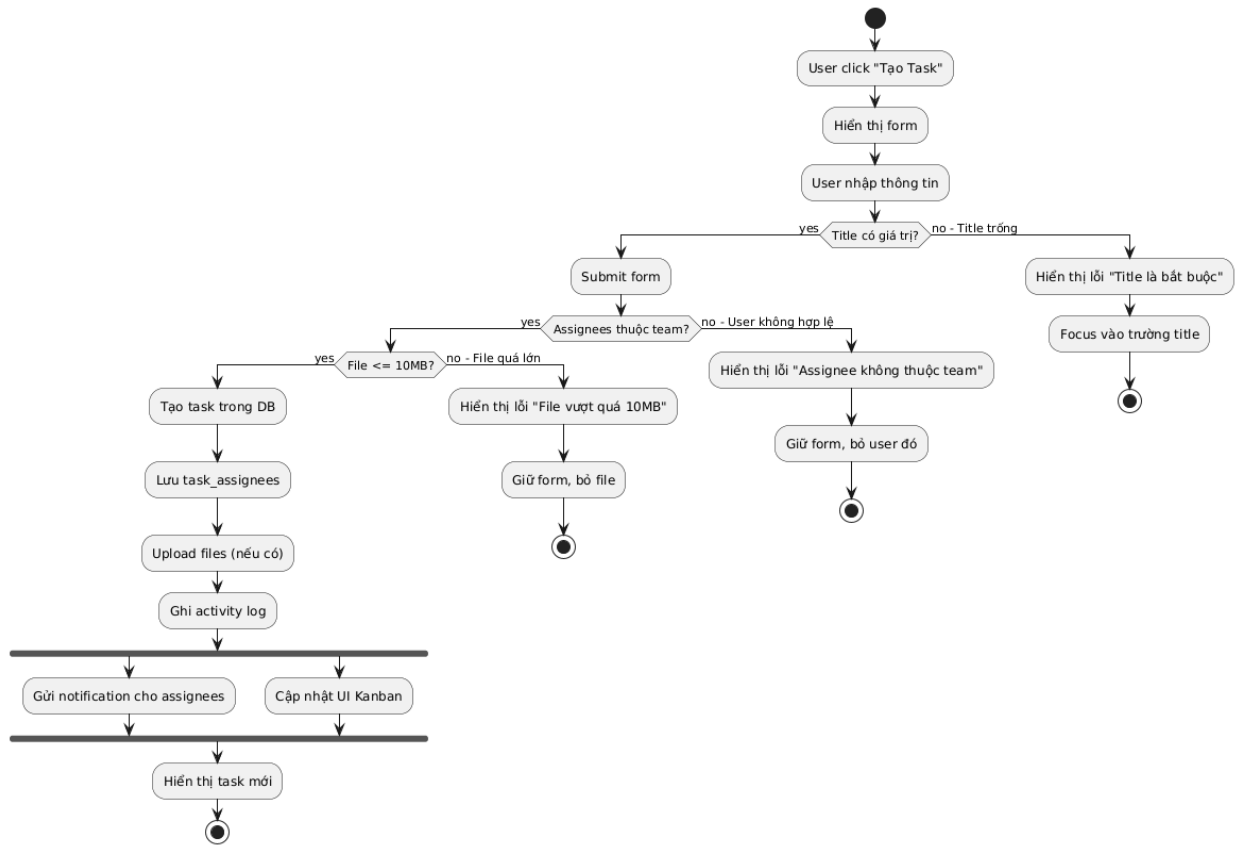


Diagram 8: Activity Diagram - Quy trình tạo Task mới

### Cảnh báo Task sắp hết hạn (Cron Job)

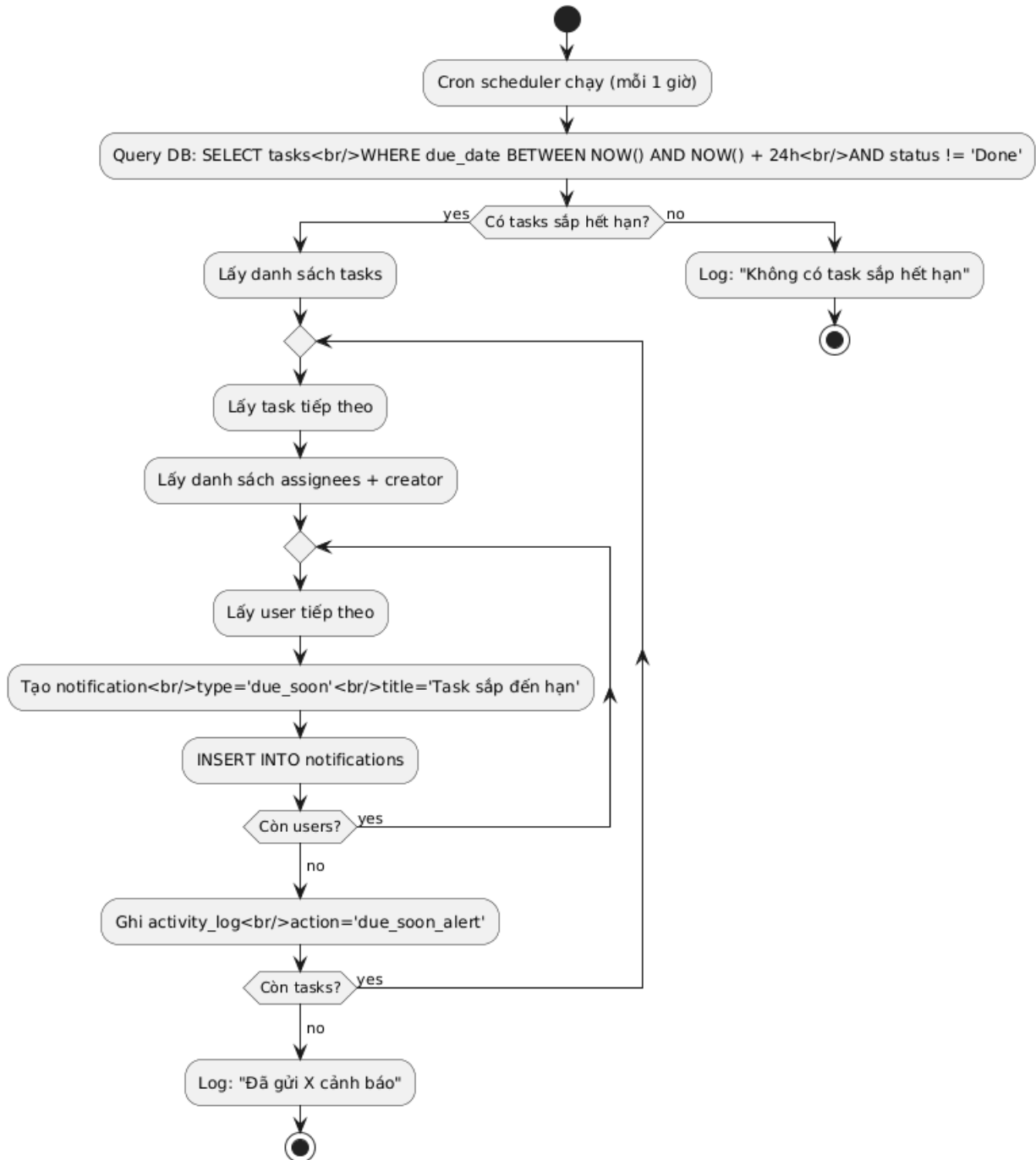


Diagram 9: Activity Diagram - Xử lý Cảnh báo Deadline

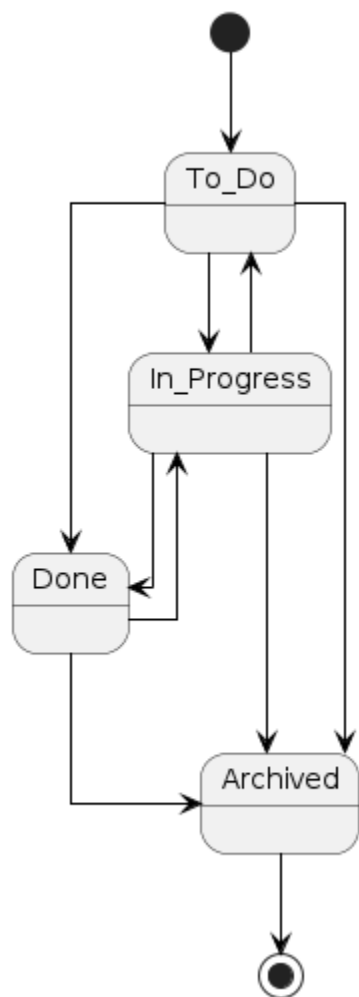


Diagram 10: State Diagram - Trạng thái Task

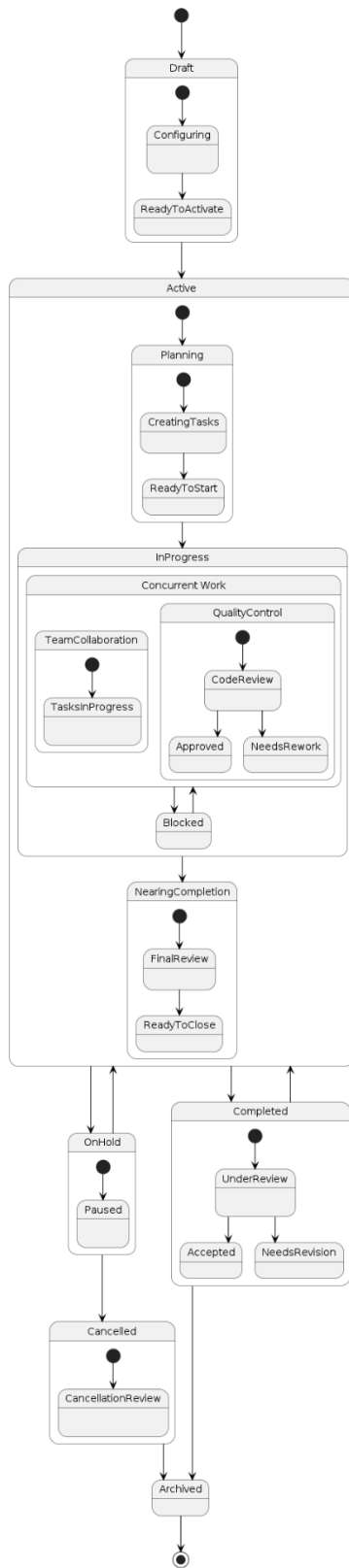


Diagram 11: State Diagram - Project Workflow States

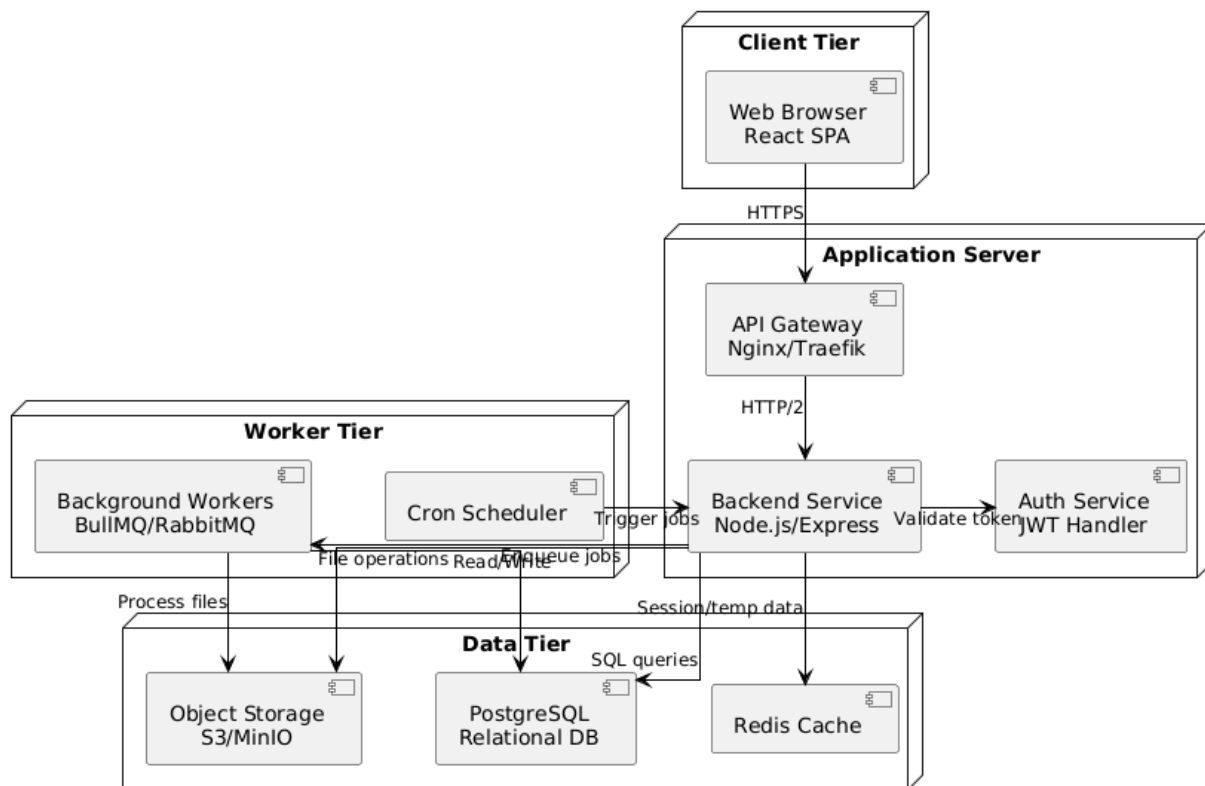


Diagram 12: Deployment Diagram - Triển khai hệ thống

**Tài liệu SRS - WorkFlow Mini**