

## 1. Tên Đề Tài

**Xây dựng hệ thống hỗ trợ ra quyết định phân công nhiệm vụ và cân bằng tải công việc cho đội nhóm bằng thuật toán phân tích.**

## 2. Tổng Quan & Mục Tiêu

### Bài toán thực tế

Trong các môi trường làm việc nhóm, việc phân công nhiệm vụ thường dựa trên kinh nghiệm và cảm tính của người quản lý. Điều này có thể dẫn đến các vấn đề như:

- **Phân công không tối ưu:** Giao việc không đúng sở trường của nhân viên, làm giảm hiệu suất.
- **Quá tải (Burnout):** Các thành viên chủ chốt liên tục nhận những công việc khó, trong khi các thành viên khác lại ít việc, gây lãng phí nguồn lực và giảm động lực làm việc.
- **Thiếu khách quan:** Khó có một cái nhìn tổng thể và dựa trên dữ liệu về khối lượng công việc thực tế của từng người.

### Mục tiêu của đồ án

Xây dựng một ứng dụng Web không chỉ là một công cụ quản lý công việc thông thường, mà còn là một **Hệ thống Hỗ trợ Ra quyết định (Decision Support System)**. Hệ thống sẽ sử dụng các thuật toán phân tích dữ liệu để cung cấp cho người quản lý những **gợi ý** và **cảnh báo** thông minh, giúp việc phân công trở nên hiệu quả và khách quan hơn.

## 3. Các Tính Năng Chính của Hệ Thống

Hệ thống được chia làm hai nhóm tính năng chính:

### Nhóm 1: Tính Năng Cốt Lõi (Nền tảng Quản lý)

- **Xác thực người dùng:** Cho phép người dùng đăng ký, đăng nhập an toàn vào hệ thống.
- **Quản lý Dự án:** Người quản lý có thể tạo ra các dự án mới và thêm các thành viên vào dự án đó.
- **Quản lý Công việc (Tasks):**
  - Tạo ra các công việc với đầy đủ thông tin: tiêu đề, mô tả, độ ưu tiên, ngày hết hạn, người thực hiện...
  - Giao diện bảng **Kanban** trực quan với các cột "Cần làm", "Đang làm", "Đã xong".
  - Người dùng có thể **kéo-thả** các công việc giữa các cột để cập nhật trạng thái.
- **Tương tác:** Cho phép các thành viên bình luận, trao đổi thông tin ngay trên từng

công việc.

## Nhóm 2: Tính Năng Thông Minh (Trọng tâm của đồ án)

Đây là các tính năng tạo nên sự khác biệt và giá trị cho đề tài.

- **Hệ thống Gợi ý Phân công:**

- **Kịch bản:** Khi người quản lý tạo một công việc mới và chỉ định các kỹ năng cần thiết (ví dụ: Java, React), hệ thống sẽ tự động phân tích và đề xuất 3 thành viên phù hợp nhất để nhận việc.
- **Thuật toán:** Gợi ý dựa trên "Điểm phù hợp" (Match Score), được tính toán từ:
  - **Điểm năng lực (Skill Score):** Hệ thống tự "học" và chấm điểm kỹ năng của mỗi nhân viên dựa trên lịch sử họ đã hoàn thành các công việc tương tự.
  - **Điểm quá tải (Workload Score):** Hệ thống tính toán khối lượng công việc hiện tại của mỗi người dựa trên số lượng và độ ưu tiên của các task họ đang đảm nhận.

- **Dashboard Cân Bằng Tải:**

- Cung cấp cho người quản lý một giao diện biểu đồ trực quan, thể hiện "Điểm quá tải" của tất cả thành viên trong đội nhóm theo thời gian thực.
- Hệ thống sẽ tự động đưa ra các **cảnh báo** khi một thành viên có dấu hiệu bị quá tải, giúp người quản lý có thể can thiệp và điều phối lại công việc kịp thời.

- **Hồ Sơ Năng Lực Động:**

- Trang thông tin của mỗi người dùng sẽ tự động hiển thị các kỹ năng mà họ giỏi nhất cùng với điểm số tương ứng, tạo ra một "bản đồ năng lực" cho cả đội.

## 4. Thiết Kế Cơ Sở Dữ Liệu (CSDL)

CSDL được thiết kế để lưu trữ không chỉ dữ liệu nghiệp vụ mà còn cả các thông tin cho việc phân tích.

- **Các bảng chính:**

- users: Lưu thông tin người dùng, vai trò, và **điểm quá tải** hiện tại.
- projects: Lưu thông tin các dự án.
- tasks: Lưu thông tin chi tiết về các công việc.
- skills: Lưu danh sách các kỹ năng (ví dụ: Java, Photoshop, Testing...).
- comments: Lưu các bình luận.

- **Các bảng trung gian (Bảng nối):**

- project\_members: Cho biết người dùng nào thuộc về dự án nào.
- task\_skills: Cho biết công việc nào yêu cầu những kỹ năng gì.
- user\_skills: Bảng quan trọng nhất, lưu trữ **điểm năng lực** của mỗi người dùng

đối với từng kỹ năng cụ thể.

## 5. Công Nghệ Sử Dụng

- **Backend:** Java 21, Spring Boot, Maven, PostgreSQL.
- **Frontend:** React (với TypeScript), Vite, Tailwind CSS.
- **Database:** PostgreSQL (quản lý bởi Docker).