

# BÁO CÁO CÁ NHÂN ĐỒ ÁN CUỐI KÌ

Lưu Thanh Thuý - 22127410

24 tháng 8, 2024

## **Tóm tắt nội dung**

Đồ án này tập trung chủ yếu vào các nội dung trọng tâm của môn học như thiết kế, lắp mạch và xây dựng Website quản lý thiết bị IoT từ xa thông qua Internet. Như bản kế hoạch, ý tưởng của nhóm làm về máy giám sát thú cưng tự động, có các chức năng như gửi tín hiệu khi sắp hết đồ ăn hay nước uống và tự làm đầy khi chủ nuôi gửi tín hiệu thông qua website.

## **Giáo viên hướng dẫn**

Cao Xuân Nam

Đặng Hoài Thương

## 1 Giới thiệu chung về ReactJS

**ReactJS** là một **opensource** được phát triển và ra mắt bởi Facebook vào 2013. Là một thư viện Javascript, sử dụng chủ yếu trong develop web để xây dựng các tương tác với các thành phần trên website, **ReactJS** được ưa chuộng rộng rãi hiện nay nhờ khả năng xây dựng các giao diện người dùng (UI) hiệu quả và linh hoạt.

## 2 Giới thiệu chung về Firebase

**Firebase** là một nền tảng phát triển ứng dụng web và di động được tạo ra bởi Google, bao gồm một loạt các dịch vụ khác nhau: lưu trữ dữ liệu RealTime, xác thực người dùng, phân tích, đẩy thông báo, và nhiều chức năng khác. Nó cho phép các nhà phát triển tập trung vào việc phát triển chức năng của ứng dụng của họ, thay vì phải lo lắng về hệ thống backend.

Các thư viện **Firebase** dành cho Arduino, như trên các module Wi-Fi **ESP32** và **ESP8266**, hỗ trợ kết nối dễ dàng giữa các thiết bị IoT và nền tảng **Firebase**. Điều này cho phép sự tích hợp nhanh chóng giữa phần cứng và các dịch vụ đám mây, như lưu trữ dữ liệu hoặc đẩy thông báo từ thiết bị đến ứng dụng di động. Đây là một điểm cộng lớn cho việc sử dụng **Firebase** các đồ án lần này.

Trong cấu trúc trang web có folder **context**, bao gồm file `AuthContext.tsx` là một phần quan trọng trong việc quản lý xác thực người dùng. File này cung cấp các chức năng cơ bản như đăng ký, đăng nhập, và đăng xuất người dùng thông qua việc tương tác với **Firebase**.

## 3 Các yêu cầu căn bản đã thực hiện



Hình 1: Sơ đồ luồng xử lý

### 3.1 Frontend (ReactJS):

- Người dùng tương tác với giao diện web (ReactJS) để thực hiện yêu cầu:
  - Bật/tắt laser
  - Cho thú cưng ăn/uống.
- Yêu cầu này sẽ được gửi tới Firebase qua Realtime Database.

### 3.2 Firebase (Backend):

- Firebase lưu trữ dữ liệu và quản lý trạng thái của yêu cầu từ người dùng.
- Khi có sự thay đổi trong dữ liệu (ví dụ: người dùng yêu cầu bật/tắt laser), Firebase sẽ đồng bộ ngay lập tức tới các thiết bị hoặc ứng dụng kết nối với nó (trong trường hợp này là thiết bị phần cứng).

### 3.3 Thiết bị ngoại vi:

- Thiết bị (module laser) được kết nối với Firebase qua một vi điều khiển là ESP32.
- Khi Firebase nhận lệnh bật/tắt từ ReactJS, nó sẽ cập nhật trạng thái dữ liệu trong thời gian thực.
- Thiết bị ngoại vi đọc giá trị từ Firebase và thực hiện hành động tương ứng (ví dụ: bật hoặc tắt laser).

## 4 Các yêu cầu nâng cao đã thực hiện

### 4.1 Sử dụng module OUTPUT ngoài chương trình học: Laser Modules

- Chức năng: Bật/tắt tia laser để phát hiện hoặc cảnh báo.
- Cách thức hoạt động:
  - Laser 1 và Laser 2 được kết nối lần lượt với các chân LASER1\_PIN (chân 26) và LASER2\_PIN (chân 15).
  - Tùy theo giá trị yêu cầu từ Firebase (Request/Laser1, Request/Laser2), các tia laser sẽ được bật hoặc tắt.

## 4.2 Xem lịch sử dữ liệu đã lưu trên cloud trên giao diện web

Mỗi khi người dùng cho thú cưng ăn hoặc uống, thông tin này sẽ được ghi lại và lưu trữ trên **Database Realtime** của ứng dụng. Điều này cho phép hệ thống cập nhật dữ liệu ngay lập tức, đảm bảo rằng tất cả các hành động liên quan đến việc chăm sóc thú cưng đều được ghi nhận đầy đủ và chính xác.

## 4.3 Thông báo về điện thoại bằng thông báo nhanh

Trong thư viện `react-push-notification`, import hàm `addNotification` để đẩy thông báo lên điện thoại. Thư viện này được thiết kế để đơn giản hóa quá trình triển khai thông báo đẩy trong các ứng dụng **ReactJS**, cho phép dễ dàng gửi thông báo mà không cần phải cấu hình quá phức tạp, có thể tùy chỉnh tiêu đề, nội dung, thời gian hiển thị, và một số tùy chọn khác để phù hợp với nhu cầu của mình.

Theo thiết kế của nhóm, sau khi người dùng thực hiện tùy chỉnh và bấm nút để gửi yêu cầu lên cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo xác nhận rằng yêu cầu đã được gửi thành công. Thông báo này nhằm đảm bảo rằng người dùng biết yêu cầu của họ đã được xử lý đúng cách và giúp cải thiện trải nghiệm sử dụng bằng cách cung cấp phản hồi ngay lập tức sau khi hành động được thực hiện.

## 4.4 Giao diện web cho phép đăng kí tài khoản (ghi thành công thông tin tài khoản vào database/cloud/file)

**Firestore Authentication** là một dịch vụ xác thực người dùng được cung cấp bởi **Firestore**. Nó cung cấp một cách dễ dàng để xác thực, đăng ký và quản lý người dùng trong trang web này.

Để đăng kí tài khoản, nhóm sử dụng hàm `createUserWithEmailAndPassword` để tạo một tài khoản mới với email và mật khẩu do người dùng nhập vào. Sau khi quá trình đăng kí thành công, người dùng sẽ được tự động điều hướng đến trang đăng nhập, nơi họ có thể sử dụng tài khoản vừa tạo để đăng nhập vào hệ thống. Trường hợp không đăng kí được do email đã được sử dụng sẽ có thông báo lỗi tương ứng.

## 4.5 Giao diện web cho phép đăng nhập hệ thống (đọc thành công thông tin tài khoản đăng nhập từ database/cloud/file)

Nhóm sử dụng hàm `signInWithEmailAndPassword` của **Firestore** để đăng nhập người dùng với email và mật khẩu đã cung cấp. Khi đăng nhập thành công, sẽ được điều hướng đến Dashboard.

Lưu ý, có sử dụng component có tên là `ProtectedRoute` được sử dụng như một lớp bảo vệ, đảm bảo rằng chỉ những người dùng đã đăng nhập mới có thể truy cập vào các trang web cụ thể. Nếu người dùng cố gắng truy cập vào các trang này mà chưa đăng nhập, họ sẽ bị chặn và không thể truy cập được nội dung, giúp nâng cao tính bảo mật.

Ngoài ra, để xử lý việc đăng xuất, nhóm sử dụng hàm `signOut` của **Firestore**. Khi người dùng chọn đăng xuất, hàm `signOut` sẽ kết thúc phiên làm việc của họ và họ sẽ được điều hướng trở lại trang đăng nhập. Điều này đảm bảo rằng sau khi đăng xuất, người dùng không còn quyền truy cập vào các trang được bảo vệ, và phải đăng nhập lại nếu muốn tiếp tục sử dụng các tính năng của ứng dụng.