**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**



**BÁO CÁO ĐỒ ÁN**

**LẬP TRÌNH NHÚNG CĂN BẢN**

**Đề tài**

**GAMEPAD**

*Giảng viên hướng dẫn:* **TRẦN NGỌC ĐỨC**

*Sinh viên thực hiện:*

1. Trần Nam Bàng 15520032
2. Nguyễn Trung Sĩ 15520726
3. Hồ Nhật Tường 13521012

Mục lục

[LỜI CẢM ƠN 4](#_1fob9te)

[CHƯƠNG 1 TỔNG QUAN ĐỀ TÀI 9](#_3znysh7)

[1.1.](#_2et92p0) Mục đích đối tượng và phạm vi nghiên cứu 9

[1.1.1.](#_tyjcwt) Mục đích nghiên cứu 9

[1.1.2.](#_3dy6vkm) Đối tượng nghiên cứu 9

[1.1.3.](#_1t3h5sf) Phạm vi nghiên cứu 9

[1.2.](#_4d34og8) Phương pháp thực hiện 9

[1.3.](#_2s8eyo1) Thuận lợi và khó khăn 9

[CHƯƠNG 2 CƠ SỞ LÝ THUYẾT 11](#_17dp8vu)

[2.1.](#_3rdcrjn) Giao thức UART 11

[2.1.1](#_26in1rg) Khái niệm 11

[2.1.2](#_lnxbz9) Các thông số cơ bản 11

[2.1.3](#_35nkun2) Hoạt động 12

[2.2.](#_1ksv4uv) Linh kiện được sử dụng trong quá trình thực hiện đồ án. 12

[2.2.1.](#_44sinio) Kit NuMaker Uno 12

[2.2.2.](#_2jxsxqh) Module NODEMCU ESP8266 13

[2.2.3.](#_z337ya) Động cơ rung. 13

[2.3.](#_3j2qqm3) Socket IO 14

[2.4.](#_1y810tw) Tổng quan về NodeJS 15

[2.4.1.](#_4i7ojhp) Khái niệm 15

[2.4.2.](#_2xcytpi) Ưu điểm 15

[2.4.3.](#_1ci93xb) Nhược điểm. 16

[2.5.](#_3whwml4) Tổng quan về mongodb 16

[2.5.1.](#_2bn6wsx) Khái niệm 16

[2.5.2.](#_qsh70q) Ưu điểm 16

[CHƯƠNG 3 THIẾT KẾ VÀ HIỆN THỰC HỆ THỐNG 17](#_3as4poj)

[3.1.](#_1pxezwc) Mô hình thiết kế tổng thể của hệ thống 17

[3.2.](#_49x2ik5) Thiết kế phần cứng 18

[3.3.](#_2p2csry) Quá trình xử lý tay cầm. 20

[3.4.](#_147n2zr) Quá trình xử lý ở server. 20

[3.5.](#_3o7alnk) Quá trình xử lý ở dữ liệu. 20

[3.6.](#_23ckvvd) Quá trình xử lý ở web client 20

[TÀI LIỆU THAM KHẢO 23](#_32hioqz)

# LỜI CẢM ƠN

Nhóm chúng em xin cảm ơn đến Khoa Kỹ Thuật máy tính và các thầy (cô) đã tạo điều kiện cho chúng em làm đồ án môn Lập Trình Nhúng Căn Bản (học kì I năm học 2018-2019)

Đặc biệt chúng em gửi lời cảm ơn tới thầy Trần Ngọc Đức và thầy Trần Hoàng Lộc đã tận tình giúp đỡ và hỗ trợ chúng em trong các giai đoạn thực hiện đề tài đồ án. Nhờ vậy, chúng em cũng tích lũy được vốn kiến thức và kinh nghiệm hơn trong việc xây dựng và làm việc nhóm hiệu quả.

Chúng em cũng không lường hết sai sót và độ hoàn thiện đề tài vì thế rất mong quý thầy (cô) góp ý để chúng em hoàn thành tốt hơn.

Chúng em xin chân thành cảm ơn và gửi lời chúc sức khỏe tới các thầy (cô).

DANH MỤC HÌNH VẼ

Hình 2.1 Sơ đồ truyền UART

Hình 2.2 Sơ đồ hoạt động của UART

Hình 2.3 Board NuMaker Uno

Hình 2.4 Node MCU ESP8266

Hình 2.5 Động cơ rung

Hình 2.6 Mô hình tổng quát

Hình 2.7 Schematic

Hình 2.8 PCB layout

Hình 2.9 Mạch hoàn chỉnh

Hình 3.1: Sơ đồ khối hoạt động của tay cầm.

Hình 3.2 Sơ đồ khối xử lý động cơ rung

DANH MỤC BẢNG

DANH MỤC TỪ VIẾT TẮT

UART: Universal Asynchronous Receiver – Transmitter

TÓM TẮT ĐỒ ÁN

Đồ án xây dựng trò chơi gamepad. Hệ thống gồm phần cứng là các tay cầm chơi game, các thiết bị kết nối internet, server, giao diện người dùng. Hệ thống mang tính thời gian thực.

# CHƯƠNG 1 TỔNG QUAN ĐỀ TÀI

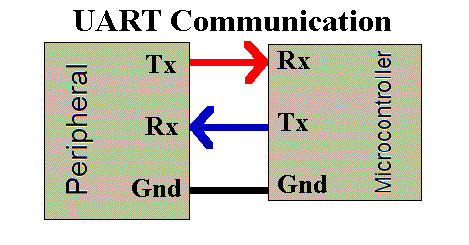
* 1. Mục đích đối tượng và phạm vi nghiên cứu
     1. Mục đích nghiên cứu
* Đồ án tập trung phát triển hệ thống chơi game realtime.
* Đồ án sử dụng kit NuMaker Uno dùng để điều khiển trung tâm các module hỗ trợ khác như ESP8266… để tạo một hệ thống hoàn chỉnh. Cụ thể đồ án thực hiện:
* Tìm hiểu giao tiếp giữa các thành phần trong hệ thống như kit NuMaker Uno, module ESP8266.
* Xây dựng server, database, web client để tạo thành một giao diện chơi game hoàn chỉnh.
  + 1. Đối tượng nghiên cứu
* Trò chơi bắn thuyền mang tính realtime.
  + 1. Phạm vi nghiên cứu
* Trong giới hạn đồ án, nhóm giới hạn nghiên cứu 2 tay cầm để hai người dùng có thể chơi với nhau. Ngoài ra việc mở rộng rất dễ dàng.
  1. Phương pháp thực hiện
* Nhóm đã tìm và nghiên cứu các tài liệu liên quan đến hệ thống để rút ra những thông tin cần thiết để phục vụ cho đồ án. Sau đó nhóm thực hiện tầng thành phần của hệ thống. Cuối cùng kết nối các thành phần của hệ thống và giải quyết các vấn đề phát sinh để hoàn thiện hệ thống.
* Tìm hiểu tài liệu, các dự án đã và đang được phát triển trên kit NuMaker Uno và datasheet của kit.
* Tìm hiểu về ESP8266 để lựa chọn phiên bản và nền tảng phù hợp. Sau đó tìm hiểu socket.IO để ứng dụng vào đồ án.
* Tìm hiểu về lập trình web qua các bài viết hướng dẫn trên mạng sử dụng HTML, JavaScipt để xây dựng giao diện web đơn giản.
* Tìm hiểu về socket.IO để kết hợp với NodeJs ở backend và ESP8266.
* Giải quyết các vấn đề chính xác của module, độ ổn định.
  1. Thuận lợi và khó khăn
     1. Thuận lợi
* Trong quá trình làm đề tài nhóm đã nhận được sự giúp đỡ của bạn bè, thầy cô.
* Ngoài ra sự phát triển của internet đã đem lại cho nhóm nhiều tài liệu bổ ích và rất thiết thực.
* Sự hợp tác tốt giữa các thành viên trong nhóm là một phần đóng góp thành công cho đồ án.
  + 1. Khó khăn
* Kiến thức và kinh nghiệm trong lập trình web và server còn hạn chế nên trong việc phát triển web về giao diện cũng như bảo mật còn gặp nhiều khó khăn.

# CHƯƠNG 2 CƠ SỞ LÝ THUYẾT

## Giao thức UART

1. Khái niệm

* UART (Universal Asynchronous Receiver – Transmitter) là bộ truyền nhận nối tiếp không đồng bộ.



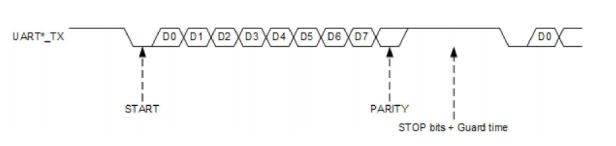
Hình 2.1 Sơ đồ truyền UART

1. Các thông số cơ bản

* Baund rate: khoảng thời gian dành cho 1bit được truyền. Phải được cài đặt giống nhau ở giữ và nhận.
* Frame: Khung truyền quy định về số bit trong mỗi lần truyền.
* Start bit: Là bit đầu tiên được truyền trong 1 Frame. Báo cho thiết bị nhận có một gói dữ liệu sắp được truyền đến.
* Data: Dữ liệu cần truyền. Bit có trọng số nhỏ nhất LSB được truyền trước sau đó đến bit MSB.
* Parity bit: Bit kiểm tra dữ liệu.
* Stop bit: là bit báo cho thiết bị rằng các data đã được gửi xong. Thiết bị nhận sẽ tiến hành kiểm tra khung truyền nhằm đảm bảo tính đúng đắn của dữ liệu.
* Stop bit: Là bit báo cho thiết bị rằng các data đã được gửi xong. Thiết bị nhận sẽ tiến hành kiểm tra khung truyền nhằm đảm bảo tín đúng đắn của dữ liệu.

1. Hoạt động

* Để bắt đầu cho việc truyền dữ liệu bằng UART, một START bit được gửi đi, sau đó là các bit dữ liệu và kết thúc quá trình truyền là bit STOP bit.



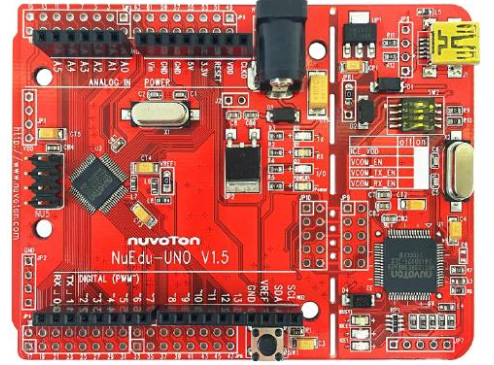
Hình 2.2 Sơ đồ hoạt động của UART

* Khi ở trạng thái chờ mức điện thế ở mức 1 (hight).
* Khi bắt đầu truyền, START bit sẽ chuyển từ 1 xuống 0 để báo hiệu cho bộ nhận là quá trình truyền dữ liệu sắp xảy ra.
* Sau START bit là đến các bit dữ liệ D0-D7 (các bit này có thể mức Hight hay Low tùy theo dữ liệu).
* Cuối cùng là STOP bit là bảo cho thiết bị rằng các bit đã được gửi xong. Thiết bị nhận sẽ tiến hành kiểm tra khung truyền nhằm đảm bảo tính đúng đắn của dữ liệu.

## Linh kiện được sử dụng trong quá trình thực hiện đồ án.

* + 1. Kit NuMaker Uno

Kit NuMaker Uno là vòng vi điều khiển 32bit của hãng Nuvuton, dựa trên lõi ARM Cortex-M0.



Hình 2.3 Board NuMaker Uno

* Thông số kỹ thuật.
* Kiến trúc ARM Cortex M0 32 bits với tốc độ tối đa 50 MHZ.
* Chuẩn giao tiếp: I2C, SPI, UART/USART.
* Ngoại vi: ADC, DMA, PWM, WDT, RTC.
* Điện áp hoạt động: 2.5 V to 5.5 V.

### Module NODEMCU ESP8266

* Module wifi được tích hợp nhân xử lý SoC ESP8266 sử dụng trong các sản phẩm IOT và ứng dụng truyền nhận dữ liệu qua wifi.



Hình 2.4 Node MCU ESP8266

* Thông số kỹ thuật.
* Tốc độ xử lý: 80 MHZ.
* WiFi: 2.4 GHz hỗ trợ chuẩn 802.11 b/g/n
* Điện áp hoạt động: 3.3V.
* Điện áp vào: 5V thông qua cổng USB.
* Số chân I/O: 11 (tất cả các chân I/O đều có Interrupt/PWM/I2C/One-wire, trừ chân D0).
* Số chân Analog Input: 1 (điện áp vào tối đa 3.3V).
* Bộ nhớ Flash: 4MB.
* Giao tiếp: Cable Micro USB.
* Hỗ trợ bảo mật: WPA/WPA2.
* Tích hợp giao thức TCP/IP.
* Lập trình trên các ngôn ngữ: C/C++, Micropython, NodeMCU – Lua.
  + 1. Động cơ rung.
* Động cơ rung điện thoại 0834 8x3.4mm loại động cơ rung có chức năng chuyển năng lượng điện thành năng lượng điện thành năng lượng cơ ở trạng thái rung, lắc....



Hình 2.5 Động cơ rung

* Thông số kỹ thuật.
* Điện áp cấp: 2.5~4VDC.
* Tốc độ: 12~25.000RPM.
* Dòng điện sử dụng: 70mA.
* Dòng điện khởi động: 90mA.

## Socket IO

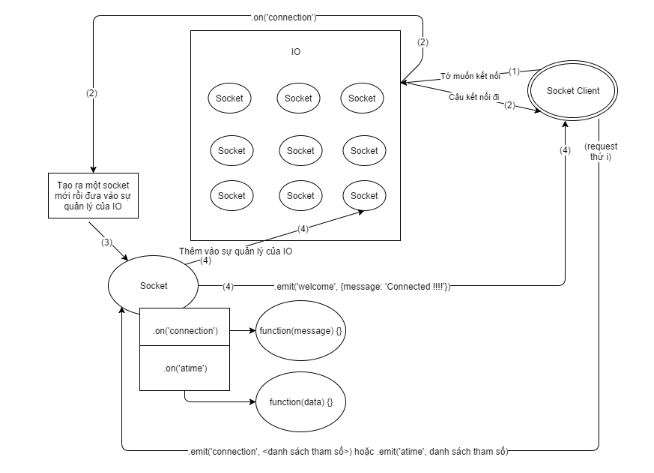
Socket IO là một bộ thư viện dành cho các ứng dụng web, mobile realtime. Với đặc trưng mạnh mẽ và dễ sử dụng. Thư viện gồm 2 phần:

* Phía client: Gồm bộ thư viện viết cho Web(JavaScript), IOS, Android.
* Phía server: Viết bằng JavaScript và dụng cho máy chủ node.JS

Một số hàm cơ bản:

* Connect(): kết nối với server socket.
* On (event\_name, listener): đăng kí lắng nghe sự kiện.
* Emit (event\_name, data): giử một sự kiện đi.
* Off(event\_name): ngừng lắng nghe một sự kiện nào đó.

Cơ chế hoạt động.



Hình 2.4 Mô hình hoạt động của socket IO

Khi client muốn kết nối tới server thì giử socket.emit() với tham số là “Connection” tới server và server dùng hàm socket.on() dùng để lắng nghe sự kiện từ client. Nếu như server nhận được sự kiện thì nó sẽ dùng hàm socket.emit() để trả về cho client với tham số “connected”. Socket sẽ được thêm vào sự quản lý của IO. Vì vậy mà ứng dụng dễ dàng quản lý và mở rộng khi cần thiết.

* 1. Tổng quan về NodeJS
     1. Khái niệm

Node.js là một nền tảng chạy trên môi trường V8 JavaScript runtime - một trình thông dịch JavaScript cực nhanh chạy trên trình duyệt Chrome.

* + 1. Ưu điểm

Phần core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.

Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.

Nodejs áp ụng cho các sản phẩn có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ.

Những ứng dụng có thể viết bằng Nodejs: Websocket server, Fast file upload client, ad server, cloud services, Restful API.

* + 1. Nhược điểm.
* Phải triển khai trên host nên việc đó không dễ dàng.
* Nodejs đang trong giai đoạn phát triển nên tài liệu đi kèm không ồn định, đem lại rủi ro.
* Do các packge là mã nguồn mở nên việc bảo mật trở nên rất khó khăn cho các ứng dụng.
  1. Tổng quan về mongodb
     1. Khái niệm

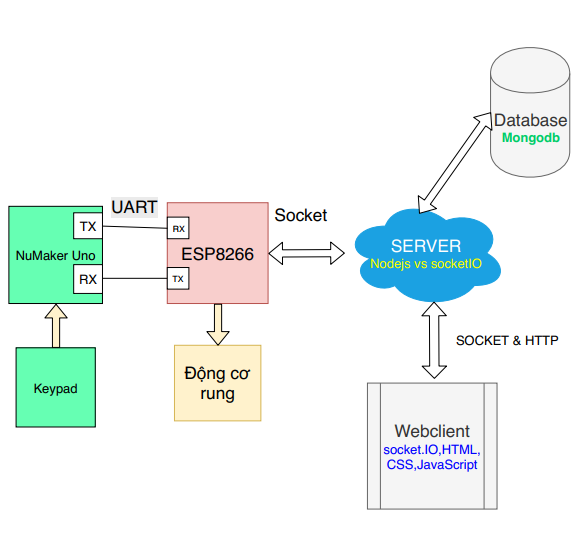
MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở thuộc họ NoSQL. Nó được thiết kế theo kiểu hướng đối tượng, các bảng trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ trên bảng không cần tuân theo một cấu trúc nhất định nào cả.

MongoDB lưu trữ dữ liệu theo hướng document, các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh

* + 1. Ưu điểm
* Schema linh hoạt
* Cấu trúc đối tượng rõ ràng.
* Truy vấn nhanh.
* MongoDB rất dễ mở rộng.
* Không có các join: Điều này cũng góp phần tạo nên tốc độ truy vấn cực - nhanh trên mongoDB.
* MongoDB phù hợp cho các ứng dụng realtime.

# CHƯƠNG 3 THIẾT KẾ VÀ HIỆN THỰC HỆ THỐNG

## Mô hình thiết kế tổng thể của hệ thống

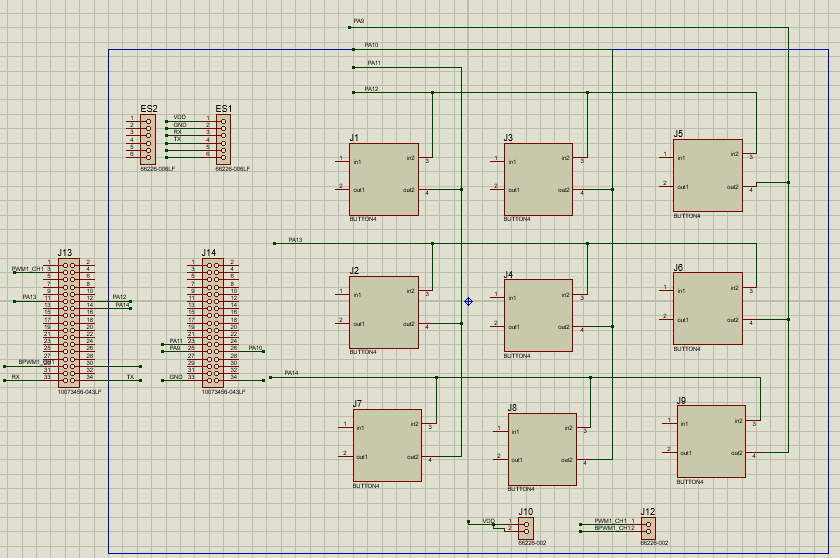


Hình 2.6 Mô hình tổng quát

* Hệ thống gồm 4 phần chính: Web client, Server, Database, Tay cầm.
* Web client được viết bằng ngôn ngữ HTML, CSS, JavaScript và socketIO. Web client để nhận dữ liệu từ server gửi xuống, hiển thị việc đặt tàu, đánh tàu, thời gian time out, lích sử người đánh game….
* Server được viết bằng ngôn ngữ nodejs với framework express và SocketIO. Dùng để nhận dữ liệu từ t gửi lên, xử lý cái yêu cầu từ Webclient, giao tiếp dữ liệu với database.
* Database sử dụng mongdb: MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở thuộc học NoSQL. Nó được thiết kế theo kiểu hướng đối tượng, các bảng trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ trên bảng không cần tuân theo một cấu trúc nhất định nào cả (điều này rất thích hợp để làm big data). MongoDB lưu trữ dữ liệu theo hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.
* Tay cầm gồm keypad, kit NuMaker Uno, ESP 8266. Các thao tác đặt thuyền, đánh thuyền được xuất phát từ keypad truyền qua NuMaker Uno, và NuMaker Uno truyền các thao tác đó choe ESP 8266 thông qua giao thức UART và ESP 8266 chuyển các thao tác đó lên server.
* Chức năng chính của hệ thống.
* Chơi game bắn thuyền giữ hai người.

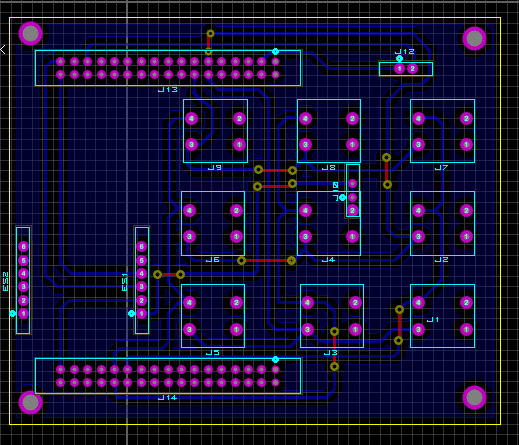
## Thiết kế phần cứng

* Mạch bao gồm các thành phần chính:
* Kit NuMaker Uno.
* Keypad.
* Node MCU ESP8266.
* Thiết kế mạch.



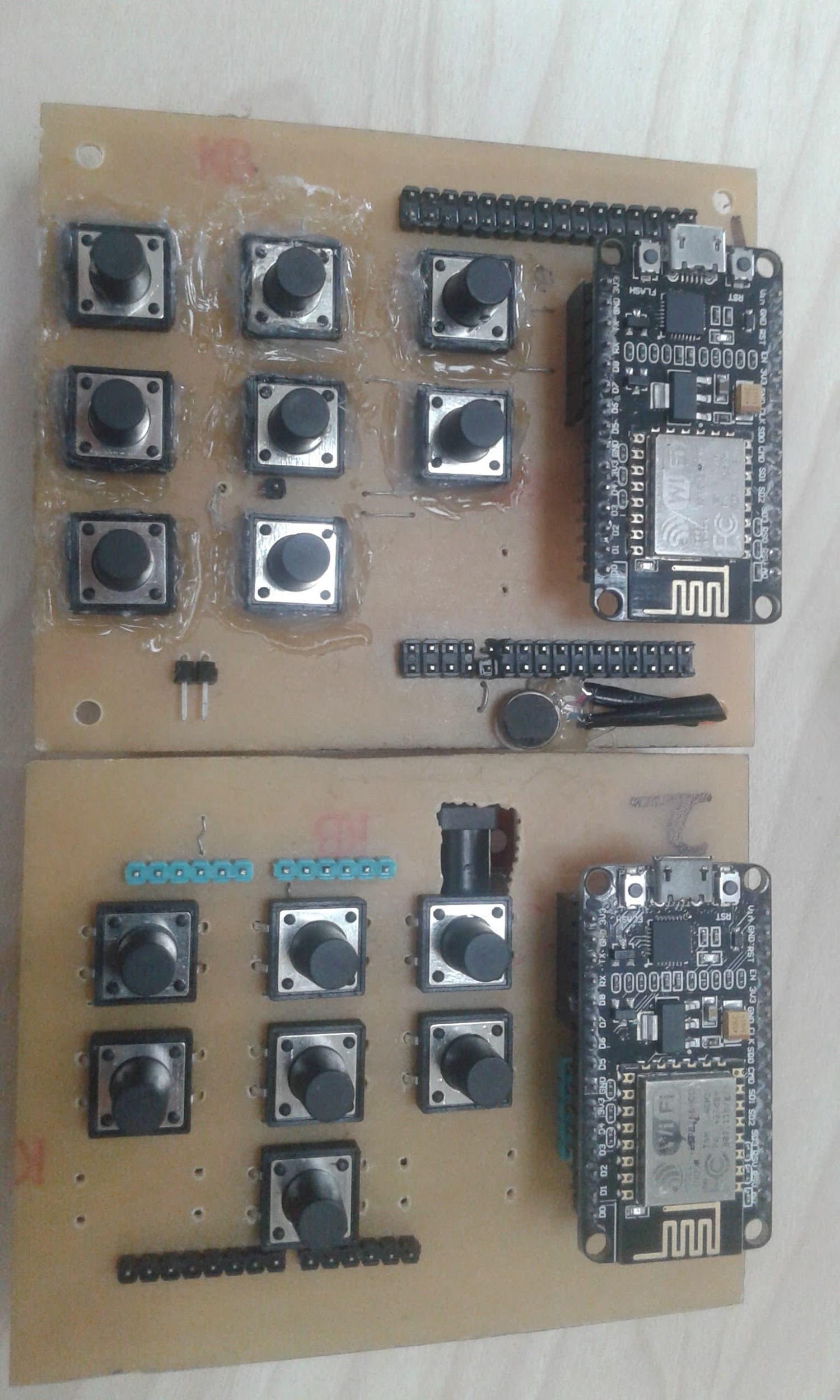
Hình 2.7 Schematic

* Thiết kế mạch layout



Hình 2.8 PCB layout

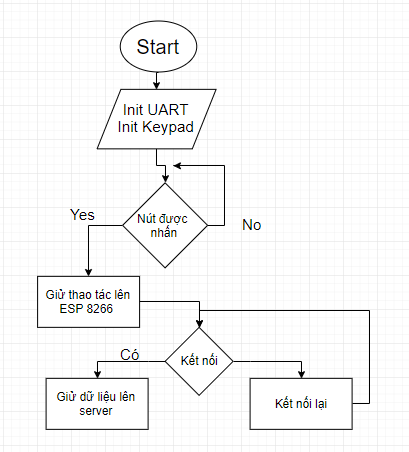
* Gia công PCB hoàn chỉnh.



Hình 2.9 Mạch hoàn chỉnh

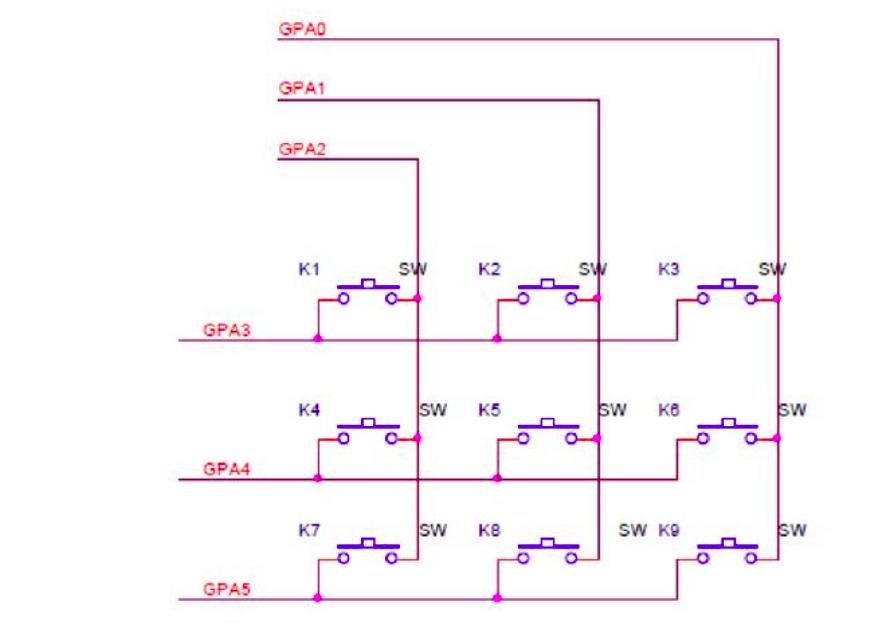
## Quá trình xử lý tay cầm.

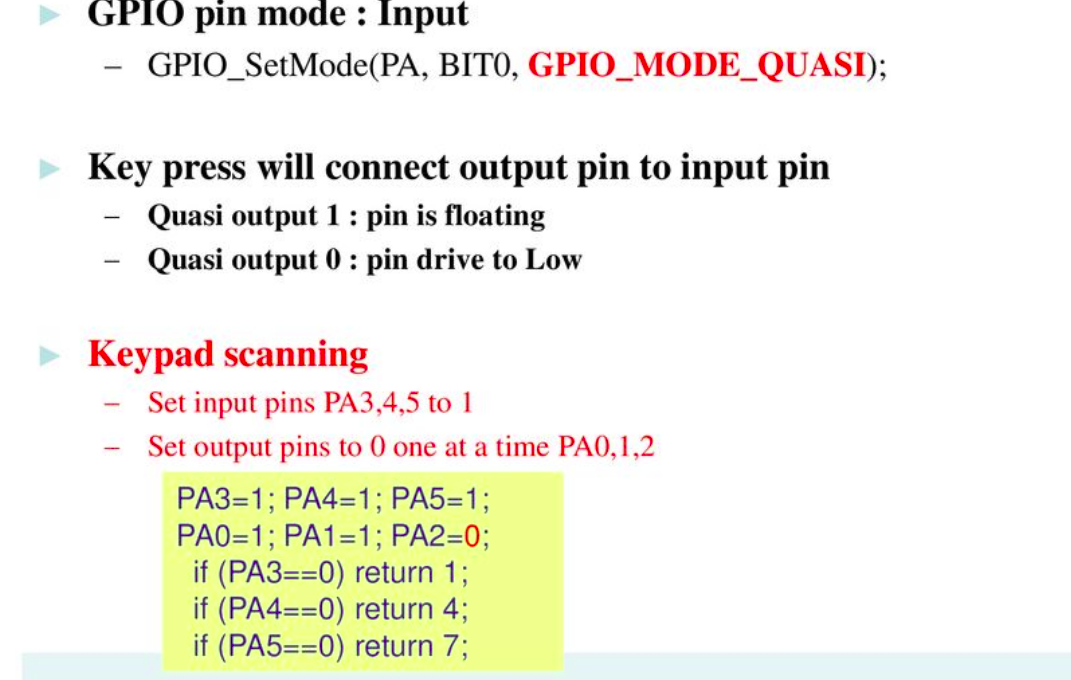
### 3.3.1. Quá trình thao tác nút.

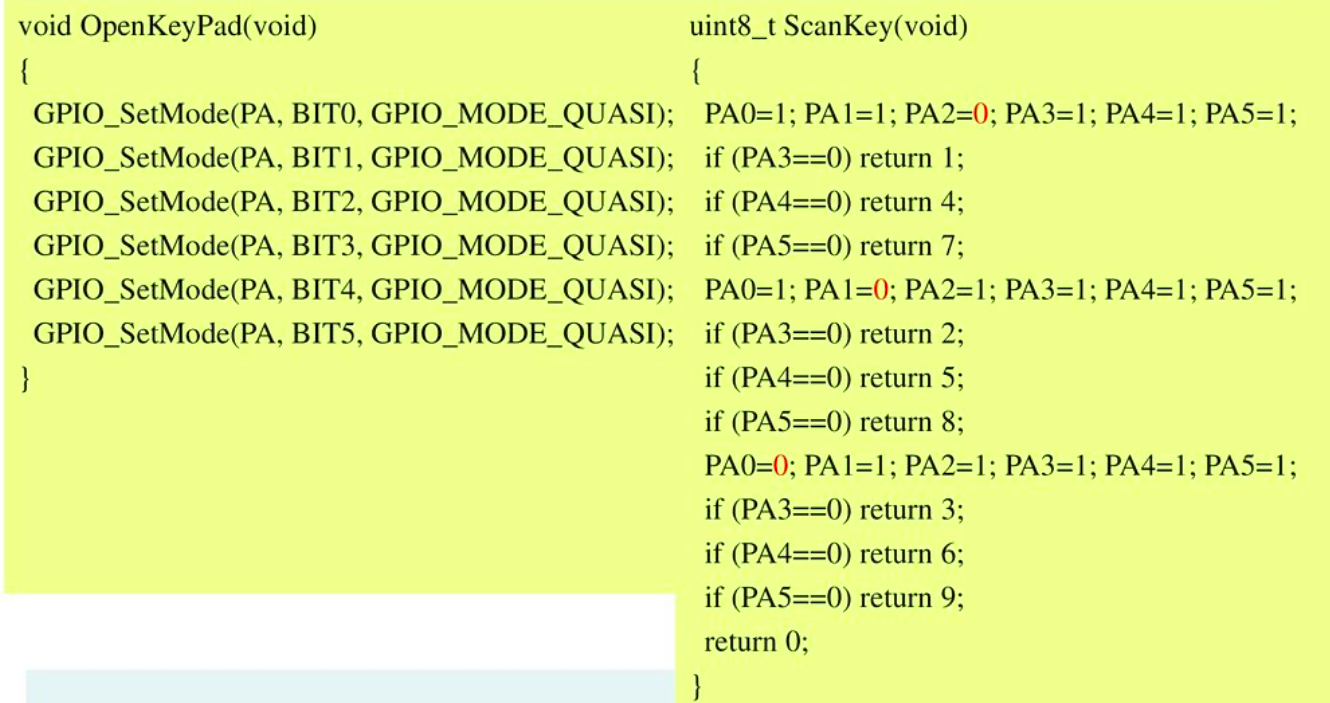
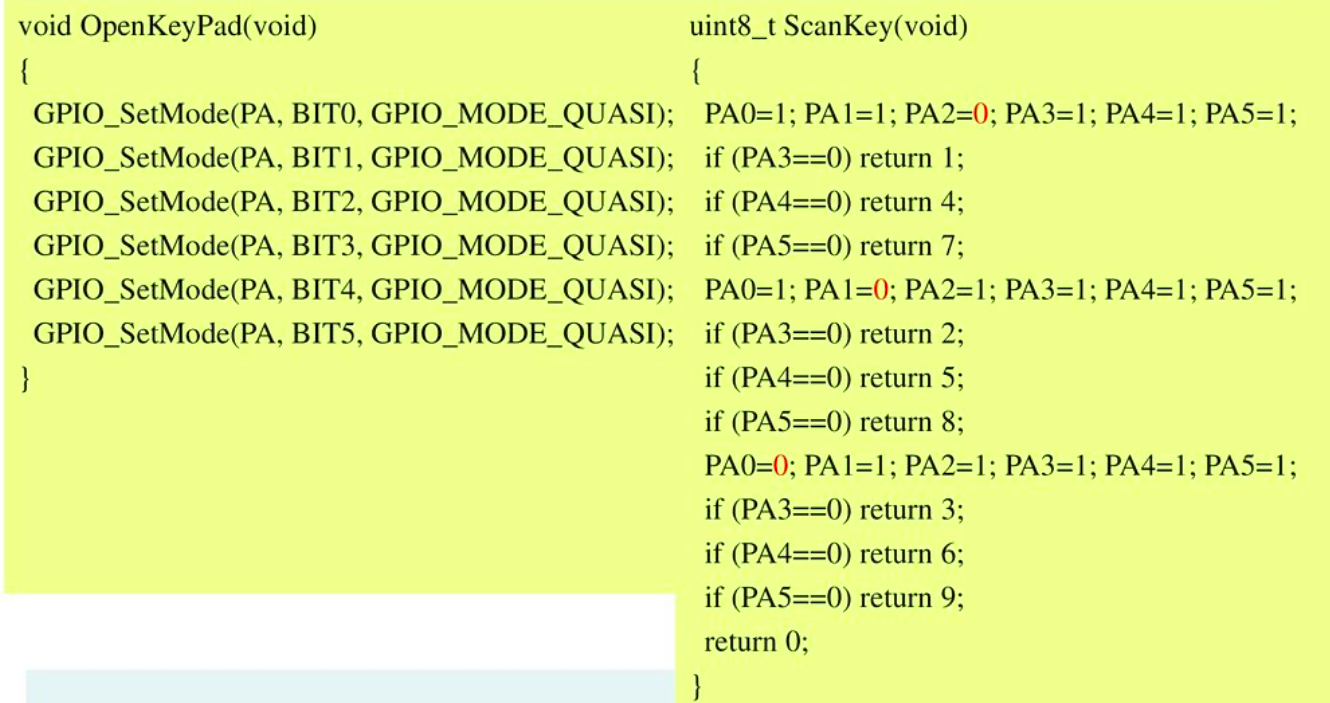


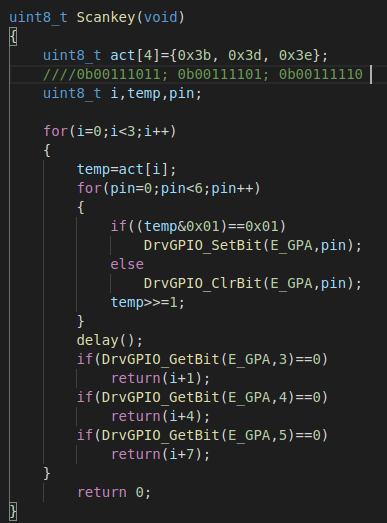
Hình 3.1: Sơ đồ khối hoạt động của tay cầm.

- xử lý nút ấn:





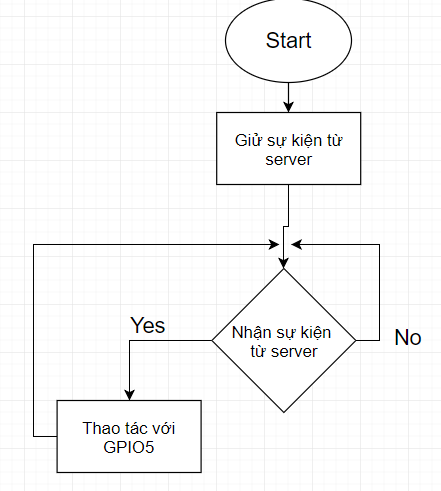




- có thể chọn 1 trong 2 chương trình scankey() trên.

Khi nút được nhấn thì NuMaker sẽ giử thao tác lên ESP 8266 thông qua UART. Nếu như có kết nối thì

### 3.3.2. Quá trình xử lý động cơ rung.



Hình 3.2 Sơ đồ khối xử lý động cơ rung

Khi nhận được sự kiện bị bắn trúng hay bắn trúng thuyền địch, win thì server gửi sự kiện về cho ESP 8266 và tùy mỗi sự kiện mà thao tác GPIO5 trên ESP 8266 khác nhau để làm cho động cơ rung mạnh hay yếu.

## Quá trình xử lý ở server.

-giao tiếp với client:

+board :board gửi chuỗi json tương ứng lên server khi nút nhấn được nhấn và nhận chuỗi json tương ứng khi bắn trúng hoặc bắn hụt  
 +web client : trao đổi thông tin với server và các client khác thông qua server bằng những phương thức tạo sự kiện và bắt sự kiện. từ board nhận được sự kiện gì thì server dẩy sự kiện đó xuống web client và web client sẽ thực hiện sự kiện đó (di chuyển thuyền, đặt thuyền, bắn thuyền)

-xử lý dữ liệu

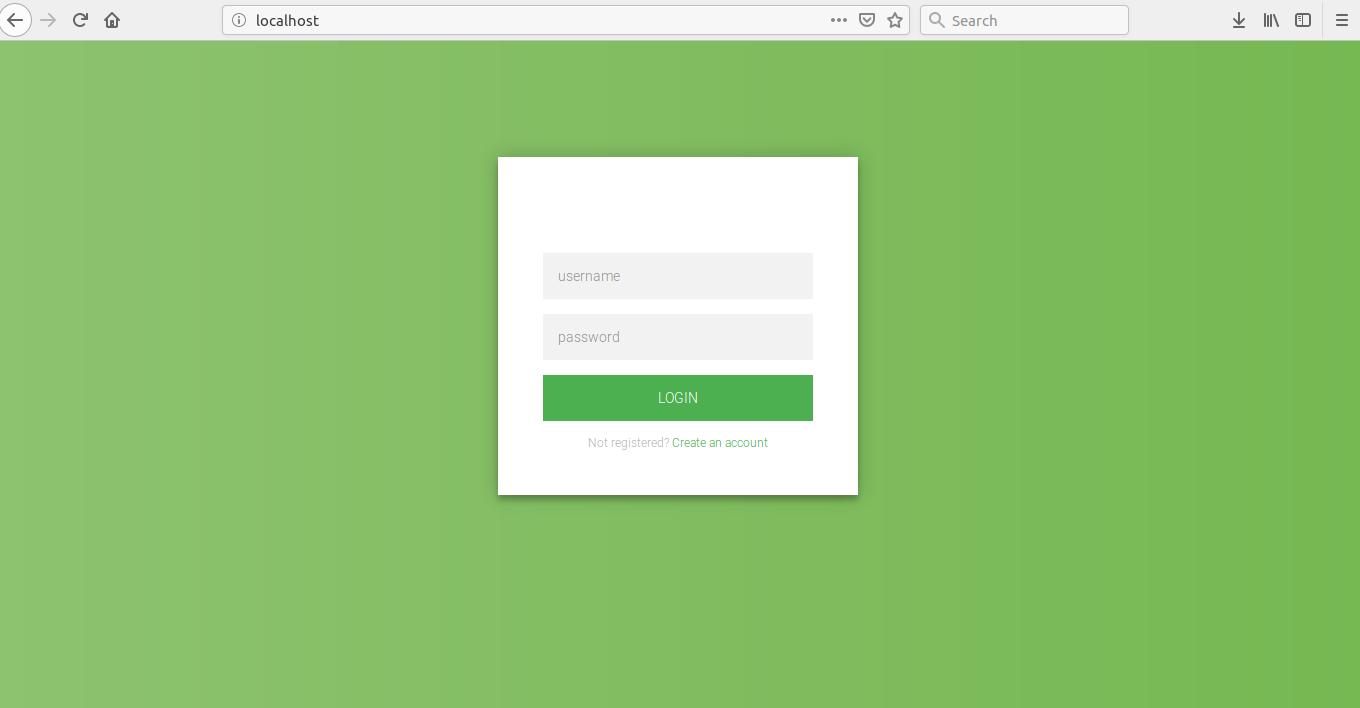
+thông tin người dùng: gồm có username và pass để đăng nhập được lưu vào database

+data của game: khi người chơi đặt thuyền xong web client sẽ tạo một mảng 2 chiều lưu giá trị của map, 1 tương ứng với co thuyền và 0 tương ứng với không có thuyền và gửi map lên server, khi server nhận được 2 map từ 2 user sẽ cho game bắt đầu và tạo ra một biến turn để chia lược chơi.

## Quá trình xử lý ở web client

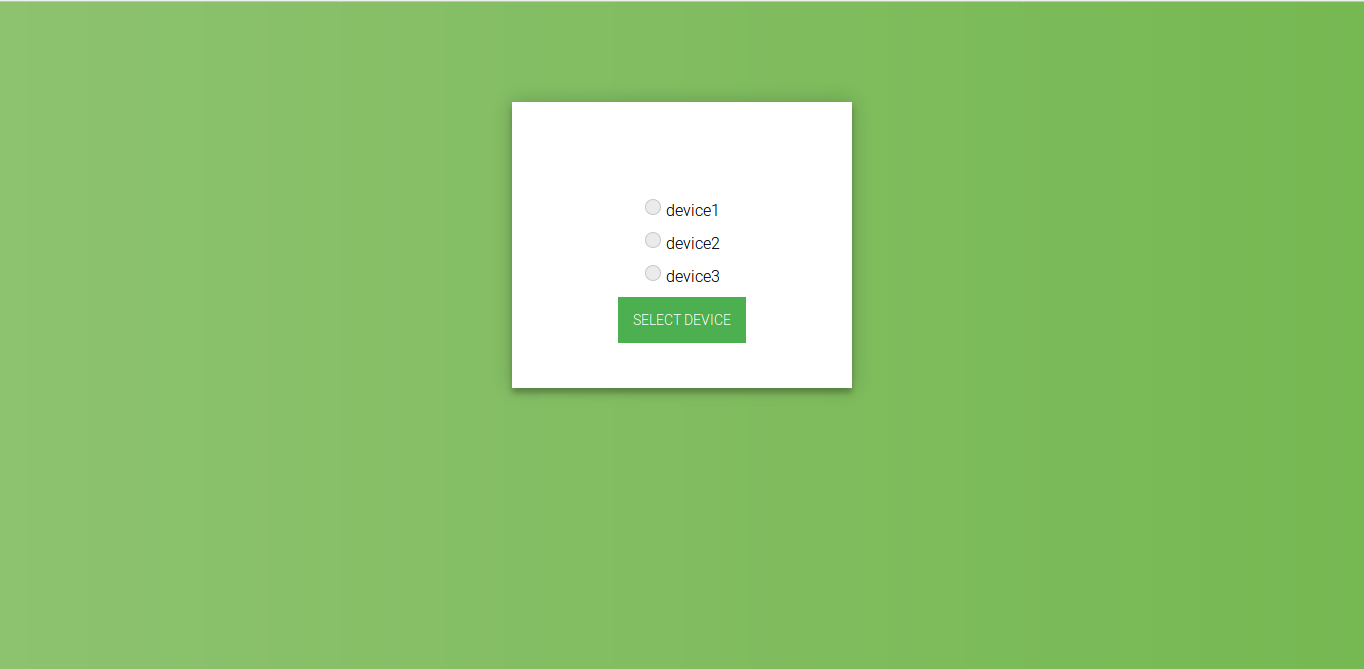
1. Đăng nhập

Người dùng sẽ được đăng ký tài khoản và sư dụng tại khoản đó để đăng nhập, khi đăng đăng nhập thành công web client sẽ tự tạo 1 seasion key ngăn có người cố tình truy cập đến trang chơi game bằng đường dẫn trực tiếp mà không qua đăng nhập



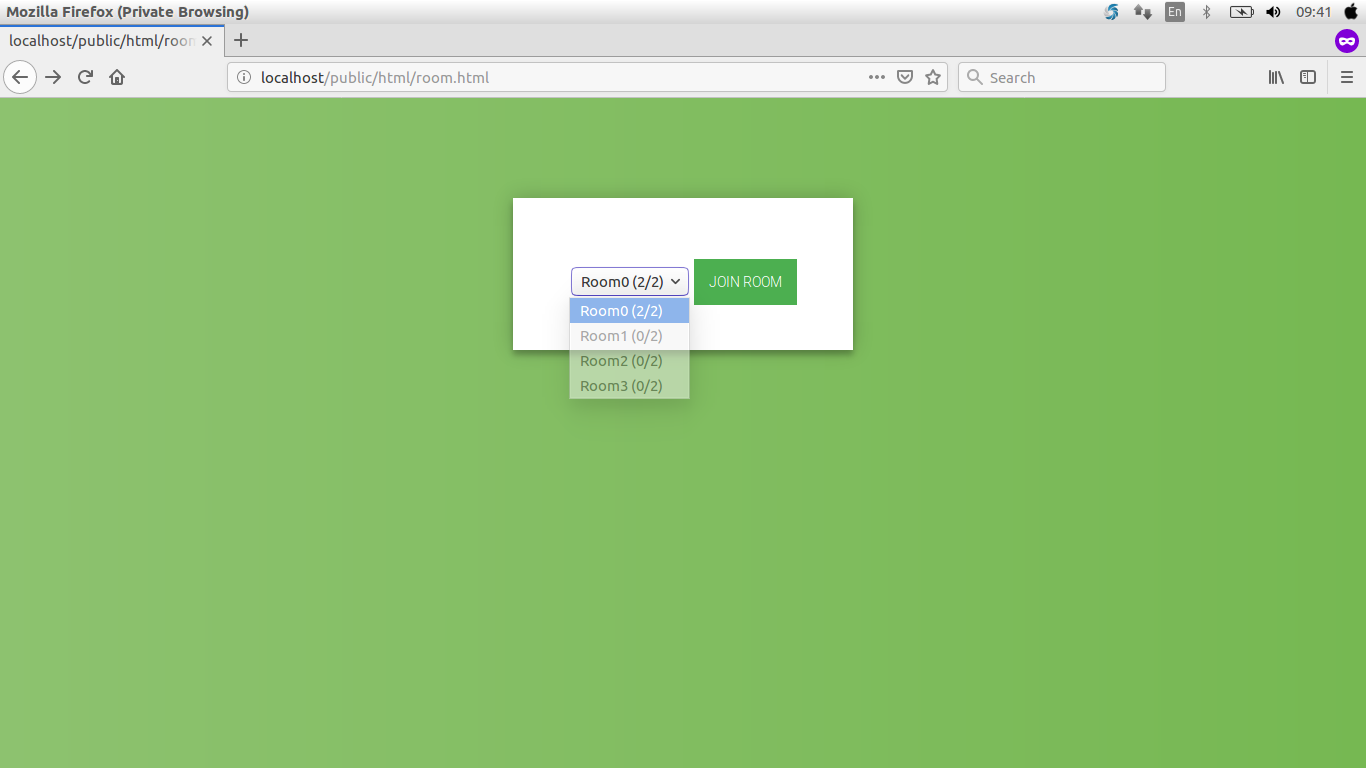
1. Chọn device

Khi board kết nối với server device sẻ hiện lên và user chọn để tiếp tục chơi game



1. Chọn phòng

Khi chọn phòng server sẽ bắt được sự kiện và kiểm tra xem phòng còn trống không, nếu đã đầy thì không cho vào nữa



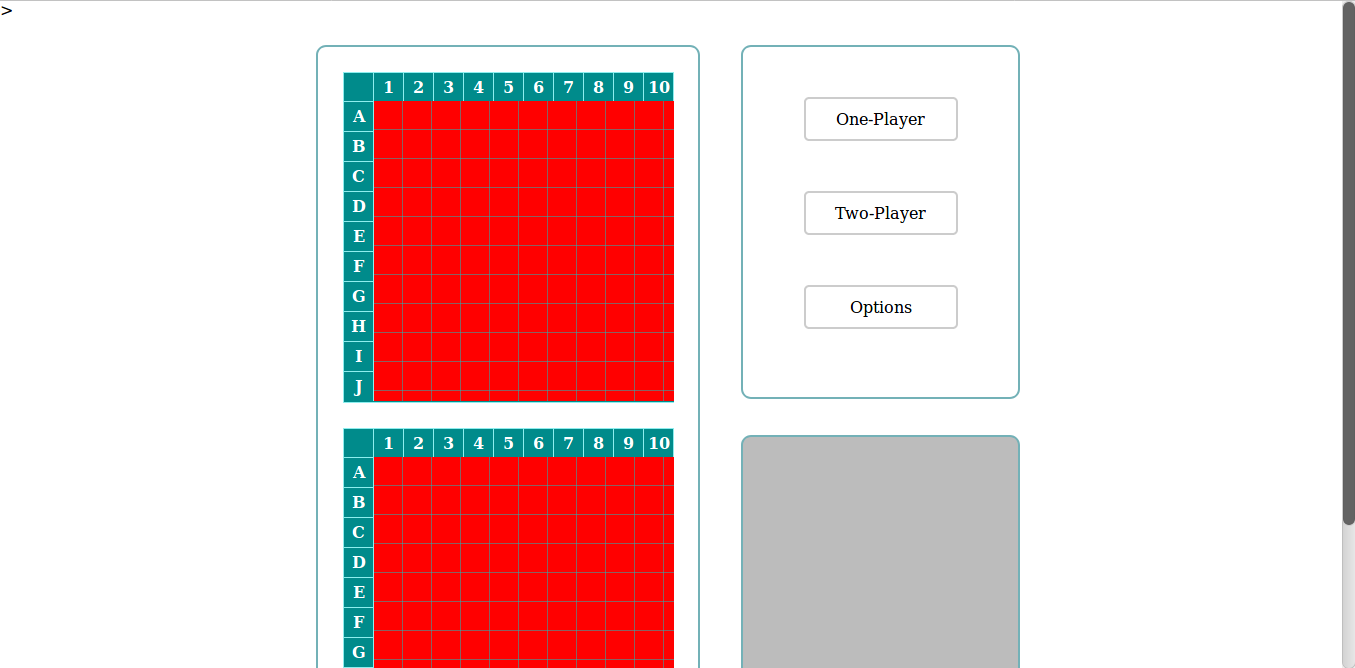
1. Chơi game

Khi web client nhận sự kiện từ server

+di chuyển thuyền: detect data sự kiện (right, left, up, down) và xử lý giao diện tương ứng

+đặt thuyển: đặt ngang và đặt dọc, tương ứng với 2 sự kiện khác nhau,

và đồng thời kiểm tra thuyền có bị đặt chồng lên nhau không, khi đặt đủ 4 thuyền con trỏ tự động di chuyển đến map địch để thực hiện bắn thuyền  
 +bắn thuyền : khi bắn tạo một sự kiện cho server bắt và server kiểm tra xem bắn trung hay hụt bằng cách duyện map của đối phương và trả về cho client biết trúng hay hụt mà có những hành động tương ứng



CHƯƠNG 4 KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ

* Kết quả

Xây dựng được hệ thống hoàn chỉnh gồm đặt phòng, chọn thiết bị chơi, đặt thuyền và bắn thuyền cho hai người chơi.

* Nhận xét và đánh giá
* Hướng phát triển

Xây dựng hệ thống hoàn chỉnh không bị lỗi, có thể chụi được nhiều người dùng vào chơi cùng một lúc.

# TÀI LIỆU THAM KHẢO

<https://en.wikipedia.org/wiki/MongoDB>

<http://arduino.vn/>

<http://www.nuvoton.com/hq/support/tool-and-software/development-tool-hardware/numaker-uno>

<https://github.com/singuyen123/PirateKing.git>

<https://slideplayer.com/slide/12975180/>